EnsembleGAN: Adversarial Learning for Retrieval-Generation Ensemble Model on Short-Text Conversation

Jiayi Zhang*
Turing Robot
zhangjiayi@uzoo.cn

Chongyang Tao*
ICST, Peking University
chongyangtao@pku.edu.cn

Zhenjing Xu
Turing Robot
xuzhenjing@uzoo.cn

Qiaojing Xie
Turing Robot
xieqiaojing@uzoo.cn

Wei Chen
Turing Robot
weichen@uzoo.cn

Rui Yan†
ICST, Peking University
ruiyan@pku.edu.cn

ABSTRACT

Generating qualitative responses has always been a challenge for human-computer dialogue systems. Existing dialogue systems generally derive from either retrieval-based or generative-based approaches, both of which have their own pros and cons. Despite the natural idea of an ensemble model of the two, existing ensemble methods only focused on leveraging one approach to enhance another, we argue however that they can be further *mutually* enhanced with a proper training strategy. In this paper, we propose ensembleGAN, an adversarial learning framework for enhancing a retrieval-generation ensemble model in open-domain conversation scenario. It consists of a language-model-like generator, a ranker generator, and one ranker discriminator. Aiming at generating responses that approximate the ground-truth and receive high ranking scores from the discriminator, the two generators learn to generate improved highly relevant responses and competitive unobserved candidates respectively, while the discriminative ranker is trained to identify true responses from adversarial ones, thus featuring the merits of both generator counterparts. The experimental results on a large short-text conversation data demonstrate the effectiveness of the ensembleGAN by the amelioration on both human and automatic evaluation metrics.

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking**; *Web applications*; • **Computing methodologies** → *Natural language processing*;

KEYWORDS

Generative adversarial network, short-text conversation, ensemble method, retrieval-based conversation, generation-based conversation

*Equal contribution.
†Corresponding author: Rui Yan (ruiyan@pku.edu.cn).

SIGIR '19, July 21–25, 2019, Paris, France

ACM ISBN 978-1-4503-6172-9/19/07…$15.00
https://doi.org/10.1145/3331184.3331193

ACM Reference Format:
Jiayi Zhang, Chongyang Tao, Zhenjing Xu, Qiaojing Xie, Wei Chen, and Rui Yan. 2019. EnsembleGAN: Adversarial Learning for Retrieval-Generation Ensemble Model on Short-Text Conversation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19), July 21–25, 2019, Paris, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3331184.3331193

1 INTRODUCTION

Natural language human-computer conversation has long been an attractive but challenging task in artificial intelligence (AI), for it requires both language understanding and reasoning [17]. While early works mainly focused on domain-specific scenarios such as ticket booking, the open-domain chatbot-human conversations has gained popularity recently, not only for their commercial values (e.g., Xiaoice[1] from Microsoft), but for the rapid growth of online social media as well, along with tremendous data available for data-driven deep learning methods to be proved worthwhile. Current conversation systems could be generally divided into two different categories, namely the retrieval-based and the generative-based approach.

Given an user input utterance (also called a query), a retrieval-based system usually retrieves a number of response candidates from a pre-constructed index, and then selects the best matching one as a response to a human input using semantic matching [25, 29, 33]. The retrieved responses usually have various expressions with rich information and language fluency. However, limited by the capacity of the pre-constructed repository, the selected response might seem less customized for unobserved novel queries .

Meanwhile the generative conversation system works differently, for it generates responses token by token according to conditional probabilistic language models (LM) such as seq2seq with attention [1], which generates appropriate and tailored responses to most queries, but often suffers from the lack of language fluency and the problem of universal responses (e.g., "I don't know" and "Me too") due to statistical model incapabilities [2]. Various ameliorations have been proposed to enrich the generation, either by better exploring internal features such as mutual-information-based objective function [9], dynamic vocabularies [30] and diverse beam search [23], or by incorporating external knowledge, such as topic information [31], cue words [15, 37], dialog acts [40], and common sense knowledge [38].

[1]http://www.msxiaoice.com/

On the other hand, studies seeking for an ensemble of both retrieval and generative approaches show great improvement to dialogue generation performance. Song et al. [18] proposed Multi-Seq2Seq model that focuses on leveraging responses generated by the retrieval-based dialog systems to enhance generation-based dialog systems, thus synthesizing a more informative response. Similarly, Weston et al. [27] designed a retrieval-and-refine model which treats the retrieval as additional context for sequence generator to avoid universal issues such as producing short sentences with frequent words. Wu et al. [28] introduced a prototype-then-edit paradigm for their conversation system by building a retrieval-based prototype editing with a seq2seq model that increases the diversity and informativeness of the generation results.

Despite the performance gain of an ensemble compared with either retrieval or generative model, previous works only focused on ameliorating one approach based on the other, still leaving great potentials for making further progress by allowing both methods to be mutually enhanced. Inspired by adversarial learning [5], we propose a generative adversarial framework for improving an ensemble on short-text conversation, which is called EnsembleGAN throughout the paper. Particularly, EnsembleGAN consists of two generators and a discriminator. The LM-like generator (G_1) is responsible for synthesizing tailored responses via a sequence-to-sequence framework, while the ranking-based generator (G_2) aims at selecting highly competitive negative responses from a pre-retrieval module and G_1, and finally the ranking-based discriminator (D) endeavors to distinguish the ground-truth and adversarial candidates provided by pre-retrieval module and two generators (G_1 and G_2).

The motivation behind is that through adversarial learning, with G_1 generating improved highly relevant responses, and G_2 providing enriched and fluent unobserved as well as synthetic candidates, the discriminative ranker could be further trained to identify responses that are highly correlated, informative and fluent, thus absorbing the merits of both its generative counterparts. The proposed EnsembleGAN framework is intuitively suited for improving a combination of any neural-based generative and retrieval approaches towards better global optimal results. The main contribution of this paper is three-folded and it's summarized as follows:

- We introduce a novel end-to-end generative adversarial framework that aims to mutually enhance both generative and retrieval module, leading to a better amelioration of a dialogue ensemble model.
- We make extensive studies on ensembles of various generators and discriminators, providing insights of global and local optimization from the ensemble perspective through both quantitative and qualitative analysis.
- We demonstrate the effectiveness of the proposed EnsembleGAN by performing experiments on a large mixed STC dataset, the gain on various metrics confirms that the ensemble model as well as each of its modules could all be enhanced by our method.

2 RELATED WORK

Open-domain dialogue systems have been attracting increasing attention in recent years. Researchers have made various progress on building both generative-based [15, 17, 20] and retrieval-based conversation system [22, 29, 33–35]. Besides, with the success of generative adversarial networks (GANs) [5] on computer vision such as image translation [41] and image captioning [3], studies of GAN applications also start to emerge in the domain of natural language processing (NLP), such as dialogue generation [10, 32], machine translation [36] and text summarization [13], all demonstrating the effectiveness of GAN mechanism in the domain of NLP. With respect to dialogue generation framework, the GAN-related researches could also be generally categorized as the GAN on generative-based and retrieval-based models.

As for sequence generation models, also regarded as sequential decision making process in reinforcement learning, Yu et al. [39] proposed seqGAN framework that bypasses the differentiation problem for discrete token generation by applying Monte Carlo rollout policy, with recurrent neural network (RNN) as generator and binary classifier as discriminator. What follows are RankGAN [11] which treats the discrimination phase as a learning-to-rank optimization problem as opposed to binary classification, dialogueGAN [10] that adapts the GAN mechanism on a seq2seq model for dialogue generation scenario with its discriminator capable of identifying true query-response pairs from fake pairs, as well as DPGAN [32] that promotes response diversity by introducing an LM-based discriminator that overcomes the saturation problem for classifier-based discriminators. Nevertheless, even state-of-the-art generative approaches couldn't achieve comparable performance as retrieval-based approaches in terms of language fluency and diversity of response generation.

As for retrieval-based models, Wang et al. [26] proposed IRGAN framework that unifies both generative and discriminative ranking-based retrieval models through adversarial learning. While the generator learns the document relevance distribution and is able to generate (or select) unobserved documents that are difficult for discriminative ranker to rank correctly, the discriminator is trained to distinguish the good matching query-response pair from the bad. However effective IRGAN is, in a conversation scenario, a pure retrieval system would always be limited by the constructed query-response repository. The adversarial responses, observed or not, might not be suitable for novel queries after all, which is a common problem for retrieval-based conversation system that is beyond IRGAN's capability.

While previous GAN-related studies only focused on the improvement of either generative-based or retrieval-based single approach, our work in this paper could be categorized as a unified GAN framework of the aforementioned GAN mechanism on both retrieval model and sequence generation model of an ensemble, which is constructed with each of its modules getting involved in adversarial learning with different roles. While being most related to rankGAN and IRGAN, our work has the following differences:

1) RankGAN only trains a language model through point-wise ranking of independent human-written and synthetic sentences, while EnsembleGAN trains a generative seq2seq model (G_1) through pair-wise ranking (D) of ground-truth and negative responses, with both G_1 and D conditioned on the user's query, let alone the existence of another strong competitor G_2 providing negative adversarial samples.

2) While IRGAN allows for both generative and discriminative retrieval model to compete against each other, EnsembleGAN allows for both rankers G_2 and D to compete against each other as ensembles, with the constant involvement of response generation module G_1 included in a more delicate three-stage sampling strategy.
3) EnsembleGAN unifies both GAN mechanism with a shared overall learning objective among all generators and discriminator, enhancing an ensemble of generative and retrieval-based approaches towards better global optimal results.

3 PRELIMINARIES

Before diving into details of our EnsembleGAN Framework, we first introduce the generation-based conversation model and the retrieval-based conversation model, which is the basis of our Ensemble model.

Response Generation Model. An LM-based probabilistic conversation model usually employs the seq2seq encoder decoder framework, where in general the encoder learns the query representation and the decoder generates the response sequence token by token based on encoder output [17]. For an RNN-based seq2seq model with attention mechanism [1], the generation probability of the current word w_t of the response given query q of length T_q could be generally modeled as follows:

$$\begin{aligned} p(w_t|w_{t-1},\cdots,w_1,q) &= f_{\text{de}}(s_t, w_{t-1}, c_t) \\ c_t &= f_{\text{att}}(s_t, \{h_i\}_{i=1}^{T_q}) \\ h_i &= f_{\text{en}}(w_i, h_{i-1}) \end{aligned} \tag{1}$$

where f_{en} and f_{de} are the recurrence functions. $h_i \in \mathbb{R}^{d_1}$ and $s_t \in \mathbb{R}^{d_2}$ represent the hidden state of the encoder and decoder, and c_t the context vector obtained by attention mechanism based on f_{att}, which often takes the form of a weighted sum of $\{h_i\}_{i=1}^{T_q}$. The weight factor is generally computed as a similarity between s_t and each $h_i \in \{h_i\}_{i=1}^{T_q}$, allowing the decoder to attend to different part of contexts at every decoding step. The cross entropy loss $\mathcal{L}_{ce} = -\sum y_t \log p(w_t)$ is often applied for the model training, with y_t the ground-truth corresponding word.

Response Ranking Model. Given a query q and some candidate provided by a fast pre-retrieval module[1], the ranking model learns to compute a relevance score between each candidate and the query q. Instead of the absolute relevance of individual responses (a.k.a, point-wise ranking), we train the model through pair-wise ranking, for a user's relative preference on a pair of documents is often more easily captured [26]. Hence, the probability of a response pair $\langle r_1, r_2\rangle$ with r_1 more relevant than r_2 (noted as $r_1 \succ r_2$) being correctly ranked can be estimated by the normalized distance of their matching relevance to q as:

$$\begin{aligned} p(\langle r_1, r_2\rangle|q) &= \sigma(g(q, r_1) - g(q, r_2)) \\ &= \frac{\exp(g(q, r_1) - g(q, r_2))}{1 + \exp(g(q, r_1) - g(q, r_2))} \end{aligned} \tag{2}$$

[1]We apply Lucene (https://lucenenet.apache.org/) to index all query-response pairs and use the built-in TF-IDF method to retrieve candidates, following Song et al. [18].

where σ is the sigmoid function, and $g(\cdot,\cdot)$ is the ranker's scoring function defined by any matching model. We train the ranker to rank the ground-truth response r_{pos} higher than a sampled negative candidate r_{neg}, with the pair-wise ranking loss L_{rank} defined as a hinge function [6]:

$$\mathcal{L}_{\text{rank}} = \frac{1}{N}\sum_{i=1}^{N} \max(0,\, \delta + g(q, r_{\text{neg}}) - g(q, r_{\text{pos}})) \tag{3}$$

where N is the number of $(q, \langle r_{\text{pos}}, r_{\text{neg}}\rangle)$ training samples and δ denotes the margin allowing for a flexible decision boundary. While both the response generation and ranking model could be used alone as single model, they form an ensemble when the latter reranks both pre-retrieved candidates and generated responses and finally selects the response of the top ranking.

4 ENSEMBLEGAN FRAMEWORK

4.1 Model Overview

Figure 1 illustrates the overall architecture of our proposed EnsembleGAN framework. Given a set of user queries $Q = \{q_1, q_2, \cdots, q_N\}$, the original ensemble applies both its generation and pre-retrieval module to synthesize and retrieve response candidates $\{\tilde{r}_1, \cdots, \tilde{r}_{M_1}\}$ and $\{\hat{r}_1, \cdots, \hat{r}_H\}$ for each $q \in Q$, respectively. All candidates are ranked together based on the scoring function $g(q, r)$ of the ranking module.

1) Generative seq2seq model. $G_{\theta_1}(\tilde{r}|q)$, which inherits from the generation module of the ensemble, is responsible for synthesizing response candidates $\{\tilde{r}_1, \cdots, \tilde{r}_{M_1}\}$ given query q, as depicted in Eq.(7), with the application of Monte Carlo (MC) roll-out policy. By combining with the ground-truth response r, we directly generate negative response pairs $\{\langle r, \tilde{r}_m\rangle\}_{m=1}^{M_1}$ aiming at receiving high ranking scores from discriminator, the process of which is also noted as $G_{\theta_1}(\langle r, \tilde{r}\rangle|q)$ for formulation coherence.

2) Generative ranking model. $G_{\theta_2}(\langle r, \hat{r}\rangle|q)$, which inherits from the response ranking model of the ensemble, learns to approximate the true relevance distribution over response pairs $p_{\text{true}}(\langle r, \hat{r}\rangle|q)$. Hence with the true response r, we generates highly competitive negative samples $\{\langle r, \hat{r}_h\rangle\}_{h=1}^{H}$ as specified in Eq.(9) so as to challenge the discriminator.

3) Discriminative ranking model. $D_\phi(\langle r, r_{\text{neg}}\rangle|q)$, which inherits from the same ranking model as G_2, endeavors however to distinguish the true response pairs from adversarial candidates provided by both generators (G_1 and G_2). After the adversarial training, all G_1, G_2 and D could be used alone as single model, or we could also form an improved ensemble consisting of a generation model G_1 and a ranking model (either G_2 or D) as described previously.

4.2 Adversarial Training for the Ensemble

4.2.1 Overall Objective. In our generative adversarial framework for the ensemble, both generators try to generate fake samples that get high ranking scores so as to fool the discriminator, the discriminator on the contrary is expected to distinguish the good samples from the bad by ranking more precisely as well as scoring down negative samples. We summarize the minimax game among generators G_1,G_2 and the discriminator D with the objective

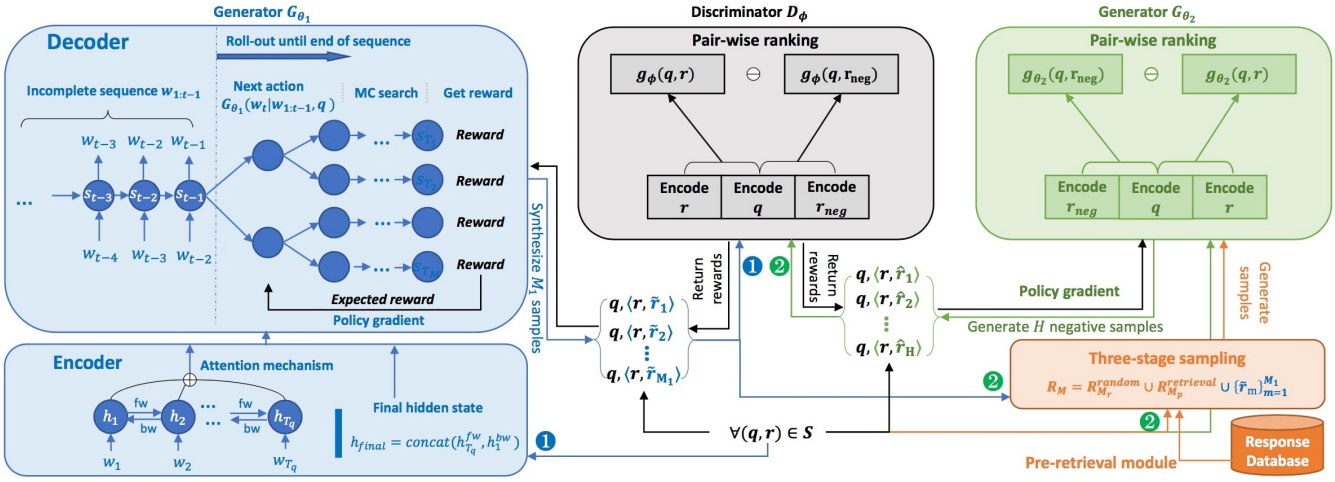


Figure 1: Illustration of EnsembleGAN Architecture (best viewed in color): generators G_1, G_2, discriminator D as well as three-stage sampling strategy are represented by blue, green, grey and orange colored blocks respectively.① and ② denote the training phase of G_1-steps and G_2-steps respectively, which is defined in algorithm 1.

function $\mathcal{L}$ as follows:

$$\begin{aligned}\mathcal{L} &= \min_{\theta_1,\theta_2}\max_{\phi} L(G_{\theta_1}, G_{\theta_2}, D_\phi) = \min_{\theta_1,\theta_2}\max_{\phi}(\mathcal{L}_1 + \mathcal{L}_2)\\ \mathcal{L}_1 &= \sum_{n=1}^{N}\mathbb{E}_{o\sim p_{\text{true}}(o|q_n)}[\log(D_\phi(o|q_n))]\\ \mathcal{L}_2 &= \sum_{n=1}^{N}\mathbb{E}_{o'\sim G_{\theta_1,\theta_2}(o'|q_n)}[\log(1-D_\phi(o'|q_n))]\end{aligned} \tag{4}$$

where $\mathbb{E}$ denotes the mathematical expectation, N the number of training samples, $o = \langle r, r_{\text{neg}}\rangle$ and $o' = \langle r, r'_{\text{neg}}\rangle$ are the true and generated response pair, respectively.

4.2.2 Optimizing Discriminative Ranker. As shown in Eq.(2) previously, we design $D_\phi(\langle r, r_{\text{neg}}\rangle|q) = p_\phi(\langle r, r_{\text{neg}}\rangle|q)$ to evaluate the probability of a response pair $\langle r, r_{\text{neg}}\rangle$ being correctly ranked given query q. Combining the ground-truth responses with the fake ones generated by both current G_1 and G_2, the optimal parameters of D_ϕ are obtained as follows:

$$\phi^* = \operatorname*{argmax}_{\phi}(\mathcal{L}_1 + \mathcal{L}_2) \tag{5}$$

where $\mathcal{L}_1$ and $\mathcal{L}_2$ are defined in Eq.(4), such an optimization problem is usually solved by gradient descent as long as D_ϕ is differentiable with respect to ϕ. When training generators, D_ϕ is used to provide reward of generated negative samples, which will be detailed later in this section.

4.2.3 Optimizing Generative Seq2Seq. At the first stage, we enhance the generative seq2seq model G_{θ_1} through discriminative ranker D_ϕ. When given a user query q, the generation of a sequence $\tilde{r} = \{w_0, w_1, ..., w_T\}$ could be regarded as a series of decision making at T time steps by policy $\pi = G_{\theta_1}(w_t|w_{1:t-1}, q)$ as defined in Eq.(1). However, since D_ϕ only provides the reward for a complete sequence, the lack of intermediate reward for every time step leads to the ignorance of long term reward causing the model to be short-sighted. We hence apply MC roll-out policy [11, 39] to tackle with the problem, which repeatedly rolls out incomplete sequences until the end-of-sequence token so as to get an expected reward from D_ϕ for every time step. With the true response r, the expected end reward of a response pair $o' = \langle r, \tilde{r}\rangle$ is defined as follows:

$$\begin{aligned}J_{\theta_1}(o'|q) &= \sum_{t=1}^{T}\mathbb{E}_{o'_m\sim G_{\theta_1}(o'_m|q)}[\log D_\phi(o'_m|q)|w_{1:t-1}]\\ &= \sum_{t=1}^{T}\sum_{w_t} G_{\theta_1}(w_t|w_{1:t-1}, q)Q_{D_\phi}^{G_{\theta_1}}(w_{1:t-1}, w_t|q, \text{MC}_{m_1}^{\pi_r})\end{aligned} \tag{6}$$

where $w_{1:t-1}$ is the current state with $t-1$ tokens already generated in $\tilde{r}$, w_0 the initial token. The response pair $o'_m = \langle r, w_{1:T_m}\rangle$, where $w_{1:T_m}$ is the completed T_m-length sequence rolled out from current $w_{1:t-1}$ according to m_1-time MC roll-out policy π_r (noted as $\text{MC}_{m_1}^{\pi_r}$), resulting in the action-value function defined as follows:

$$Q_{D_\phi}^{G_{\theta_1}}(w_{1:t-1}, w_t|q, \text{MC}_{m_1}^{\pi_r}) = \begin{cases}\dfrac{1}{m_1}\displaystyle\sum_{m=1}^{m_1}\log D_\phi(o'_m|q), & \text{for } t < T\\ \log D_\phi(o'|q), & \text{for } t = T\end{cases} \tag{7}$$

Hence, the instant reward for time step t is calculated as the average ranking scores from D_ϕ of all sampled response pairs $\{o'_m\}_{m=1}^{m_1}$ obtained by repeatedly rolling out $w_{1:t-1}$ for m_1 times based on $\text{MC}_{m_1}^{\pi_r}$. We note $M_1 = m_1 * T$ as the total number of generations for $\tilde{r}$ of length T. In contrast to the original rankGAN, both generator $G_{\theta_1}(w_t|w_{1:t-1}, q)$ and discriminator $D_\phi(o'|q)$ are conditioned on the given query q, which is a necessary adaptation in the case of dialogue generation. Note that such a configuration could also be referred to as conditionalGAN framework [14].

4.2.4 Optimizing Generative Ranker. The second stage involves the amelioration of generator G_2 through discriminator D_ϕ, with the objective function defined as below:

$$J_{\theta_2|\theta_1}(q) = \mathbb{E}_{o'\sim G_{\theta_2|\theta_1}(o'|q)}[\log(1-D_\phi(o'|q))] \tag{8}$$

where $\theta_2|\theta_1$ denotes that this second stage is actually based on the first stage discussed above, with G_{θ_1} fixed as a result. Inheriting from the same ranking model as D_ϕ, we train G_{θ_2} to generate competitive negative response pairs that receive high ranking scores from D_ϕ, where both ranking-based generative and discriminative

models could get improved [26]. More precisely, when given a true (q, r) pair and a scoring function g_{θ_2}, the chance of G_{θ_2} selecting a negative sample $o'_h = (r, \hat{r}_h)$ according to the relevance distribution of response pairs $\{\langle r, \hat{r}_h\rangle | \hat{r}_h \succ r, \hat{r}_h \in R_M\}$ is defined by a softmax function as follows:

$$\begin{aligned} G_{\theta_2}(o'_h|q) &= \frac{\exp(g_{\theta_2}(q, \hat{r}_h) - g_{\theta_2}(q, r))}{\sum_{\hat{r}_h \in R_M} \exp(g_{\theta_2}(q, \hat{r}_h) - g_{\theta_2}(q, r))} \\ &= \frac{\exp(g_{\theta_2}(q, \hat{r}_h))}{\sum_{\hat{r}_h \in R_M} \exp(g_{\theta_2}(q, \hat{r}_h))} = P_{\theta_2}(\hat{r}_h|q) \end{aligned} \tag{9}$$

where R_M represents the M-sized candidate pool with ground-truth responses excluded. Despite other possible configurations as observed in Wang et al. [26], we follow $G_{\theta_2}(o'_h|q) = P_{\theta_2}(\hat{r}_h|q)$ as directly being the relevance distribution of an individual response $\hat{r}_h$, not only for the simplicity, but for the coherence of both G_{θ_1} and G_{θ_2} being able to sample responses independently of the ground-truth response, as it's the real usage case after training.

The candidate pool R_M is of importance for the capability of G_{θ_2} to sample H unobserved as well as highly competitive responses. In addition to the random sampling strategy that generates M_r random responses ($R^{\text{random}}_{M_r}$) from the database as the original IRGAN, we apply both the pre-retrieval module to retrieve M_p candidates ($R^{\text{retrieval}}_{M_p}$) similar to ground-truth responses regardless of queries, and also G_{θ_1} to synthesize M_1 relevant responses, all of which are summarized as a three-stage sampling strategy:

$$R_M(M_r, M_p, M_1) = R^{\text{random}}_{M_r} \cup R^{\text{retrieval}}_{M_p} \cup \{\tilde{r}_m\}_{m=1}^{M_1} \tag{10}$$

The design of R_M not only compensates for the ineffectiveness of random sampling for generating competitive responses from a huge dialogue database in our case, it also enables the generator G_2 to work as an ensemble with the response generation model G_1, thus always considering the cooperation of both generative-based and retrieval-based approaches during adversarial learning.

4.2.5 Policy Gradient. Following Sutton et al. [19], we apply policy gradient to update generators' parameters through feedback of D_ϕ , for the sampling process of both generators are non-differential. Hence, with D_ϕ fixed, for each query q with true-negative response pair $o' = (r, r'_{\text{neg}})$, the minimization of $\mathcal{L}$ in Eq.(4) with respect to θ_1,θ_2 could be deduced as follows [11, 26]:

$$\begin{aligned} &\min_{\theta_1,\theta_2} \mathcal{L} = \max_{\theta_1} \sum_{n=1}^{N} \mathbb{E}_{o'\sim G_{\theta_1}} J_{\theta_1}(o'|q_n) - \max_{\theta_2|\theta_1} \sum_{n=1}^{N} J_{\theta_2|\theta_1}(q_n) \\ &\nabla_{\theta_1} J_{\theta_1}(o'|q_n) \simeq \sum_{t=1}^{T} \sum_{w_t} \nabla_{\theta_1} \log G_{\theta_1}(w_t|w_{1:t-1}, q_n) Q^{G_{\theta_1}}_{D_\phi} \\ &\nabla_{\theta_2} J_{\theta_2}(q_n) \simeq \frac{1}{H} \sum_{h=1}^{H} \nabla_{\theta_2} \log G_{\theta_2}(o'_h|q_n) \log D_\phi(o'_h|q_n) \end{aligned} \tag{11}$$

where J_{θ_1}, J_{θ_2} are defined in Eq.(6) and Eq.(8) respectively. ∇ is the differential operator, T the generated sequence length by G_{θ_1} and H the negative sampling size of G_{θ_2}.

4.2.6 Reward Setting. Normally, we would consider that the reward $R \equiv \log D_\phi(r, r_{\text{neg}}|q)$. It's however problematic that the logarithm leads to instability of training [5]. We thus follow Wang et al. [26] with the advantage function of reward implementation defined as below:

$$\begin{aligned} R &= 2 \cdot D_\phi(r, r_{\text{neg}}|q) - 1 \\ &= 2 \cdot [\sigma(g_\phi(q, r) - g_\phi(q, r_{\text{neg}}))] - 1 \end{aligned} \tag{12}$$

Algorithm 1 EnsembleGAN Minimax Game

Require:
Generators G_{θ_1},G_{θ_2}, and discriminator D_ϕ;
Training data $\mathscr{D}_{\text{s2s}}$, $\mathscr{D}_{\text{rank}}$ and retrieval database $\mathscr{D}_{\text{ret}}$;
Three-stage sampling approach R_M as in Eq.(10);
M_1,H the sampling size of G_{θ_1} and G_{θ_2} respectively.
Ensure:
Ensemble of seq2seq model G_{θ_1} and ranker model G_{θ_2}, D_ϕ
Initialize G_{θ_1}, D_ϕ with random weights θ_1,ϕ;
Pretrain G_{θ_1}, D_ϕ on $\mathscr{D}_{\text{s2s}}$, $\mathscr{D}_{\text{rank}}$ respectively
for G_1-steps **do**
  $G_{\theta_1}(\cdot|q)$ generates M_1 samples for each $(q, r) \in \mathscr{D}_{\text{s2s}}$;
  Update G_{θ_1} via policy gradient defined in Eq.(11);
end for
for G_2-steps **do**
  for each $(q, r) \in \mathscr{D}_{\text{rank}}$ **do**
    $G_{\theta_1}(\cdot|q)$ generates M_1 samples
    $G_{\theta_2}(\cdot, r|q)$ generates H samples via R_M;
  end for
  Update G_{θ_2} via policy gradient defined in Eq.(11);
end for
for D-steps **do**
  $G_{\theta_1}(\cdot|q)$ generates M_1 samples for each $(q, r) \in \mathscr{D}_{\text{rank}}$;
  $G_{\theta_2}(\cdot, r|q)$ generates H samples via R_M and combine with positive samples from $\mathscr{D}_{\text{rank}}$;
  Train discriminator D_ϕ according to Eq.(5)
end for

4.2.7 Overall Algorithm. We summarize the ensembleGAN algorithm in Algorithm 1, where all the generators G_1, G_2 and discriminator D are initialized by a pretrained ensemble, with G_2 and D sharing the same parameter initialization.

Despite the very existence of Nash equilibrium between generator and discriminator for their minimax game, it remains an open problem of how they could be trained to achieve the desired convergence [5]. In our empirical study, we confirm that both the ranker D_ϕ and generator G_1 are enhanced by ensembleGAN, while the ranker generator G_2 encounters a loss of performance after adversarial training, as also observed in Wang et al. [26].

5 EXPERIMENTS

In this section, we compare our EnsembleGAN with several representative GAN mechanism on a huge dialogue corpus. The goal of our experiments is to 1) evaluate the performance of our generation module and retrieval module for response generation and selection, and 2) evaluate the effectiveness of our proposed EnsembleGAN framework from the ensemble perspective.

5.1 Dataset

We conduct our experiments on a large mixed dialogue dataset crawled from online Chinese forum Weibo[1] and Toutiao[2] containing millions of query-response pairs. For data pre-processing, we remove trivial responses like "wow" as well as the responses after first 30 ones for topic consistency following Shang et al. [17]. We use Jieba[3], a common Chinese NLP tool, to perform Chinese word

[1]https://www.weibo.com/
[2]https://www.toutiao.com/
[3]https://github.com/fxsjy/jieba

Table 1: The Statistics of Mixed Short-Text Conversation Dataset. Resp. is response for short, # Sent, # Vocab and Avg_L denote the number of sentences, vocabularies and the average sentence length, respectively.

Features \ Dataset		Retrieval Pool	Ranking Set	Generation Set	Test Set
Corpus		Weibo+Toutiao	Weibo	Toutiao	Toutiao
Post	# Sent	2,065,908	30,000	1,000,000	2,000
	# Vocab	251,523	29,272	120,996	5,642
	Avg_L	11.4	13.1	9.3	10.1
Resp.	# Sent	5,230,048	360,000	1,000,000	2,000
	# Vocab	628,254	28,000	121,763	4,544
	Avg_L	8.7	9.8	7.1	7.7
Pair	# Pairs	6,000,000	360,000	1,000,000	2,000

segmentation on all sentences. Each query and reply contain on average 10.2 tokens and 8.44 tokens, respectively. From the remaining query-response pairs, we randomly sampled 6,000,000 pairs as retrieval pool for the pre-retrieval module, 1,000,000 and 50,000 pairs for training and validating the sequence generation model, 360,000 and 2000 pairs for training and validating the ranking model (we apply three-stage sampling strategy to generate 11 negative samples for 30,000 true query-response pairs), and finally 2,000 pairs as test set for both models. We make sure that all test query-response pairs are excluded in training and validation sets. More detailed data statistics are summarized in Table 1.

5.2 Baselines

We introduce baseline models and GAN competitors on three levels, namely the generation approach, the retrieval approach and the ensemble approach. We note GAN-G (D) for the generator (discriminator) of a GAN mechanism in this section. EnsembleGAN is represented by ensGAN for ease of demonstration.

DialogueGAN. We consider dialogueGAN [10] as our GAN competitor for the generation part, with a seq2seq generator and a binary-classifier-based discriminator that is trained to distinguish the true query-response pairs from the fake ones. In order to eliminate structure biases for a fair comparison, we adopt the very same deep matching model structure as our ranking model (which will be detailed later) for its discriminator, instead of the hierarchical recurrent architecture applied by the original paper.

DPGAN. We consider diversity-promoting GAN (DPGAN) [32] as a second GAN competitor for the generation part, with a seq2seq generator and a language-model-based discriminator that is trained to assign higher probability (lower perplexity) for true responses than fake ones. The LM-based discriminator is consisted with a uni-directional LSTM [7] as the original paper.

RankGAN. We consider RankGAN as another GAN competitor. The original RankGAN [11] is an unconditional language model that is unsuitable for dialogue generation scenario as discussed previously, we hence modify RankGAN to consist of a seq2seq generator and a pairwise discriminative ranker, which could be considered as ensGAN without getting generator G_2 involved.

IRGAN. We also consider IRGAN [26] as a GAN competitor. Similarly, this could be considered as ensGAN without any involvement with seq2seq generator or the three-stage sampling strategy. All GAN mechanism are applied on exactly the same pre-trained generation or ranking model for a fair comparison, and we evaluate single component (generator or discriminator) as well as the derived ensemble (Generation + Ranking) for each GAN mechanism, resulting in various combinations which will be detailed later.

Response Generation Models (S2SA). We compare with the attention-based seq2seq model (S2SA) [1], which has been widely adopted as a baseline model in recent studies [10, 17]. As a result, we have three derived adversarial sequence generators, namely the dialogueGAN-G, DPGAN-G, RankGAN-G that compete against ensGAN-G_1. Besides, We include mutual information enhanced seq2seq model (MMI-S2SA) [9] as another generative baseline method.

Pre-Retrieval Module (TF-IDF). The pre-retrieval module, as the basis of retrieval approach, first calculates similarities among utterances (queries) based on simple TF-IDF scores and then retrieve the corresponding responses [18]. We report the Top-{1,2} responses, noted as TF-IDF-{1,2}, respectively.

Response Ranking Models (Ranking). The pure retrieval system is consisted with a pre-retrieval module and a ranking (matching) model, where the pre-retrieved candidates is reranked by the ranker, for which we apply state-of-the-art attentive conv-RNN model [24] for our ranker baseline. Therefore, we have 5 derived adversarial rankers based on the same original ranker, namely the RankGAN-D, IRGAN-G and IRGAN-D that compete against our ensGAN-G_2 and ensGAN-D.

Ensemble Models (Generation+Ranking). Ensemble models are constructed with a generation model, a pre-retrieval module and a ranking model. When given a query, the generative model (e.g., S2SA, RankGAN-G and ensGAN-G_1) synthesizes candidate responses. Then the ranking model (e.g., conv-RNN, IRGAN-D, RankGAN-D and ensGAN-D) is required to rerank both pre-retrieved candidates and synthetic responses, and select the top one response in the end. Besides, following Song et al. [18] and Wu et al. [28], we also consider Multi-Seq2Seq + GBDT reranker and Prototype-Edit as two baseline ensemble models.

5.3 Implementation Details

The seq2seq model is trained with a word embedding size of 300 for source and target vocabulary of 30,000 most frequent tokens of queries and responses in the generation training set, covering 97.47% and 97.22% tokens that appear in queries and responses respectively. The rest tokens are treated as "UNK" as unknown tokens. We set the hidden size of the encoder and decoder to 512. The adversarial sampling size $m_1 = 20$ during G_1 training steps.

The conv-RNN ranker is trained with 200-dimensional word embedding for a shared vocabulary of 40,000 tokens, covering 93.54% words in the retrieval pool. The size of GRU is set to 200. The window size of the convolution kernel is set to (2, 3, 4, 8), with number of filters equal to (250, 200, 200, 150), following Wang et al. [24]. We pretrain the ranker to rank the ground-truth response to the top from $k = 11$ negative samples including 5 random samples, the top 5 pre-retrieved candidates and a synthetic one generated by seq2seq model. During adversarial training, the ranker generator G_2 generates $H = 8$ negative samples from a candidate pool $R_M(100, 10, 10)$ according to the three-stage sampling strategy.

We use dropout of 0.2 for all models, and Adam optimizer [8] with a mini-batch of 50. The learning rate of S2SA and conv-RNN

are respectively 0.0002 and 0.001 during pre-training, 2×10^{-6} and 1×10^{-5} during adversarial learning.

5.4 Evaluation Metrics

We adopt multiple automatic evaluation criteria as well as human evaluation for a comprehensive comparison.

BLEU. BLEU [16] evaluates the word-overlap between the proposed and the ground-truth responses. Typically, we use BLEU_n ($n = 1, 2, 3, 4$) to calculate their n-grams-overlap, where BLEU_n denotes the BLEU score considering n-grams of length n.

Embedding-based metrics (EA, GM, VE). Following Liu et al. [12], we alternatively apply three heuristics to measure the similarity between the proposed and ground-truth response based on pre-trained word embeddings[1], including Embedding Average (EA), Greedy Matching (GM), and Vector Extrema (VE).

Semantic Relevance (RUBER_A, RUBER_G). Together with the embedding similarity, Tao et al. [21] evaluates the semantic relatedness between a response and its query based on neural matching models. Following the original paper, we report the arithmetic and geometric mean of embedding similarity and semantic relatedness, denoted as RUBER_A and RUBER_G, respectively.

Retrieval Precision (P@1). We evaluate pure ranking-based retrieval systems by precision at position 1 (P@1), which calculates the ratio of relevant responses (in our case, the ground-truth response) within top-1 reranked responses.

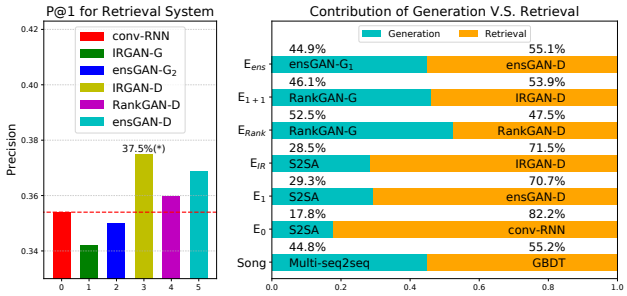
Human evaluation. We also conduct human evaluations for generation and ensemble models since automatic metrics might not be consistent with human annotations [12, 21]. Following previous studies [18, 21, 31], we invited 3 well educated volunteers to judge the quality of 100 randomly generated responses by different models[2], based on the following criteria: a score of 0 indicates a bad response that is either dis-fluent or semantically irrelevant; +1 means a relevant but universal response; +2 indicates a fluent, relevant and informative response. We report the proportion of each score (0, +1, +2) for each model. Fleiss' kappa [4] scores are also reported.

5.5 Results and Analysis

5.5.1 Overall Performance. Our evaluation is divided into three parts, namely the evaluation for pure generation module, pure retrieval module and the ensemble. Table 2 summarizes the general dialogue generation performance including various automatic metrics of word overlap, embedding similarity and semantic relevance. Figure 2 shows the P@1 scores for different retrieval systems as well as the study of contribution of two modules that consist of an ensemble, together with Table 3 of human evaluation results for representative models. The human agreement is validated by the Kappa with a value range of 0.4 to 0.6 indicating "moderate agreement" among annotators. A higher value denotes a higher degree of agreement, such as 0.65 for S2SA which is probably because it generates more dis-fluent or irrelevant responses that are easy to recognize. We first make several observations as follows:

[1]We apply pre-trained Chinese word embedding which is available at https://github.com/Embedding/Chinese-Word-Vectors.

[2]Due to numerous generation + ranking possibilities and space limitations, we only asked annotators to evaluate representative models with high automatic metric scores.



(a) (b)

Figure 2: (a) P@1 scores for various ranker-based retrieval systems. * denotes significant precision improvement (compared with conv-RNN) according to the Wilcoxon signed-rank test; and (b) The final response contribution of generation and ranking modules for ensembles.

1) As for generation module, we first notice that GAN-enhanced seq2seq models achieve plausible improvement on most evaluation metrics, outperforming S2SA and MMI-S2SA baselines. Both RankGAN-G and ensGAN-G_1 aim at synthesizing responses that approximate true responses with higher ranking scores, which is demonstrated by the obvious gain of their contribution ratios for ensembles shown in Figure 2(b), with more than 40% contribution for both RankGAN ensemble (E_{Rank}) and ensGAN ensemble (E_{ens}). Their comparable enhancement to RUBER scores indicates better generations in terms of the semantic relevance. Despite the outperforming word overlap and embedding average of RankGAN-G, ensGAN-G_1 is not only better at improving the GM and VE metrics, indicating more generation of key words with important information that are semantically similar to those in the ground-truth [12], but it's also capable of generating more satisfying responses with fewer 0 human scores according to Table 3.

2) As for retrieval methods, we see that they often achieve advantageous higher order BLEU scores (e.g., BLEU_3 and BLEU_4) than generative approaches, since generating responses of better language fluency (hence higher n-gram overlaps to some extent) is undoubtedly their strong points. They are however inferior to generative methods in terms of RUBER scores, for the latter are generally better at generating more tailored responses of high semantic relatedness [18], similar results are also obtained by Tao et al. [21]. Together with P@1 scores in Figure 2(a), all discriminative rankers of GAN approaches (IRGAN-D, RankGAN-D, ensGAN-D) are generally ameliorated on various aspects, with generative rankers (IRGAN-G, ensGAN-G_2) somehow deteriorated, which is also confirmed by Wang et al. [26]. Similarly, one possible explanation might be the sparsity of the positive response distribution compared with negative ones during training, making it hard for a generative ranker to get positive feedbacks from discriminator. Without any generation module, IRGAN outperforms others on enhancing a pure retrieval system, notably achieving the highest P@1 score. On the other hand however, the P@1 score for all methods remains low compared with common QA task [24, 26], which might be explained by a more complicated and chaotic nature of STC dataset [25].

3) As for the ensembles, they commonly outperform previous single approaches, for the scores in the third block (Ensemble)

Table 2: Overall performance of baselines and GAN competitors. Ranking($\mathcal{M}$) means that candidate responses (generated by the pre-retrieval module) are re-ranked by the ranking model $\mathcal{M}$. Bold scores denote the highest score within each block. The RUBER scores for ground-truth are 0.815, 0.798 for RUBER$_A$ and RUBER$_G$, respectively.

Modules \ Automatic Metrics		Word Overlap				Embedding Similarity			RUBER	
		BLEU$_1$	BLEU$_2$	BLEU$_3$	BLEU$_4$	EA	GM	VE	RUBER$_A$	RUBER$_G$
Generation	S2SA	7.334	2.384	**0.987**	0.340	0.503	0.154	0.332	0.550	0.500
	MMI-S2SA	8.468	2.464	0.956	0.404	0.526	0.149	0.342	0.557	0.521
	DialogGAN-G	9.465	2.483	0.912	0.349	0.533	0.161	0.344	0.560	0.533
	DPGAN-G	8.578	2.474	0.922	0.385	0.535	**0.165**	0.345	0.588	0.557
	RankGAN-G	**10.033**	**2.545**	0.967	**0.436**	**0.560**	0.145	0.343	**0.602**	0.580
	ensGAN-G1	9.530	2.487	0.872	0.352	0.531	0.163	**0.347**	0.598	**0.584**
Retrieval	TF-IDF-1 (pre-retrieval)	7.026	2.175	0.928	0.460	0.537	0.152	0.337	0.541	0.486
	TF-IDF-2 (pre-retrieval)	7.120	2.108	0.990	**0.581**	0.538	0.153	0.338	0.539	0.499
	Ranking (conv-RNN)	7.242	2.213	0.933	0.488	0.543	0.151	0.339	0.558	0.519
	Ranking (RankGAN-D)	7.441	2.194	0.945	0.490	0.547	0.152	0.341	0.571	0.535
	Ranking (IRGAN-G)	7.225	2.166	0.867	0.409	0.540	0.152	0.337	0.560	0.518
	Ranking (IRGAN-D)	7.451	**2.362**	**1.012**	0.528	**0.553**	**0.156**	**0.343**	0.573	**0.542**
	Ranking (ensGAN-G$_2$)	7.057	2.129	0.897	0.460	0.539	0.150	0.338	0.549	0.516
	Ranking (ensGAN-D)	**7.452**	2.320	1.004	0.527	0.548	0.153	0.341	**0.579**	0.539
Ensemble	Multi-Seq2Seq + GBDT [18]	7.542	2.173	0.993	0.569	0.540	0.152	0.338	0.592	0.568
	Prototype-Edit [28]	7.926	2.334	1.120	0.571	0.557	0.164	0.346	0.610	0.587
	S2SA + conv-RNN	7.630	2.299	1.125	0.555	0.544	0.153	0.341	0.564	0.535
	RankGAN-G + conv-RNN	7.755	2.275	0.889	0.432	0.549	0.150	0.340	0.572	0.543
	ensGAN-G$_1$ + conv-RNN	7.570	2.168	0.871	0.423	0.544	0.156	0.339	0.568	0.540
	RankGAN-G + IRGAN-D	8.827	2.693	1.234	0.716	0.560	0.152	0.348	0.608	0.577
	S2SA + IRGAN-D	8.375	2.850	1.232	0.637	0.558	0.162	0.347	0.600	0.573
	S2SA + ensGAN-D	8.535	2.749	**1.297**	0.715	0.547	0.159	0.345	0.595	0.569
	RankGAN-G + RankGAN-D	8.715	2.501	1.075	0.580	**0.561**	0.154	0.347	0.615	0.591
	ensGAN-G$_1$ + ensGAN-D	**9.339**	**2.876**	1.277	**0.763**	0.559	**0.178**	**0.352**	**0.621**	**0.605**

Table 3: Results of human evaluation for generation and ensemble models. "Kappa" means Fleiss' kappa.

Model \ Score	+2	+1	0	Kappa
S2SA	0.12	0.40	0.48	0.65
ensGAN-G$_1$	0.14	0.49	0.36	0.55
RankGAN-G	0.16	0.39	0.45	0.43
S2SA + conv-RNN	0.21	0.33	0.47	0.52
S2SA + IRGAN-D	0.22	0.35	0.43	0.47
S2SA + ensGAN-D	0.25	0.35	0.40	0.49
RankGAN-G + RankGAN-D	0.28	0.35	0.36	0.46
RankGAN-G + IRGAN-D	0.30	0.36	0.35	0.53
ensGAN-G$_1$ + ensGAN-D	0.37	0.38	0.26	0.45

are generally better than the first two blocks (Generation and Retrieval), which is especially true for those GAN-enhanced ensembles. Among various combinations of generation + ranking, the ensGAN ensemble (ensGAN-G$_1$ + ensGAN-D) outperforms both IRGAN (S2SA + IRGAN-D) and RankGAN (RankGAN-G + RankGAN-D) ensembles with the largest gain on almost all metrics, as well as achieving the most +2 and the fewest 0 scores for human judgement. While RankGAN and IRGAN bring specific enhancement to the generative and retrieval module respectively, the ensGAN improves the whole ensemble by allowing each its module to compete against each other, which might be regarded as seeking for a global optimum compared with other GAN that searches for local optimum of a single approach. While the ensGAN-G$_1$ generation module

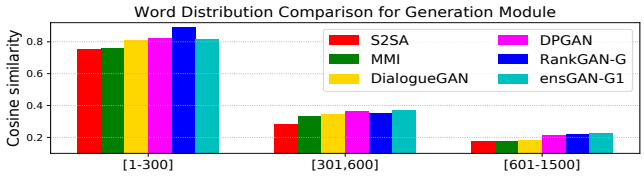
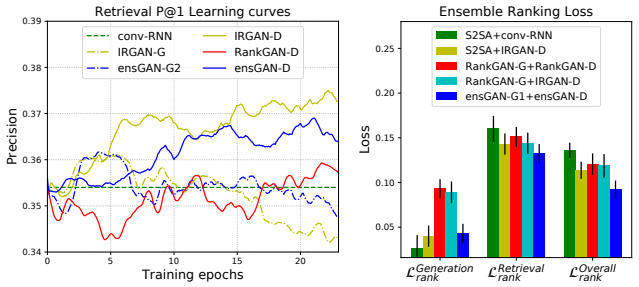


Figure 3: Cosine similarity between ground-truth and synthetic word distribution by various generative models on test data for different word frequency level (e.g., top 300 frequent words). EnsGAN achieves satisfying performance especially when considering words of lower frequency.

accounts more for the ensemble's final selection, the ensGAN-D learns to rank (select) responses featuring advantages of both generative and retrieval approach as expected, with the help of another strong negative sampler G$_2$ during adversarial training.

5.5.2 Discussion. In addition to previous observations, we'd also like to provide further insights of the EnsembleGAN framework on several interesting aspects in this section.

Ranking versus LM versus Binary-Classification. As for the amelioration of generative seq2seq model, while DialogueGAN uses a binary classifier as the discriminator, DPGAN utilizes an LM-based discriminator, and both RankGAN and ensGAN apply ranking-based discriminator. As a result, the superiority of adversarial ranking over binary-classification is not only observed in our experiment, but confirmed in [11] as well. The LM-based discriminator (DPGAN-D) on the other hand, by addressing the saturation



(a) (b)

Figure 4: (a) Ranker P@1 learning curves; and (b) Error bars of mean and standard deviation of ranking loss for different modules of ensembles. Results are calculated on test set.

issue of binary classification [32], brings comparable improvement as adversarial ranking, all of which help generate responses of higher quality as observed previously, as well as achieving better cosine similarity of word distributions (Figure 3). In particular, we apply the adversarial ranking in our work for it's the very bridge that connects the adversarial training of both generative-based and retrieval-based methods in the EnsembleGAN framework.

1+1≠2 for Ensemble Approach. Although it's unreasonable to exhaustively study all possible generation + ranking combinations, it's however interesting to directly combine the seemingly best two modules of their separate worlds, namely the RankGAN-G + IRGAN-D, to see how such an ensemble performs compared with ensGAN. Apart from the overall results in Table 2 which already indicate that these two "best winners" do not get along as well as ensGAN-G_1 + ensGAN-D to some extent, a further evidence lies in the analysis on the ranking module shown in Figure 4. On one hand, the P@1 adversarial learning curves (Figure 4(a)) show that the IRGAN is better at enhancing a pure retrieval system, while RankGAN-D encounters a higher oscillation which is probably due to its concentration on ranking the synthetic responses to the top, making its P@1 pure retrieval performance unpredictable. On the other hand, the ensemble of ensGAN-G_1 + ensGAN-D turns out to be clearly advantageous in terms of the ranking loss (Figure 4(b)) defined in Eq.(2) among ensemble approaches. More specifically, we calculate the module-wise ranking loss for the final chosen responses (considered as r_{neg}) from the generation ($\mathcal{L}_{rank}^{Generation}$) or the pre-retrieval module ($\mathcal{L}_{rank}^{Retrieval}$), the overall ranking loss ($\mathcal{L}_{rank}^{Overall}$) is computed as the weighted sum of the two losses based on the module contribution. We see that ensGAN-D generally achieves the lowest ranking loss with moderate variance, which clearly demonstrates that EnsembleGAN is indeed more inclined towards global optimum without unilaterally enhancing a single module and thus is more adapted for an ensemble of multiple modules, especially when we note that the direct combination of the two "best winners" RankGAN-G + IRGAN-D does not result in the lowest overall ranking loss (not even close).

The Merits of the Ranking Module. In addition, we find that there also exists a clear performance gap among the ensembles themselves. As shown in Table 2, the combinations of original S2SA + GAN-enhanced rankers generally bring better ameliorations compared with the combinations of GAN-enhanced S2SA + original conv-RNN, suggesting the very importance of a re-ranker for a dialogue ensemble, which is reasonable because all candidates have to be reranked by this final decision maker. Hence, despite the trend on the amelioration of generative approaches, it's also plausible to concentrate on the research of retrieval or ensemble methods so as to improve the open domain human-computer conversation.

Table 4: Response generation case study. The final decision of rankers are marked by √. White and gray cells denote valid and inaccessible candidates for a ranker when combined with its corresponding generative module as an ensemble. The original ranker is noted D_{O}, and D_{IR}, D_{R}, D_{E}, D_{T} denote IRGAN-D, RankGAN-D, ensGAN-D and GBDT respectively.

Response generation cases	D_{O}	D_{IR}	D_{R}	D_{E}	D_{T}
q: I can't play sniper.					
r: You're messing up with me.					
TF-IDF-1: Really can't play this.	√	√			
TF-IDF-2: I really can't play games.					√
S2SA: Have you ever played?					
MMI-S2SA: Right, I've been playing for a year.					
DialogueGAN: I've played once, I don't know.					
DPGAN: They're my men, what're you thinking?					
RankGAN-G: You'll know once you've played.			√		
ensGAN-G1: What I played was real.				√	
Multi-Seq2Seq: I've played this game, and you?					
q: Looking for the title of this song.					
r: It's the theme song from Rudy.					
TF-IDF-1: It's Faith.	√	√			
TF-IDF-2: Faith by Xinzhe Zhang.				√	√
S2SA: UNK.					
MMI-S2SA: I'd like to know where you're from.					
DialogueGAN: UNK.					
DPGAN: Yeah!					
RankGAN-G: UNK by UNK.			√		
ensGAN-G1: Thanks for your love.					
Multi-Seq2Seq: UNK.					
q: You're pretty, with or without fringe.					
r: Well, I've got a big face actually.					
TF-IDF-1: Hahahaha.		√			
TF-IDF-2: UNK.	√				
S2SA: Thanks.					
MMI-S2SA: Haha, thanks.					
DialogueGAN: I think so.					
DPGAN: I'm UNK.					
RankGAN-G: Haha, I'm also UNK.			√		
ensGAN-G1: Thank you my friend.				√	
Multi-Seq2Seq: Haha.					√
q: Who's called hot pepper?					
r: It's a girl we saw previously.					
TF-IDF-1: Our favourite fast food.	√				√
TF-IDF-2: You don't even know hot pepper?		√		√	
S2SA: I know you're male.					
MMI-S2SA: I know you're male, I'm female.					
DialogueGAN: Let me guess who you are.					
DPGAN: I mean that I'm actually her.					
RankGAN-G: Are you talking about your joke?			√		
ensGAN-G1: I knew it was you!					
Multi-Seq2Seq: Yeah, me too.					

5.5.3 Case Study. Table 4 shows several example response generation by ensembles, together with various baselines. It's obvious that an ensemble becomes plausible for selecting one final response from multiple candidates in case a single approach fails to respond correctly, just as the second and third case, corresponding to generative-failure and retrieval-failure respectively. We could also observe that as for generation module, most enhanced seq2seq models are better than S2SA in terms of both language fluency and

informativeness. Moreover, the GAN-enhanced seq2seq models are generally better than MMI-S2SA which generates irrelevant responses like "I know you're male" given the query "who's called hot pepper?" in the last case. Among GAN-based generators, all DPGAN, RankGAN and ensGAN achieve similar performances in terms of the generation enrichness, which seem slightly better than dialogueGAN. Besides, while the original GBDT ranker and IRGAN-D mostly prefer the retrieved candidates, RankGAN-D however largely favors synthetic responses, conforming with their respective GAN initiatives. In contrast, the ensGAN-D is able to perform more balanced and logical selections between its generation module and pre-retrieval module, demonstrating its ability to leverage both advantages of single retrieval-based and generation-based approach in dialogue generation scenarios.

6 CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel generative adversarial framework that aims at enhancing a conversation retrieval-generation ensemble model by unifying GAN mechanism for both generative and retrieval approaches. The ensembleGAN enables the two generators to generate responses getting higher scores from the discriminative ranker, while the discriminator scores down adversarial samples and selects responses featuring merits of both generators, allowing for both generation and retrieval-based methods to be mutually enhanced. Experimental results on a large STC dataset demonstrate that our ensembleGAN outperforms other GAN mechanism on both human and automatic evaluation metrics and is capable of bringing better global optimal results.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their constructive comments. This work was supported by the National Key Research and Development Program of China (No. 2017YFC0804001), the National Science Foundation of China (NSFC Nos. 61672058 and 61876196).

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *ICLR*.
[2] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A Survey on Dialogue Systems: Recent Advances and New Frontiers. *SIGKDD Explorations* 19, 2 (2017), 25–35.
[3] Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. 2017. Towards Diverse and Natural Image Descriptions via a Conditional GAN. In *ICCV*. 2989–2998.
[4] Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin* 76, 5 (1971), 378–382.
[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *NIPS*. 2672–2680.
[6] R Herbrich. 2008. Large margin rank boundaries for ordinal regression. *Advances in Large Margin Classifiers* 88 (2008).
[7] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.
[8] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.
[9] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A Diversity-Promoting Objective Function for Neural Conversation Models. In *NAACL*. 110–119.
[10] Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial Learning for Neural Dialogue Generation. In *EMNLP*. 2157–2169.
[11] Kevin Lin, Dianqi Li, Xiaodong He, Ming-Ting Sun, and Zhengyou Zhang. 2017. Adversarial Ranking for Language Generation. In *NIPS*. 3158–3168.
[12] Chia-Wei Liu, Ryan Lowe, Iulian Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation. In *EMNLP*. 2122–2132.
[13] Linqing Liu, Yao Lu, Min Yang, Qiang Qu, Jia Zhu, and Hongyan Li. 2018. Generative Adversarial Network for Abstractive Text Summarization. In *AAAI*.
[14] Mehdi Mirza and Simon Osindero. 2014. Conditional Generative Adversarial Nets. *CoRR* abs/1411.1784 (2014).
[15] Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016. Sequence to Backward and Forward Sequences: A Content-Introducing Approach to Generative Short-Text Conversation. In *COLING*. 3349–3358.
[16] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*. 311–318.
[17] Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural Responding Machine for Short-Text Conversation. In *ACL*. 1577–1586.
[18] Yiping Song, Cheng-Te Li, Ming Zhang, Dongyan Zhao, and Rui Yan. 2018. An Ensemble of Retrieval-Based and Generation-Based Human-Computer Conversation Systems. In *IJCAI*. 4382–4388.
[19] Richard S. Sutton, David A. McAllester, Satinder P. Singh, and Yishay Mansour. 1999. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *NIPS*. 1057–1063.
[20] Chongyang Tao, Shen Gao, Mingyue Shang, Wei Wu, Dongyan Zhao, and Rui Yan. 2018. Get the Point of My Utterance! Learning Towards Effective Responses with Multi-head Attention Mechanism. In *IJCAI*. 4418–4424.
[21] Chongyang Tao, Lili Mou, Dongyan Zhao, and Rui Yan. 2018. RUBER: An Unsupervised Method for Automatic Evaluation of Open-Domain Dialog Systems. In *AAAI*. 722–729.
[22] Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. Multi-Representation Fusion Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. In *WSDM*. 267–275.
[23] Ashwin K. Vijayakumar, Michael Cogswell, Ramprasaath R. Selvaraju, Qing Sun, Stefan Lee, David J. Crandall, and Dhruv Batra. 2018. Diverse Beam Search for Improved Description of Complex Scenes. In *AAAI*. 7371–7379.
[24] Chenglong Wang, Feijun Jiang, and Hongxia Yang. 2017. A Hybrid Framework for Text Modeling with Convolutional RNN. In *SIGKDD*. 2061–2069.
[25] Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. 2013. A Dataset for Research on Short-Text Conversations. In *EMNLP*. 935–945.
[26] Jun Wang, Lantao Yu, Weinan Zhang, Yu Gong, Yinghui Xu, Benyou Wang, Peng Zhang, and Dell Zhang. 2017. IRGAN: A Minimax Game for Unifying Generative and Discriminative Information Retrieval Models. In *SIGIR*. 515–524.
[27] J. Weston, E. Dinan, and A. H. Miller. 2018. Retrieve and Refine: Improved Sequence Generation Models For Dialogue. *CoRR* abs/1808.04776 (2018).
[28] Yu Wu, Furu Wei, Shaohan Huang, Zhoujun Li, and Ming Zhou. 2018. Response Generation by Context-aware Prototype Editing. *CoRR* abs/1806.07042 (2018).
[29] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *ACL*. 496–505.
[30] Yu Wu, Wei Wu, Dejian Yang, Can Xu, and Zhoujun Li. 2018. Neural Response Generation With Dynamic Vocabularies. In *AAAI*. 5594–5601.
[31] Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. Topic Aware Neural Response Generation. In *AAAI*. 3351–3357.
[32] Jingjing Xu, Xu Sun, Xuancheng Ren, Junyang Lin, Bingzhen Wei, and Wei Li. 2018. DP-GAN: Diversity-Promoting Generative Adversarial Network for Generating Informative and Diversified Text. In *EMNLP*. 3940–3949.
[33] Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System. In *SIGIR*. 55–64.
[34] Rui Yan and Dongyan Zhao. 2018. Coupled context modeling for deep chit-chat: towards conversations between human and computer. In *SIGKDD*. 2574–2583.
[35] Rui Yan, Dongyan Zhao, and Weinan E. 2017. Joint learning of response ranking and next utterance suggestion in human-computer conversation system. In *SIGIR*. 685–694.
[36] Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Improving Neural Machine Translation with Conditional Sequence Generative Adversarial Nets. In *NAACL*.
[37] Lili Yao, Yaoyuan Zhang, Yansong Feng, Dongyan Zhao, and Rui Yan. 2017. Towards Implicit Content-Introducing for Generative Short-Text Conversation Systems. In *EMNLP*. 2190–2199.
[38] T. Young, E. Cambria, I. Chaturvedi, M. Huang, H. Zhou, and S. Biswas. 2018. Augmenting End-to-End Dialogue Systems with Commonsense Knowledge. In *AAAI*. 4970–4977.
[39] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. In *AAAI*. 2852–2858.
[40] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In *ACL*. 654–664.
[41] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. 2017. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *ICCV*, Vol. 2223–2232.