A. Wasserstein Distance

As shown in the equation (9) in the regular paper, the Wasserstein distance is formulated as:

$$W(P_r, P_\theta) = \inf_{\gamma \in \prod(P_r, P_\theta)} \mathbb{E}_{(x,y)\sim\gamma} \left[\|x - y\|\right] \tag{21}$$

where $\prod(P_r, P_\theta)$ is the set of all joint distributions $\gamma(x, y)$ whose marginals are respectively P_r and P_θ. Intuitively, $\gamma(x, y)$ indicates how much "mass" must be transported from x to y in order to transform the distributions P_r into the distribution P_θ. The Wasserstein distance then is the "cost" of the optimal transport plan, which is also called the Earth-Mover distance. The Wasserstein distance is proven to be much weaker than many other common distances (*e.g.* JS distance) so simple sequences of probability distributions are more likely to converge under this distance (Arjovsky et al., 2017). In this paper, we prove that our proposed objective function is equivalent to minimizing the Wasserstein distance between the synthetic data distribution and the real data distribution.

B. Configuration Details

We implement our models based on the OpenNMT framework[3] (Klein et al., 2017).

B.1. Dialog Generation

The sentences are tokenized by splitting on spaces and punctuation. We set the LSTM hidden unit size to 500 and set the number of layers of LSTMs to 2 in both the encoder and the decoder. Optimization is performed using stochastic gradient descent, with an initial learning rate of 1.0. The learning rate starts decaying at the step 15000 with a decay rate of 0.95 for every 5000 steps. During training, we iteratively update the noise δ, forward network and backward network respectively. The mini-batch size for the update is set at 64. We set the dropout (Srivastava et al., 2014) ratio as 0.3. For the Transformer, The dimension of input and output is 512, and the inner-layer has the dimension of 2048. we employ 8 parallel attention layers (or heads). We set the dropout ratio as 0.1, batch size as 64. We use the Adam optimizer (Kingma & Ba, 2014) with $\beta_1 = 0.9, \beta_2 = 0.998$ and $\epsilon = 10^{-9}$. We use a maximum dialog history length of 3 to prune the irrelevant context information. The λ in the latent noise sampling (LNS) is set as 0.1.

B.2. Neural Machine Translation

When training our NMT systems, we split the data into subword units using BPE (Sennrich et al., 2016). We train 4-layer LSTMs of 1024 units with bidirectional encoder, the embedding dimension is 1024. The model is trained with stochastic gradient descent with a learning rate that began at 1. We train for 340K steps; after 170K steps, we start halving the learning rate every 17K step. During training, we iteratively update the noise δ, forward network and backward network for 5k steps, 35 steps, 10k steps respectively. Our batch size is set as 128, and the dropout rate is 0.2. We adopt the default setting of the base Transformer for NMT in the OpenNMT framework. The λ in the latent noise sampling (LNS) is set as 0.1.

[3] https://github.com/OpenNMT/OpenNMT-py

C. Error Analysis

In Table 7, we show two examples for a representative problem of leveraging the mutual information in dialog generation. In these cases, both the MMI and AMI generate sentences about the hobbies or preferences, which have strong connection with the word "fun" in the dialog history of the first example, and "favorite" in the dialog history of the second example but not suitable for the given contexts. This indicates that maximizing the mutual information between the source and target may sometimes overly focuses on only a part of the sequences, thus ignores the whole semantic meanings and results in improper responses. In contrast, although the original Transformer produces a common sentence "I do it for a living" in the first case, and an unrelated sentence "Are you a hunter, or have you ever been" in the second case, they are the better choices compared to another two candidates in the table.

[A]: I teach a kindergarten class in Seattle. [B]: That must be fun, but also a lot. **Transformer:** I do it for a living. **Transformer + MMI:** Do you watch game of thrones? **Transformer + AMI:** Okay. Do you have any hobbies? **Human:** It is difficult but it is my passion. Passion has trouble paying the bills though.
[A]: What are you going shopping for? [B]: Clothes. I am a tall guy and kind of big so shopping is not my favorite. **Transformer:** Are you a hunter, or have you ever been? **Transformer + MMI:** I listen to much music. Do you play the guitar? **Transformer + AMI:** I do not like to clean up after my husband. **Human:** Do not like shopping much either, cause i am broke. I will be successful soon though!

Table 7. Two Examples of dialog generation in PersonaChat dataset. "Human" denotes the original response in the dataset, and all the generation models represent the role [A].