Identifying Untrustworthy Samples: Data Filtering for Open-domain Dialogues with Bayesian Optimization

Lei Shen\textsuperscript{1,2}, Haolan Zhan\textsuperscript{2}, Xin Shen\textsuperscript{3}, Hongshen Chen\textsuperscript{4}, Xiaofang Zhao\textsuperscript{1,4}, Xiaodan Zhu\textsuperscript{5}

\textsuperscript{1}Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
\textsuperscript{2}University of Chinese Academy of Sciences, Beijing, China
\textsuperscript{3}Australian National University, Australia
\textsuperscript{4}JD.com, Beijing, China
\textsuperscript{5}Queen’s University, Canada

\texttt{shenlei17z@ict.ac.cn, zhanhaolan316@gmail.com, u6498962@anu.edu.au, ac@chenhongshen.com, zhaoxf@ict.ac.cn}

\textbf{ABSTRACT}

Being able to reply with a related, fluent, and informative response is an indispensable requirement for building high-quality conversational agents. In order to generate better responses, some approaches have been proposed, such as feeding extra information by collecting large-scale datasets with human annotations, designing neural conversational models (NCMs) with complex architecture and loss functions, or filtering out untrustworthy samples based on a dialogue attribute, e.g., Relatedness or Genericness. In this paper, we follow the third research branch and present a data filtering method for open-domain dialogues, which identifies untrustworthy samples from training data with a quality measure that linearly combines seven dialogue attributes. The attribute weights are obtained via Bayesian Optimization (BayesOpt) that aims to optimize an objective function for dialogue generation iteratively on the validation set. Then we score training samples with the quality measure, sort them in descending order, and filter out those at the bottom. Furthermore, to accelerate the “filter-train-evaluate” iterations involved in BayesOpt on large-scale datasets, we propose a training framework that integrates maximum likelihood estimation (MLE) and negative training method (NEG). The training method updates parameters of a trained NCMs on two small sets with newly maintained and removed samples, respectively. Specifically, MLE is applied to maximize the log-likelihood of newly maintained samples, while NEG is used to minimize the log-likelihood of newly removed ones. Experimental results on two datasets show that our method can effectively identify untrustworthy samples, and NCMs trained on the filtered datasets achieve better performance.

\textbf{CCS CONCEPTS}

- Computing methodologies → Discourse, dialogue and pragmatics.

\textbf{KEYWORDS}

Dialogue systems, Data filtering, Bayesian optimization

\textsuperscript{*}Corresponding author.

Table 1: Examples of untrustworthy samples from two open-domain dialogue datasets: OpenSubtitles (Case 1 and 2) and DailyDialog (Case 3). “c” and “r” represent context and response, respectively. “→” denotes the concatenation of consecutive utterances.

| Case | Context | Response |
|------|---------|----------|
| 1    | c       | A big nail should be put in your head. |
|      | r       | Who are they? (An unrelated response) |
| 2    | c       | We met yesterday. → Oh, you’re Thomas. → Yes. |
|      | r       | We haven’t met. (An inconsistent response) |
| 3    | c\textsubscript{1} | There! You can see a window there. |
|      | c\textsubscript{2} | It’s about an hour. |
|      | r       | You can buy a bus schedule in a news stand. |
|      | c\textsubscript{3} | I see. Thank you. (A generic response) |

\textbf{ACM Reference Format:}

Lei Shen, Haolan Zhan, Xin Shen, Hongshen Chen, Xiaofang Zhao, Xiaodan Zhu. 2021. Identifying Untrustworthy Samples: Data Filtering for Open-domain Dialogues with Bayesian Optimization. In Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM ’21), November 1–5, 2021, Virtual Event, QLD, Australia. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3459637.3482352

\section{INTRODUCTION}

With the availability of large-scale data, such as posts on social media or forums, scripts of movies or TV series, and datasets from collection or crowdsourcing [8, 16, 23], neural conversational models (NCMs) [44, 45] have developed rapidly and their performance has been improved steadily. Current NCMs are mainly trained on context-response pairs\textsuperscript{*} with maximum likelihood estimation (MLE). However, the generated responses may suffer from some notorious problems, such as being generic, inconsistent, or unrelated to the given contexts. Previous studies tried to solve these issues by feeding extra information, e.g., topics [25], sentence types [36], personas [19], emotions [43, 60], documents [30], multi-modal [46] or knowledge [17, 27, 55, 56, 61], augmenting the model itself [54, 59], or modifying the loss function [18].

In addition to the above mentioned approaches, some researchers pay attention to data filtering methods that remove untrustworthy training samples and further improve the performance of open-domain dialogue generation. Take Table 1 as an example. The response in Case 1 is unrelated to the given context, while the response in Case 2 is inconsistent to the previous utterances. In
Case 3, the response can be used to respond to three different contexts, and is considered as a generic response. When it comes to the question “how to identify untrustworthy samples?”, each work firstly defines a measure based on an individual dialogue attribute, including Coherence [54], Genericness [10], Relatedness and Connectivity [1]. Then samples with low (or high) scores are regarded as untrustworthy ones. However, due to the subjectivity and open-ended nature of human conversations, the quality of dialogue data varies greatly [7]. Moreover, it is hard to evaluate the dialogue quality just grounding on a single metric [28, 29, 32]. In general, a reasonable assessment should contain multiple aspects of attributes [38], such as the fluency and specificity of the response, the topical relatedness and factual consistency between the context and response, etc.

In this paper, we propose a data filtering method to identify untrustworthy samples in training data with the consideration of seven dialogue attributes. Instead of using these attributes in isolation, we gather them together to learn a measure of data quality based on Bayesian Optimization (BayesOpt) [6]. Specifically, the measure is a combination of seven attributes and the weights are learnt via BayesOpt. BayesOpt aims to iteratively find a group of weights that optimizes an objective function on the validation set for dialogue generation. After obtaining the optimal weights and calculating the measure score for each sample, we can filter out data with low scores and use the maintained (filtered) dataset to train an NCM. Nevertheless, simply applying BayesOpt for data filtering on large-scale dialogue datasets is inappropriate, since an iteration of “filter-train-evaluate” based on lots of samples is time-consuming. To accelerate the process, we design a training framework, named Diff-MLE-NEG, which combines maximum likelihood estimation (MLE) and negative training method (NEG) to accelerate the optimization iterations performed on large-scale datasets.

2 DATA FILTERING METHOD

In this section, we introduce our proposed data filtering method for open-domain dialogues in detail. We firstly give a description of the task, and then illustrate the Bayesian Optimization approach used for data filtering as well as the definitions of two important components, i.e., dialogue attributes and the objective function. Finally, we design a training method that combines maximum likelihood optimization (MLE) and negative training method (NEG) to accelerate the optimization iterations performed on large-scale datasets.

2.1 Task Definition

Given a dialogue corpus $X$ with $N$ pairs of context $c$ and response $r$, i.e., $X = \{(c_i, r_i)\}_{i=1}^N$, data filtering aims to remove untrustworthy utterance pairs from the training data, and further improve the performance of NCMs for dialogue generation [1].

To score each sample $(c, r)$ pair, we propose a measure $S$ as a linear combination of several dialogue attributes:

$$S = w^T \varphi(X),$$

where $\varphi(X) \in \mathbb{R}^{nx1}$ is the dialogue attributes further described in Section 2.3 for each training sample, $n$ is the number of attributes, and $w \in \mathbb{R}^{nx1}$ denotes the weights we want to obtain for these attributes. After each sample is scored according to the measure $S$, we consider pairs with low scores as untrustworthy samples and filter them out for the dialogue generation task. We define the objective function $J$ as the automatic evaluation metric for dialogue generation performed on the validation set. Then we convert the dialogue data filtering task into an optimization problem, and our goal is to define $\varphi(X)$ and find the weights $w$ that optimizes $J$.

2.2 Bayesian Optimization for Data Filtering

The defined measure $S$ is agnostic of the objective function $J$, thus we cannot apply gradient-based methods for optimization. Other search methods such as random search or grid search requires exponential traverses regarding the number of parameterization of $w$, which is time-consuming and not suitable for our task.

Inspired by Tsvetkov et al. [50] and Ruder and Plank [37], we use Bayesian Optimization [6] here, which is a framework used to globally optimize any black-box function [41]. Bayesian Optimization can be considered as a sequential approach to performing a regression from high-level model parameters (e.g., hidden state dimension, the number of layers in a neural network, or $w$ in our method) to the loss function or the objective function [50].

In general, given a black-box function $f : \mathbb{X} \rightarrow \mathbb{R}$, Bayesian Optimization tries to find an input $x^* = \arg \min_{x \in \mathbb{X}} f(x)$ that globally minimize $f$. To fulfill this, it requires a prior $p(f)$ over the function and an acquisition function $a_p(f) : \mathbb{X} \rightarrow \mathbb{R}$ that calculates the utility of any evaluation at any $x$. Bayesian Optimization then proceeds in an iterative manner. At iteration step $t$, (1) it obtains the most promising input $x_t \in \arg \max_{x \in \mathbb{X}} a_p(f)(x)$ via numerical optimization; (2) then, it evaluates the surrogate function $y_t = f(x_t) + N(0, \sigma^2)$ on input $x_t$, and adds the resulting data point $(x_t, y_t)$ to the set.
where $\varphi(NIDF)$ to measure word rareness:

$$\varphi(NIDF) = \frac{\text{IDF}(w) - \text{idf}_{\text{min}}}{\text{idf}_{\text{max}} - \text{idf}_{\text{min}}},$$

where $\text{IDF}(w) = \log \frac{N_r}{N_w}$, $N_r$ is the number of responses in the dataset, and $N_w$ is the number of responses that contain word $w$. $\text{idf}_{\text{min}}$ and $\text{idf}_{\text{max}}$ are the minimum and maximum IDFs, taken over all words in the vocabulary. Then, the specificity of a response $r$, $\text{Spec}(r)$, is the mean NIDF of all words in $r$.

Repetitiveness. Specificity cares more about inter-utterance difference, for intra-utterance diversity, we use Rept($r$) [7] to measure the repetitiveness of a response $r$:

$$\text{Rept}(r) = \frac{\sum_{i=1}^{n} I(w_i \in \{w_0, ..., w_{i-1}\})}{|r|},$$

where $I(\cdot)$ is an indicator function that takes the value 1 when $w_i \in \{w_0, ..., w_{i-1}\}$ is true and 0 otherwise. $|r|$ is the length of $r$.

Relatedness. Cosine similarity between the vectors of two utterances, $c^6$ and $r$, has been widely used to capture their topical relatedness [1, 54, 57]:

$$\text{Rel}(c, r) = \cos(v(c), v(r)),$$

where $v(\cdot)$ is the utterance embedding, and is computed as the weighted sum of word embeddings: $v(s) = \frac{1}{|s|} \sum_{w \in s} \text{NIDF}(w) e(w)$ [2], where $e(w)$ and $p(w)$ are the embedding and probability$^7$ of word $w$, respectively.

Continuity. A good open-domain conversational agent should be able to interact with users in more turns, that is, a response is also responsible to encourage the next utterance. Following Cai et al. [7], Cont($r, u$) is introduced as the cosine similarity between vectors of $r$ and the subsequent utterance $u$ with the same calculation as Relatedness.

Coherence. Coherence evaluates whether a response can be considered as an appropriate and natural reply to the context, and it is also described as Connectivity [1]. Coherence between $c$ and $r$ can be either measured as next sentence prediction [49] or conditional language modeling task [32]. With the support of large-scale pre-trained language models, e.g., GPT-2 [35], the latter one could be more reliable to capture the coherence between $c$ and $r$ after we further fine-tune GPT-2 on dialogue datasets. Following Pang et al. [32], Coh($c, r$) is defined as follows (ranging from 0 to 1):

$$\text{Coh}(c, r) = \frac{\text{max}(C_5, P_G(r|c) - C_5)}{C_5},$$

where $P_G$ represents the fine-tuned GPT-2 model, and $P_G(r|c) = \frac{1}{|c|} \sum_{t=1}^{n} \log P_G(r_t|r_{<t}, c)$. Since $P_G(r|c)$ is a negative unbounded number, a lower bound $C_5$ used to normalize Coherence is defined as 5th percentile of the score distribution. Please refer to Pang et al. [32] for more details.

Fluency. Fluency assesses the grammatical correctness and readability of an utterance. Similar to Coherence, the fluency score is computed as:

$$\text{Flu}(r) = \frac{\text{max}(F_5, P_G(r) - F_5)}{F_5},$$

where $P_G(r) = \frac{1}{|r|} \sum_{t=1}^{n} \log P_G(r_t|r_{<t})$, and $F_5$ denotes 5th percentile of the score distribution.

**Consistency.** Factual consistency is also an important attribute for a dialogue, and there should not be logic contradictions. Natural Language Inference (NLI) [5] tries to map a sentence pair to an entailment category including "Entailment", "Neutral", and "Contradiction" (abbreviated as "Contra" below). Besides, it has been used in dialogue systems to measure the logic consistency [32, 48, 53].

---

$^6$For contexts with multiple utterances, we concatenate them into a long string.

$^7$Probability is calculated based on the maximum likelihood estimation on the training data.
We apply the pre-trained model RoBERTa\(^6\) while the undesirable samples are defined as newly removed ones. As we can see in Figure 1, in each iteration of Bayesian Optimization, we leave applying these metrics as \(J\) are computed by Bayesian Optimization based on likelihood estimation (MLE) and negative training method (NEG) \(^{15}\). For the latter one, please refer to Cai et al. \(^{7}\) for more details. For simplicity, we denote these two types of \(J\) as “+ppl” and “-metric”, respectively. Some automatic metrics \(\{28, 29, 32, 47, 49, 58\}\) that correlate more with human judgment have been proposed recently, even to sort training samples properly.

### 2.4 Training Acceleration

As we can see in Figure 1, in each iteration of Bayesian Optimization, we need to repeat filtering, training and evaluation. Since the maintained dataset \(\hat{M}_t\) is still large, training an NCM with it is extremely time-consuming. To accelerate the process, we design a training method, Diff-MLE-NEG, which combines maximum likelihood estimation (MLE) and negative training method (NEG) \(^{15}\) together.

Following He and Glass \(^{15}\), MLE is used to maximize the log-likelihood of desirable samples, while NEG aims to minimize the log-likelihood of undesirable ones. Here the desirable samples are defined as newly maintained ones in iteration \(t\), i.e., \(\hat{M}_t = M_t - M_0\), while the undesirable samples are defined as newly removed ones in iteration \(t\), i.e., \(\hat{R}_t = R_t - R_0\), which is illustrated in Figure 2\(^{10}\). \(M_0\) and \(R_0\) are obtained by randomly removing bottom \(n\%\) samples in \(X\). An NCM is trained to convergence with \(M_0\) at first, and then Diff-MLE-NEG (shown in Algorithm 1) updates the model parameter \(\theta\) on \(\{M_t\}_{t=1}^k\) and \(\{R_t\}_{t=1}^k\) iteratively, where \(k\) is the number of optimization iterations.

![Figure 2: Illustration of the training acceleration process.](Image)

**Figure 2:** Illustration of the training acceleration process. \(M_0\) and \(R_0\) are obtained by randomly removing bottom \(n\%\) samples in \(X\). An NCM is trained on \(M_0\) to convergence, and evaluated on the validation set to get \(J_0\). The maintained sample sets \(\{M_t\}_{t=1}^k\) and removed sample sets \(\{R_t\}_{t=1}^k\) are the outputs of “score-sort-filter” process, where the weights \(w_t\) in measure score \(S\) are computed by Bayesian Optimization based on \(J_{t-1}\). The newly maintained sample sets \(\{\hat{M}_t\}_{t=1}^k\) are the difference sets of \(\{M_t\}_{t=1}^k\) and \(M_0\), respectively. The newly removed sample sets \(\{\hat{R}_t\}_{t=1}^k\) are the difference sets of \(\{R_t\}_{t=1}^k\) and \(R_0\), respectively. \(k\) is the iteration number of Bayesian Optimization.

We aim at answering the following research questions:

\[\text{(RQ1): What is the performance of our approach on automatic and human evaluations? Does our model outperform other comparable methods?}\]

\[\text{(RQ2): Is our method able to identify untrustworthy samples, and even to sort training samples properly?}\]

\[\text{(RQ3): Does our data filtering method improve the generation quality and generate better response?}\]

\[\text{(RQ4): What are the relations among chosen attributes? Are they reasonable to be included in measure } S?\]

---

\(^{6}\)We use RoBERTa released by Huggingface: https://huggingface.co/roberta-large-mnli.

\(^{9}\)Bayesian Optimization needs to minimize \(J\), but most metrics we used for evaluating dialogue generation should be larger when indicating a better response.

\(^{10}\)We use the difference set of \(M_t\) (or \(R_t\)) and \(M_0\) (or \(R_0\)), rather than \(M_t\) (or \(R_t\)) and \(M_{t-1}\) (or \(R_{t-1}\)) (the difference set of two consecutive iterations), to achieve a stable update process.
We employ a 2-layer bidirectional GRU \[9\] as the encoder and a
also compare our method with an entropy-based filtering ap-
What is the difference between using only one attribute and our
Table 2: Statistics of two experimental datasets, OpenSubti-
(size of latent variable for GVT equals to 300. The word embedding is
unidirectional one as the decoder for S2S and CVAE. The hidden
experiments, we apply IDENTITY-BOTH training data for promoting less-safe response generation. In our
proach \[10\], which aims to remove generic utterances from the
• S2S \[3\]: a sequence-to-sequence model with cross-attention
• CVAE \[59\]: a conditional variational auto-encoder model
• TRS \[51\]: the Transformer model, which is an encoder-decoder
• GVT \[22\]: a Transformer model with a global latent variable.
We also compare our method with an entropy-based filtering ap-
approach \[10\], which aims to remove generic utterances from the
training data for promoting less-safe response generation. In our
experiments, we apply IDENTITY-BOTH\[12\] method of Csáky et al.
and denote it as “Ent” to show the results.

3.4 Implementation Details
We employ a 2-layer bidirectional GRU \[9\] as the encoder and a
unidirectional one as the decoder for S2S and CVAE. The hidden
size is set to 300, and the size of latent variable is set to 64 for CVAE.
For TRS and GVT, the hidden size, number of attention heads and
number of hidden layers are set to 300, 2, and 2, respectively. The
size of latent variable for GVT equals to 300. The word embedding is
initialized with the 300-dimensional pre-trained GloVe embeddings
\[34\] for both encoder and decoder. KL annealing and the BOW loss are applied as in Zhao et al. \[59\]. \(k\), the number of Bayesian Optimization iteration, is set to 100. The proportion (\(\%\)) of training samples we need to filter out is 26% and 12% for OpenSubtitles and DailyDialog, respectively.\[13\] In the test time, we use greedy decoding strategy for all models. Each model is trained on two kinds of datasets: the original dataset and the filtered dataset obtained from our data filtering method, keeping other configurations the same. To restrict the NEG training method and avoid the loss being too large, we utilize gradient penalty following Gulrajani et al. \[14\]. All the models are implemented with Pytorch\[14\] and are trained on four Titan Xp GPUs.

3.5 Evaluation Measures
We use both automatic metrics and human judgement for evaluation in our experiments.

Automatic Metrics. We adopt several automatic metrics in existing work to measure the performance of dialogue generation models, including: (1) BLEU \[33\] measures how much a generated response containing n-gram overlaps with the ground-truth response\[15\]. (2) Perplexity \[39\] measures the high-level general quality of the generation model, and usually a relatively lower Perplexity value indicates a more fluent response. (3) Distinct \[18\] measures the degree of diversity. Specifically, we leverage Dist-1/2/3 \[13\] to compute the average of distinct values within each generated response. (4) Embedding-based metrics \[24,40,54\] contain three metrics to compute the word embedding similarity between the generated response and the ground truth \[24\]: 1. Greedy: greedily matching words in two utterances based on the cosine similarities between their embeddings; 2. Average: cosine similarity between the averaged word embeddings in the two utterances; 3. Extrema: cosine similarity between the largest extreme values among the word embeddings in the two utterances\[16\]. Inspired by Xu et al. \[54\], we use Coherence to calculate the cosine distance of two semantic vectors of a context and the generated response\[17\]. We use GloVe vectors pre-trained on Twitter as the word embeddings \[34\].

Human Evaluation. A human evaluation is also conducted to validate the effectiveness of our proposed method. Firstly, we randomly sample 100 contexts from the test set and get the generated responses of NCs trained on either filtered dataset or original dataset. Next, we send pairs of the context and generated response to three professional annotators without orders. Annotators are then required to evaluate among “Win” (response \(r_1\) is better), “Loss” (response \(r_2\) is better) and “Tie” (they are equally good or bad) independently, considering four aspects: Relatedness, Consistency, Fluency and Informativeness. Relatedness evaluates whether the generated responses are relevant on topic with its contexts; Consistency measures whether the generated responses contain factual contradictions with respect to its contexts; Fluency assesses the

\(\text{RQ5}: \text{Where does the improvements of our method come from? What is the difference between using only one attribute and our proposed measure } S?\)

\(\text{RQ6}: \text{How does Bayesian Optimization work in this method?}\)
grammatical correctness and readability of the generated responses; Informativeness indicates whether the generated responses are informative and not generic. The result on each sample is determined by majority voting. Finally, we calculate the percentage of samples where an NCM trained on filtered or original dataset generates the better response and where an NCM performs similarly on filtered and original datasets.

### 4 EXPERIMENTAL RESULTS

In this section, we demonstrate our experimental results on two datasets: OpenSubtitles and DailyDialog, including automatic evaluation results, human evaluation results, and case study on data filtering and dialogue generation.

#### 4.1 Automatic Evaluation Results (RQ1)

We list the results of automatic evaluation on both the OpenSubtitles and DailyDialog datasets in Table 3 and 4, respectively. From the results, we have three main observations.

First, compared with training on original datasets, our data filtering method can not only bring improvements for all the four NCMs on almost all the evaluation metrics (t-test, p-value < 0.05), but also achieve competitive performance across two datasets. This also affirms the generalization ability of our proposed method.

Second, when it comes to the comparison between our method and a comparable data filtering approach (“Ent”) [10], our method can also achieve better performance. The work of Csáky et al. [10] aims to solve the “general response” problem and generate more
Table 5: Results of human evaluation on two datasets: (1) OpenSubtitles and (2) DailyDialog. Tests are conducted pairwise between the responses generated by NCMs that are trained on filtered and original datasets. Agreements to measure the consistency among three annotators are calculated with the Fleiss’ kappa [11].

| Filtered vs. Original | Win | Loss | Tie | Kappa |
|-----------------------|-----|------|-----|-------|
| S2S                   | 43% | 15%  | 42% | 0.514 |
| CVAE                  | 48% | 7%   | 45% | 0.493 |
| TRS                   | 52% | 8%   | 40% | 0.596 |
| GVT                   | 47% | 10%  | 43% | 0.565 |

Table 6: Case study on Data Filtering on two datasets: (1) OpenSubtitles and (2) DailyDialog. “S” is the value calculated by Equation 1 to score each training sample.

| Context | Response | S   |
|---------|----------|-----|
| 1       | Get on with it. | -1.2063 |
| 2       | Does he roll? → Does he talk in his sleep? → I don’t know. | 0.0010 |
| 3       | Well, what do you want me to do? | 0.0748 |
| 4       | What? → The favorite flower for your bouquet. → Oh, well! | 1.6667 |
| 5       | You can get it from the student business office. | -0.6359 |
| 6       | That’s all right. I understand. | -0.4400 |
| 7       | I’m feeling hungry. | 0.7346 |
| 8       | How many rooms are there? | 0.9264 |

Table 7: Case Study on Dialogue Generation with TRS on two datasets: (1) OpenSubtitles and (2) DailyDialog. “-O” and “-F” denote responses generated by a TRS model that is trained on the original datasets and filtered datasets, respectively.

| Context | Response-O | Response-F |
|---------|------------|------------|
| 1       | Where were you? → On the toilet. | I’m not going to get it. | I was just trying to get you. |
| 2       | She’s in shock! Okay? | She’s a good idea. | I’m sorry about that. |
| 3       | They’re changing the guard over. | They’re not going to be a good idea. | Then they’re not going to be here. |
| 4       | I totally need my board. I gotta go find it. | I’m not going to get it. | OK, I’ll see you later. |
| 5       | So, Paula, where are you from? | I’m in the middle of the city. | I’m from Beijing. |
| 6       | Hey man, you wanna buy some weed? | I’m sorry. I don’t know what to do. | I’m sorry. I’m going to buy a new car. |
| 7       | I have a terrible headache. | I’m sorry, Sir. | I’m sorry to hear that. |
| 8       | Excuse me, may I help you? | Yes, I’d like to buy a new one. | Yes, I’d like to book a book. |

diverse responses in open-domain conversations. Therefore, the higher Diversity scores, i.e., Dist-1/2/3 and Intra-1/2/3, further reflect the effectiveness of our proposed method.

Third, we also notice that the absolute improvement of Dist-1/2/3 value on OpenSubtitles is large (e.g., up to 0.12%/0.62%/1.32% based on the TRS model), but for DailyDialog, the improvement is relatively small. We conjecture that OpenSubtitles benefits more from the data filtering method than DailyDialog, as DailyDialog is manually-collected and many samples of it are in high quality, while OpenSubtitles is more complex and contains a larger amount of low-quality samples.

4.2 Human Evaluation Results (RQ1)
We also conduct pairwise human evaluation to confirm the improvement of our method, and the results on two datasets are shown in Table 5.

We observe that training on filtered datasets outperforms training on original datasets for all the four NCMs as the percentage of “Win” is much larger than that of “Loss”. The kappa scores indicate that the annotators came to a moderate agreement in the judgment. Meanwhile, as the DailyDialog dataset contains less low-quality training samples, the percentage of “Tie” is also high, showing that training on original or filtered dataset has similar performance. The human evaluation here is based on the overall sample quality, and we leave the fine-grained comparisons for the future work.

4.3 Case Study on Data Filtering (RQ2)
To better understand whether our method can identify untrustworthy samples, and even sort training samples based on data quality, we conduct some case studies on both OpenSubtitles and DailyDialog datasets.

As shown in Table 6, we can see that cases with repetitive words, unrelated contents, or generic expressions, named as untrustworthy samples in our work, are scored with low S scores. In contrast, informative and related cases are likely to have high S scores. The results here are correlated with human judgement, which indicates the
We conduct some further analyses to validate the effectiveness of correlations of any two attributes inspired by Cai et al. [7]. Table 8 illustrates that these attributes, in general, do not show strong correlations with each other. However, “I’m not going to get it” is irrelevant to the context, especially with a strange “it” that confuses people a lot.

### 4.4 Case Study on Dialogue Generation (RQ3)

The ultimate goal of our work is to improve the open-domain dialogue generation with data filtering method. To further check the performance of NCMs trained on filtered datasets, we show some responses generated by the TRS model on both OpenSubtitles and DailyDialog datasets in Table 7.

“Response-O” and “Response-F” denote responses generated by a TRS model that is trained on original datasets and filtered datasets, respectively. In general, responses under “Response-F” are more informative and related to the corresponding contexts. Take Case 1 as an example. The context contains a question “Where were you?”, and the response is “I was just trying to get you”. The past tenses and contents in these two utterances make them highly coherent with each other. However, “I’m not going to get it” is irrelevant to the context, especially with a strange “it” that confuses people a lot.

## 5 FURTHER ANALYSIS

### 5.1 Correlation among Attributes (RQ4)

Seven dialogue attributes are combined together when we define \( \varphi(X) \) in Section 2.3. To show their relations, we calculate the Kendall \( \tau \) correlations of any two attributes inspired by Cai et al. [7]. Table 8 illustrates the results on OpenSubtitles.

We see that the maximum absolute value of correlations is 0.451 between Specificity and Fluency, and other correlation values are around 0.1. This demonstrates that these attributes, in general, do not show strong correlations with each other. At the same time, it partially validates that dialogue quality is reflected in multiple facets. Besides, the negative correlation of Specificity and Fluency is also consistent with people’s cognition, as a generic response has a higher probability to be fluent.

### 5.2 Single vs. Multiple Attributes (RQ5)

To gain some insights into the effects of seven dialogue attributes on our proposed filtering method, we conduct the ablation study using TRS by only exploiting a single attribute as the measure to score each training sample. Table 9 reports the results on OpenSubtitles.

We observe that data filtering can lead to better performance, even with only one attribute. When applying the measure \( S \) that combines seven attributes, the performance is the best on most of automatic metrics, including Perplexity, Diversity metrics, and Embedding-based metrics representing fluency, diversity, and relatedness, respectively.

### 5.3 \( J \)-value Curves of Optimization (RQ6)

To get some intuitive feelings of the optimization procedure, Figure 3 shows the “metric” value, the sum of 13 open-domain automatic metrics, for model TRS on OpenSubtitles. The red line denotes the hypothesis space exploration of Bayesian Optimization for the “metric” value, while the blue line displays the overall best value on the validation set.

We find that Bayesian Optimization here results in a large explored space (more variance), and the “metric” value keeps increasing as the iteration continues. This plot can not only illustrate the optimization produce, but also implicitly validate the effectiveness of DIFF-MLE-NEG, as Bayesian Optimization can work normally.

## 6 RELATED WORK

Our work is related to two research branches: Data Filtering for Noisy Corpora, and Metrics for Automatic Dialogue Evaluation.

### 6.1 Data Filtering for Noisy Corpora.

To better utilize noisy dialogue corpora, model- and data-based approaches have been proposed. The former one use models to handle noise during the training process. Shang et al. [42] integrated a calibration network into a generation network. The calibration network is trained to measure the quality of the context-response pairs, and the generation network takes the scores produced by the calibration network to weight the training samples, such that the high-quality samples have more impacts on the generation model while the low-quality ones are less influential. The latter one aims to improve the data quality by pre-processing before feeding it into a model. Xu et al. [54] introduced Coherence with cosine similarity to measure the semantic relationship between contexts and responses, and then filtered out pairs with low coherence score. Csáky et al. [10] used entropy to find out the most generic contexts or responses, and then filtered out pairs with high entropy. Akama et al. [1] defined Connectivity (C) and Relatedness (R) with

Bayesian Optimization aims to minimize “metric” value, while here we plot the increase of “metric” value. 

---

**Table 8: Kendall \( \tau \) correlations among dialogue attributes on OpenSubtitles.** This table illustrates that these attributes, in general, do not show strong correlations with each other.

| Opponent                | Kendall | Opponent            | Kendall | Opponent                | Kendall |
|-------------------------|---------|---------------------|---------|-------------------------|---------|
| Specificity vs. Repetitiveness | 0.039   | Specificity vs. Relatedness | -0.044  | Specificity vs. Continuity | -0.057  |
| Specificity vs. Coherence   | -0.333  | Specificity vs. Fluency   | -0.451  | Specificity vs. Consistency | 0.005   |
| Repetitiveness vs. Relatedness | 0.011   | Repetitiveness vs. Continuity | 0.011   | Repetitiveness vs. Coherence | -0.274  |
| Repetitiveness vs. Fluency   | 0.011   | Repetitiveness vs. Consistency | 0.025   | Relatedness vs. Continuity | 0.102   |
| Relatedness vs. Coherence   | 0.104   | Relatedness vs. Fluency   | 0.005   | Relatedness vs. Consistency | 0.090   |
| Continuity vs. Coherence   | 0.047   | Continuity vs. Fluency   | 0.027   | Continuity vs. Consistency | 0.035   |
| Coherence vs. Fluency      | 0.281   | Coherence vs. Consistency | -0.002  | Fluency vs. Consistency   | 0.009   |
Table 9: Ablation Study on OpenSubtitles with TRS. "None" and "S" denote training on original dataset and filtered dataset with our proposed method, respectively. Others lines represent training on the dataset filtered by a single dialogue attribute. Average, Greedy, Extrema, and Coherence are abbreviated as Avg., Gre., Ext., and Coh., respectively. This table shows that data filtering can lead to better performance, even with only one attribute. When applying the measure $S$ that combines seven attributes, the performance is the best on most of automatic metrics.

| Filtered by | BLEU | Perplexity | Dist-1 | Dist-2 | Dist-3 | Intra-1 | Intra-2 | Intra-3 | Avg. | Gre. | Ext. | Coh. |
|-------------|------|------------|--------|--------|--------|---------|---------|---------|------|------|------|------|
| None        | 0.46 | 48.98      | 0.05   | 0.16   | 0.27   | 99.03   | 99.58   | 99.72   | 87.57 | 73.81 | 52.67 | 91.07 |
| Specificity | 0.42 | 51.65      | 0.05   | 0.15   | 0.27   | 98.33   | 99.31   | 98.91   | 87.97 | 73.84 | 50.98 | 91.50 |
| Repetitiveness | 0.61 | 47.24      | 0.09   | 0.34   | 0.63   | 99.26   | 99.76   | 99.86   | 86.00 | 72.30 | 50.47 | 90.28 |
| Relatedness | 0.55 | 47.48      | 0.08   | 0.27   | 0.51   | 92.15   | 94.81   | 97.16   | 86.01 | 72.41 | 51.21 | 90.37 |
| Continuity  | 0.83 | 47.12      | 0.09   | 0.36   | 0.69   | 97.80   | 98.55   | 98.90   | 87.34 | 74.15 | 53.31 | 91.34 |
| Coherence   | 0.65 | 47.07      | 0.11   | 0.46   | 0.93   | 97.79   | 98.91   | 99.35   | 86.58 | 73.29 | 51.92 | 91.37 |
| Fluency     | 0.68 | 47.16      | 0.12   | 0.51   | 1.08   | 98.16   | 98.81   | 99.01   | 85.34 | 73.01 | 52.97 | 89.70 |
| Consistency | 0.71 | 47.18      | 0.11   | 0.47   | 0.93   | 97.77   | 98.69   | 98.92   | 86.33 | 72.79 | 50.90 | 91.20 |

$S$ | 0.80 | 46.06 | 0.17 | 0.78 | 1.70 | 98.57 | 99.61 | 99.76 | 86.00 | 74.40 | 53.57 | 91.55 |

Figure 3: $J$-value curves of Bayesian Optimization on the validation set. The red line denotes the hypothesis space exploration of Bayesian Optimization for the “metric” value, while the blue line displays the overall best value on the validation set. We find that Bayesian Optimization results in a large explored space (more variance), and the “metric” value keeps increasing as the iteration continues.

normalized pointwise mutual information [4] and cosine similarity to remove noisy pairs with low C+R score. Li et al. [20] proposed an iteration-based way to distill data, which operates as: a neural sequence-to-sequence (S2S) model is first trained and used to generate responses to inputs in a dataset. Then a list of the most common responses is constructed, and training samples with outputs that are semantically close to these common responses are removed. As the process iterates, responses that are generic are gradually distilled, and the trained models gradually increase in specificity.

The differences between our method and the above listed ones are: (1) The proposed measure $S$ does not rely on only one or two attributes, it integrates seven dialogue attributes to estimate the quality of dialogue samples. (2) We frame the dialogue data filtering task as an optimization problem, and use Bayesian Optimization to find the optimal weights in the linear combination of $S$.

6.2 Metrics for Automatic Dialogue Evaluation.

The evaluation of open-domain dialogue generation generally consists of both automatic metrics and human judgement. Since human evaluation has high variance, high cost, and is difficult to replicate, some researchers aim to propose automatic metrics that correlate more with human intuition for this task. Dialogue quality is inherently multi-faceted [7, 38, 52], including fluency, coherence, diversity, consistency, etc. Inspired by the automatic evaluation conducted in other fields, referenced automatic metrics are applied [12, 33, 49]. Recently, unreferenced metrics are also proposed to assess different dialogue attributes, such as Distinct [18], Entropy [32], MAUDE [47], USR [29], and FED [28]. The last three ones also rely on pre-trained language models, and have higher correlation with human judgement. However, compared with some simple metrics, these new metrics are not widely used and thoroughly accepted by this community.

Our work does not aim to propose a new automatic metric for open-domain dialogue generation, we only utilize some of them to help define the measure $S$ or objective function $J$. In addition to the options we choose in Section 2.3, other metrics like FED can also be used to compute $S$ or $J$, and we leave this for the future work.

7 CONCLUSION AND FUTURE WORK

In this paper, we present a data filtering method for open-domain dialogues, which aims to identify untrustworthy training samples with a quality measure that linearly combines seven dialogue attributes. The attribute weights are obtained via Bayesian Optimization. Besides, to accelerate the optimization iterations on large-scale datasets, we propose a training method that integrates the maximum likelihood estimation (MLE) and negative training method (NEG). Experimental results on two widely-used datasets show that our method can effectively identify untrustworthy samples, and NCMs trained on the filtered datasets can further generate fluent, related, and informative responses. In the future, we will try to utilize some newly proposed automatic metrics for open-domain dialogue generation, and combine them in the definition of the quality measure and the choice of the objective function.
Chongyang Tao, Lili Mou, Dongyan Zhao, and Rui Yan. 2018. RUBER: An
Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones,
Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Brian MacWhinney, and Chris Dyer. 2020. Generating per-
Mingyue Shang, Zhenxin Fu, Nanyun Peng, Yansong Feng, Dongyan Zhao, and
Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Fre-
Koustuv Sinha, Prasanna Parthasarathi, Jasmine Wang, Ryan Lowe, William L.
Lei Shen, Haolan Zhan, Hainan Zhang, Dongyan Zhao, and
Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle
Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you
Xinnao Xu, Ondřej Dušek, Ioannis Konstas, and Verena Rieser. 2018. Better Con-
Tianyu Zhao, Divesh Lala, and Tatsuya Kawahara. 2020. Designing Precise and
Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiuqin Li, Chris Brockett, and
Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. WMT.
Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning Discourse-level
Tianyu Zhao, Divesh Lala, and Tatsuya Kawahara. 2020. Designing Precise and
Haoyu Song, Wei-Nan Zhang, Jingwen Hu, and Ting Liu. 2020. Generating per-
Yanyan Lan. 2021. Augmenting knowledge-grounded conversations with se-
Haolan Zhan, Hainan Zhang, Dongyan Zhao, and
Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiuqin Li, Chris Brockett, and
Bill Dolan. 2018. Generating Informative and Diverse Conversational Responses
Yanyan Lan. 2021. Augmenting knowledge-grounded conversations with se-
Koustuv Sinha, Prasanna Parthasarathi, Jasmine Wang, Ryan Lowe, William L.
Lei Shen, Haolan Zhan, Xin Shen, and Yang Feng. 2021. Learning to select context in a hierarchical and global perspective for open-domain dialogue generation. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE. 7438–7442.
Lei Shen, Haolan Zhan, Xin Shen, Yonghao Song, and Xiaofang Zhao. 2021. Text is NOT Enough: Integrating Visual Impressions into Open-domain Dialogue Generation. arXiv preprint arXiv:2109.05778 (2021).
Koustuv Sinha, Prasanna Parthasarathi, Jasmine Wang, Ryan Lowe, William L. Hamilton, and Joelle Pineau. 2020. Learning an Unreferenced Metric for Online Dialogue Evaluation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Online, 2430–2441.
Haoyu Song, Wei-Nan Zhang, Jingwen Hu, and Ting Liu. 2020. Generating persona consistent dialogues by exploiting natural language inference. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34. 8879–8885.
Chongyang Tao, Lili Mou, Dongyan Zhao, and Rui Yan. 2018. RUBER: An Unsupervised Method for Automatic Evaluation of Open-Domain Dialog Systems. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.). AAAI Press, 722–729.
Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Brian MacWhinney, and Chris Dyer. 2016. Learning the Curriculum with Bayesian Optimization for Task-Specific Word Representation Learning. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Berlin, Germany, 130–139.
Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Ilja Polosukhin. 2017. Attention is All you Need. In Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 5998–6008.
Marilyn Walker, Diane Litman, Candace A Kamm, and Alicia Abella. 1997. PARSED: A Framework for Evaluating Spoken Dialogue Agents. In 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics. 271–280.
Sean Welleck, Jason Weston, Arthur Szlam, and Kyunghyun Cho. 2019. Dialogue Natural Language Inference. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Florence, Italy, 3731–3741.
Xinnao Xu, Ondřej Dušek, Ioannis Konstas, and Verena Rieser. 2018. Better Conversations by Modeling, Filtering, and Optimizing for Coherence and Diversity. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Brussels, Belgium, 3981–3991.
Haolan Zhan, Hainan Zhang, Dongyan Zhao, and
Yanyan Lan. 2021. Augmenting knowledge-grounded conversations with sequential knowledge transition. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 5621–5630.
Haolan Zhan, Hainan Zhang, Dongyan Zhao, and
Yanyan Lan. 2021. Augmenting knowledge-grounded conversations with sequential knowledge transition. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 5621–5630.
Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiuqin Li, Chris Brockett, and Bill Dolan. 2018. Generating Informative and Diverse Conversational Responses via Adversarial Information Maximization. In Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada, Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolo Cesa-Bianchi, and Roman Garnett (Eds.). 1815–1825.
Tianyu Zhao, Divesh Lala, and Tatsuya Kawahara. 2020. Designing Precise and Robust Dialogue Response Evaluators. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Online, 26–33.
Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, 654–664.
Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional Chatting Machine: Emotional Conversation Generation with Internal and External Memory. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.). AAAI Press, 730–739.
Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018. Commonsense Knowledge Aware Conversation Generation with Graph Attention. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, (IJCAI-18), July 13-19, 2018, Stockholm, Sweden, Jérôme Lang (Ed.). ijcai.org, 4623–4629.