A Simple Yet Strong Domain-Agnostic De-bias Method for Zero-Shot Sentiment Classification

Yang Zhao[†], Tetsuya Nasukawa[†], Masayasu Muraoka[†], and Bishwaranjan Bhattacharjee[♢]

[†]IBM Research - Tokyo, 19-21 Nihonbashi Hakozaki-cho, Chuo City, Tokyo, 103-8510, Japan,
[♢]IBM Research, Yorktown Heights, New York 10598, USA
yangzhao@ibm.com, {nasukawa,mmuraoka}@jp.ibm.com, bhatta@us.ibm.com

Abstract

Zero-shot prompt-based learning has made much progress in sentiment analysis, and considerable effort has been dedicated to designing high-performing prompt templates. However, two problems exist; First, large language models are often biased to their pre-training data, leading to poor performance in prompt templates that models have rarely seen. Second, in order to adapt to different domains, re-designing prompt templates is usually required, which is time-consuming and inefficient. To remedy both shortcomings, we propose a simple yet strong data construction method to de-bias a given prompt template, yielding a large performance improvement in sentiment analysis tasks across different domains, pre-trained language models, and prompt templates. Also, we demonstrate the advantage of using domain-agnostic generic responses over the in-domain ground-truth data. We release the code here[1].

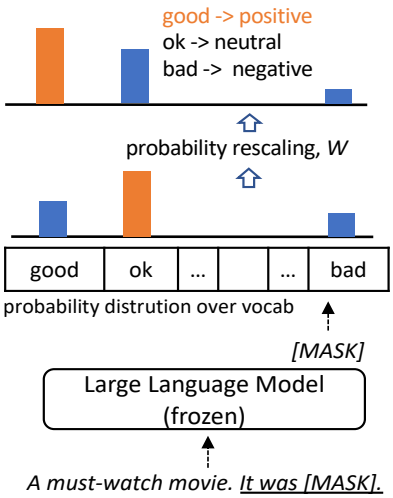

Figure 1: Zero-shot prompt-based sentiment classification for a masked language model. The prompt is *It was [MASK].* and the label token set is {good,ok,bad} which respectively stands for positive/neutral/negative sentiment polarities.

1 Introduction

Over the past few years, zero-shot prompt-based learning has become a de facto standard in many Natural Language Processing (NLP) tasks where training data is unavailable. For sentiment analysis, much effort has also been dedicated to designing effective prompt templates to trigger the capability of Large Language Models (LLMs) such as RoBERTa (Liu et al., 2019) and GPT (Radford et al., 2018) to predict sentiment polarities, e.g., positive or negative. A prompt template typically consists of prompt text and a label token set corresponding to the sentiment class. Gao et al. (2021) demonstrates that *It was {good,ok,bad}.* is a high-performing prompt template for sentiment analysis task. As Figure 1 shows, the input to LLM is *A must-watch movie. It was [MASK].*, and the token *ok* is the most probable word over the label token set.

However, two problems remain. First, an LLM is often biased to its pre-training data, leading to poor performance in prompt templates that the LLM have rarely seen. Second, when it comes to different domains, such as financial and food, re-designing appropriate prompt templates to adapt to new domains is usually required, which is time-consuming and inefficient. To mitigate the first problem, Zhao et al. (2021) proposed a probability rescaling method to calibrate the probability based on the assumption that LLMs should NOT express either positive or negative sentiment when the input is a meaningless sentence such as *N/A*.

Motivated by this, we relax their assumption and further hypothesize that a good LLM should be capable of accurately predicting the sentiment of an "absolute" positive, neutral, or negative sentence. For example, the phrase "thank you so much" mostly manifests an "absolutely" positive

[1] https://github.com/repo4nlp/

Findings of the Association for Computational Linguistics: ACL 2023, pages 3923–3931
July 9-14, 2023

sentiment no matter what the context is[2]. In this spirit, we propose a simple method to construct "absolutely" sentimental instances and use them to learn a probability rescaling layer to de-bias LLM. Particularly, we are interested in the following two research questions; RQ1: *how to construct "absolutely" positive/neutral/negative instances for training a probability rescaling layer to improve sentiment classification performance?* RQ2: *How does the quality of the constructed silver data compare to that of the ground-truth data?*

To answer these questions, we employ generic responses from a dialogue corpus and apply a rule-based sentiment tagger to automatically generate sentiment-labeled instances with various sentiment polarities. Subsequently, the labeled generic responses are utilized to learn a rescaling parameter. Experimental result on seven mainstream sentiment analysis datasets shows that the proposed method outperforms baseline approaches by a large improvement across different domains, pre-trained LLMs, and prompt templates.

Our contributions are two-fold: (1) We propose a simple yet strong data construction method to generate sentiment-labeled instances for sentiment analysis task and human assessment validated the relatively high accuracy of these sentiment-labeled instances; (2) The proposed method obtained large performance improvement in zero-shot sentiment classification task across diverse domains, pre-trained LLMs, and prompt templates. Also, we demonstrated the advantage of constructed sliver data over in-domain ground-truth data.

2 Methodology

2.1 Problem Formulation

Formally, we define an input sentence to be classified sentiment polarity as S, a prompt text as P (e.g., *It was [MASK].* and a label token set as T (e.g., *good,ok,bad*), and a verbalizer Z to map label tokens T into a class label set C (e.g., "good" -> "positive"). LM is a pre-trained LLM that outputs the probability distribution over the vocabulary V. Thus, the sentiment prediction of sentence S is

$$Z(\underset{t \in T}{\operatorname{argmax}} \operatorname{LM}(S|P)), \tag{1}$$

As shown in Figure 1, the LLM predicts the sentence as having a neutral sentiment. To adjust the output probability, a common practice is to rescale the probability distribution over vocabulary of LLMs in the softmax layer. Platt et al. (1999) applies an affine transformation to adjust the probability distribution p, $\hat{p} = \operatorname{softmax}(Wp + b)$, where a weight matrix W and a bias vector b is learnable parameters, while Zhao et al. (2021) follows Guo et al. (2017) to restrict the matrix W to be diagonal and b to be zero to prevent the parameters from growing quadratically in the size of $\hat{p}$; Then, they used meaningless strings such as "N/A" and an empty string to learn diagonal parameter W to shift the probability distribution, as shown in Figure 1. By following the same fashion, namely, our prediction is

$$Z(\underset{t \in T}{\operatorname{argmax}}\, W \cdot \operatorname{LM}(S|P)), \tag{2}$$

where $W \in R^{|V| \times |V|}$ is a diagonal matrix where all elements not on the main diagonal are equal to zero, and |V| is the vocabulary size of an LLM. Different from (Zhao et al., 2021), where they only consider meaningless inputs to learn parameter W, we construct positive, neutral, and negative instances to learn the parameter W. It is worth noting that the LLM's parameters are frozen during training, and only W is updated.

2.2 Proposed Method

We herein describe a simple procedure to construct domain-agnostic instances to learn diagonal parameter W. Central to this construction is to find instances that are not sensitive to the context change. To this end, we take inspiration from generic responses in the dialogue research and argue that generic responses serve as desirable instances for three reasons: (1) insensitive to context change; (2) relatively domain-agnostic; (3) easy to automatically annotate sentiment polarities. Thus, a two-step approach is proposed:

Step 1: Utterance Selection.
We select a dialogue corpus D and extract each utterance to form a set of utterances U. We get frequency f_i of each utterance u_i in U and sort u_i by its frequency f_i in the descent order such that $f_1 \geq f_2 \geq \ldots \geq f_N$. Then, since generic responses feature in high frequency in dialogue corpus, we determine a frequency threshold F and select utterances if its frequency f_i is no less than

[2]In the context of irony, "thank you so much" may not always convey a positive sentiment, However, such instances are very much infrequent in the corpus we are dealing with.

source	Prompt Text	Label Token Set
Gao et al. (2021)	<S> It was [MASK].	{good,ok,bad}
Liu et al. (2021)	<S> The sentiment is [MASK].	{positive,neutral,negative}

Table 1: Widely-used prompts in mainstream sentiment analysis task.

F to form a frequent utterance set S, as a set of generic responses to be sentiment-labeled.

Step 2: Annotation.
We employ a pre-defined positive word list L_{pos} containing 2,006 words and a negative word list L_{neg} containing 4,783 words from (Hu and Liu, 2004) and automatically tag the sentiment polarity t_i of each utterance u_i in S in the following way:

1. if u_i contains more than one positive word in list L_{pos} and no negative word in list L_{neg}: (a) tag t_i "positive" if the number of negation words is an even number, or (b) tag t_i "negative" if the number of negation words is an odd number.
2. if u_i contains no positive word in list L_{pos} and more than one negative word in list L_{neg}: (a) tag t_i "negative" if the number of negation words is an even number, or (b) tag t_i "positive" if the number of negation words is an odd number.
3. if u_i contains no positive word in L_{pos} and no negative word in L_{neg} , then tag t_i "neutral".

We use labeled instances (u_i, t_i) of each class to train the rescaling parameter W.

3 Experiment

3.1 Experimental Details

We use Cornell Movie-Dialog Corpus[3] (D) (Danescu-Niculescu-Mizil and Lee, 2011) which is under creative commons license and extract 304,713 dialogue utterance (U). After deduplication and removal of interrogative sentences[4], we set frequency threshold (F) to 3 to obtain 2,211 sentences. After automatic annotation, we finally yielded 274 positive instances, 176 negative instances, and 1,761 neutral instances. We selected 100 instances for each class and in total, 300 sentiment-labeled instances to train the parameter W. Please refer to the Appendix A for examples.

Quality Assessment of Automatic Annotation
To assess the quality of automatic annotation, we assigned a human annotator[5] to rate 300 instances (100 instances per class) by judging whether the instance is positive or not for 100 automatically annotated positive instances. The same process was followed for the negative and neutral classes. The results show that out of the 300 instances evaluated, 43 instances displayed inconsistencies with human judgment, resulting in an automatic annotation accuracy of 0.86.

For training, we split the data into a training set with 240 instances and dev set with 60 instances and select the best model based on the performance of the dev set. Then, we test the best model on all the datasets in Table 2. For the evaluation metric, we use accuracy for datasets whose label class is balanced (SST-2, IMDB, Yelp, and Amazon) and Macro-F1 for the datasets whose label class is unbalanced (Phrasebank, airline, and debate). We used the arithmetic average (Ave.) instead of the weighted average because we view each dataset and its representing domain equally important[6]. We use one A100 GPU to train the model by setting the batch size to 10, the learning rate to 1e-5, and the number of epochs to 100. It takes half an hour to finish the whole training. The parameters of LLMs are frozen during training.

Dataset	Domain	# of classes	Size
IMDB	Movie	2	1,000
Yelp	Restaurant	2	1,000
Amazon	Product	2	1,000
SST-2	Movie	2	9,613
Airline	Operation	3	10,445
Debate	Politics	3	5,354
Phrasebank	Finance	3	2,264

Table 2: Statistics of sentence-level sentiment datasets. It is worth noting that no training/dev/test split is needed because we use the whole dataset for evaluation only.

[3]https://www.kaggle.com/datasets/rajathmc/cornell-moviedialog-corpus

[4]It is often difficult to determine the sentiment polarity of interrogative sentences such as "is it good?".

[5]The annotator is based in Japan.

[6]We experimented with weighted average according to the size of each dataset in Table 2 and observed even better performance of the proposed method.

Prompt: <S> The sentiment is {positive,neutral,negative}.

Dataset	IMDB	Yelp	Amazon	SST-2	Debate	Airline	Phrasebank	Ave.
Metric	Acc	Acc	Acc	Acc	F1	F1	F1	-
#1 BERT	59.7	58.4	63.3	57.8	20.8	19.4	17.9	42.5
#2 BERT + cali	71.4	69.8	73.7	67.0	32.0	29.9	29.5	53.3
#3 BERT + ours	**80.5**	**76.5**	**81.9**	**72.5**	**37.9**	**41.6**	**30.5**	**60.2**
#4 RoBERTa	**89.0**	87.5	89.0	81.0	40.3	44.2	36.5	66.8
#5 RoBERTa + cali	83.9	81.1	82.7	75.0	32.9	40.5	48.5	63.5
#6 RoBERTa + ours	88.8	**93.8**	**92.5**	**83.3**	**64.6**	**60.8**	**47.4**	**73.1**
#7 GPT2-xl	64.2	59.3	58.5	55.4	16.4	21.1	37.8	44.7
#8 GPT2-xl + cali	85.2	82.0	80.8	72.0	37.2	38.3	**45.7**	63.0
#9 GPT2-xl + ours	**90.6**	**85.2**	**81.2**	**83.0**	**44.1**	**51.2**	35.6	**67.3**

Prompt: <S> It was {good,ok,bad}.

Dataset	IMDB	Yelp	Amazon	SST-2	Debate	Airline	Phrasebank	Ave.
Metric	Acc	Acc	Acc	Acc	F1	F1	F1	-
&1 BERT	62.6	60.6	62.7	56.5	17.7	22.6	16.7	42.8
&2 BERT + cali	80.3	75.7	80.7	68.6	38.9	48.5	**45.3**	62.6
&3 BERT + ours	**83.2**	**83.0**	**85.0**	**75.5**	**50.8**	**55.0**	29.8	**66.0**
&4 RoBERTa	81.7	79.6	82.7	68.2	29.1	49.0	24.6	59.3
&5 RoBERTa + cali	83.9	81.1	82.7	75.0	32.9	40.5	**48.5**	63.5
&6 RoBERTa + ours	**84.4**	**92.2**	**92.2**	**78.7**	**48.3**	**58.1**	41.3	**70.7**
&7 GPT2-xl	78.2	63.6	63.3	65.1	29.7	31.7	19.4	50.1
&8 GPT2-xl + cali	**86.6**	**87.1**	**90.3**	74.1	44.8	26.0	**60.6**	**67.1**
&9 GPT2-xl + ours	84.4	85.8	80.9	**74.5**	**47.9**	**58.7**	33.6	66.5

Table 3: Zero-shot result across three LLMs and two prompt templates. "+ cali" stands for calibration method in (**?**), while "+ ours" stands for our proposed method. Best results are in bold.

3.2 Dataset

Table 2 shows the dataset statistics. We employ publicly available sentiment analysis datasets in different domains; Kotzias et al. (2015) which contain three sentence-level sentiment datasets respectively from IMDB, yelp, and amazon datasets, SST-2 (Socher et al., 2013), Airline[7], Debate[8], and Phrasebank (Malo et al., 2014).

3.3 Prompt Template and Pre-trained LLMs

Although many prompt templates exist, we experimented with two high-performing prompt templates proposed by previous works for sentiment analysis tasks, as shown in Table 1[9]. We experimented with three mainstream LLMs, BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), and GPT2-xl (Radford et al., 2019).

[7]https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment

[8]https://www.kaggle.com/datasets/crowdflower/first-gop-debate-twitter-sentiment

[9]we remove "polarity of <aspect>" in the original template, "The sentiment polarity of <aspect> is [MASK]", for adapting to our non-aspect sentiment classification setting.

4 Result and Analysis

Table 3 presents the result on two prompt templates. We have the following observations: (a) On average, our proposed method (#3, #6, #9, &3, and &6) outperforms other baselines by a large margin on both auto-encoder LLMs (BERT and RoBERTa) and auto-regressive language model (GPT2), validating the effectiveness of the proposed method; (b) Among the three LLMs, the vanilla RoBERTa (#4 and &4) and the RoBERTa enhanced by our method (#6 and &6) achieve the highest performance on both templates. This indicates the effectiveness of data scaling, considering that RoBERTa utilizes 160GB of text data for training, while BERT and GPT2-xl use 16GB and 40GB of text data, respectively.

Interestingly, domain-agnostic generic responses, such as "thank you very much," improve the sentiment classification performances, which touches upon our RQ2: *How does the quality of our domain-agnostic silver data compared to the in-domain ground-truth data from each of seven sentiment analysis datasets?*

Dataset	IMDB	Yelp	Amazon	SST-2	Debate	Airline	Phrasebank	Ave.
Metric	Acc	Acc	Acc	Acc	F1	F1	F1	-
$1 RoBERTa + IMDB (200)	**92.3**	**94.7**	92.1	**85.3**	44.0	52.4	26.5	69.6
$2 RoBERTa + Yelp (200)	84.8	90.9	86.6	79.7	38.0	51.4	10.9	63.2
$3 RoBERTa + Amazon (200)	84.1	90.1	85.1	79.9	37.9	51.9	10.6	62.8
$4 RoBERTa + SST-2 (200)	83.2	90.4	88.3	81.8	45.7	53.3	22.9	66.5
$5 RoBERTa + Debate (300)	87.7	84.2	82.3	82.5	**62.4**	55.7	34.7	69.9
$6 RoBERTa + Airline (300)	92.1	92.0	92.2	**85.3**	38.1	**70.7**	30.1	71.5
$7 RoBERTa + Phrasebank (300)	87.3	86.1	89.3	77.6	7.4	8.9	**83.6**	62.9
$8 RoBERTa + ours (300)	88.8	93.8	**92.5**	83.3	45.3	60.8	47.4	**73.1**

Table 4: Zero-shot Result of leave-one-out experiment. For "+ *Dataset* (N)", N refers to the number of training instances. Best results are in bold.

Domain Analysis

To answer this question, we conducted a leave-one-out experiment as shown in Table 4. For each dataset (domain), we selected 100 ground-truth instances for each label class. Given its superior performance in Table 3, we employ the RoBERTa model in combination with the prompt template, "<S> The sentiment is {positive,neutral,negative}." Here are our observations: (a) Models trained with in-domain data (e.g., IMDB, Debate, Airline, and Phrasebank) perform the best in their own domains; Nevertheless, on average, (b) our method achieves the best average performance, showing the advantage of using domain-agnostic data over domain-specific data to avoid the model to be twisted too much to any specific domains.

5 Related Works

Zero-shot sentiment classification has attracted a lot of attention as it does not require training data, offering advantages in many real-world use cases, particularly in low-resource scenarios. At the core of this task is the construction of an appropriate prompt, which transforms the sentiment classification task into a fill-in-the-blank format. Gao et al. (2021) explored various prompts for sentiment analysis and identified several highly effective prompt templates. Similarly, Liu et al. (2021) experimented with prompt templates in the context of aspect-based sentiment analysis. However, a challenge comes from the sensitivity of accuracy to prompt templates, where even a slight modification in prompt templates can result in a large performance change in zero-shot sentiment analysis tasks.

Zhao et al. (2021) argues that this "instability" arises from the bias of language models towards predicting certain answers that are common in the pre-training data. To address this issue, they estimate the model's bias towards each sentiment polarity class by requesting its prediction when presented with the prompt and a content-free input, such as "N/A." Subsequently, they determine calibration parameters to ensure a uniform probability distribution for this input across all sentiment polarity classes. Later on, Min et al. (2022) borrowed the idea from machine translation research and proposed a noisy channel to shift the probability distribution for few-shot text classification. Our work aligns with the approach proposed by Zhao et al. (2021) but with a notable distinction. Rather than considering the meaningless strings, we go a step further by incorporating positive, neutral and negative sentiment-labeled instances to address the bias in LLMs.

6 Conclusion and Future

In this work, we propose a simple yet effective domain-agnostic data construction method to enhance sentiment classification tasks. Our method was evaluated using three popular LLMs, namely BERT, RoBERTa, and GPT2-xl, along with two commonly employed prompt templates. The results show significant improvements in task performance. Also, we answered the research questions and demonstrated that our constructed domain-agnostic data is superior to in-domain data in terms of overall performance. In the future, we plan to experiment with other dialogue corpora to assess the generalization capabilities of the proposed de-bias method.

Limitations

Although we sidestep the challenge of selecting a specific prompt template for experimentation by opting for widely-used templates from previously

published works, it is worth noting that numerous effective prompt templates are available, and the experimental results obtained using these templates would also provide valuable insights into testing our proposed method. Furthermore, while our method yields improvements, it is important to acknowledge that errors may exist in the rule-based automatic annotation of generic responses, which could potentially propagate to the learning of the diagonal parameter W.

Ethics Statement

While our work primarily centers on mitigating bias in Large Language Models (LLMs) using prompt templates, it also acknowledges the inherent risk shared by BERT, RoBERTa, and GPT models, as they may potentially generate biased language.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable comments. We also thank our colleagues, Issei Yoshida, Hiroshi Kanayama, and Akihiro Nakayama from their helpful suggestions and discussions.

References

Cristian Danescu-Niculescu-Mizil and Lillian Lee. 2011. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics)*, pages 4171–4186, Minneapolis, Minnesota.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 3816–3830.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Dimitrios Kotzias, Misha Denil, Nando De Freitas, and Padhraic Smyth. 2015. From group to individual labels using deep features. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 597–606.

Jian Liu, Zhiyang Teng, Leyang Cui, Hanmeng Liu, and Yue Zhang. 2021. Solving aspect category sentiment analysis as a text generation task. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4406–4416.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65.

Sewon Min, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Noisy channel language model prompting for few-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 5316–5330.

John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

A Appendix

positive instances	negative instances	neutral instances
Thank you very much.	I hate it.	Please, have a seat.
Thank you, ma'am.	I'm sick.	It looks that way.
It's a miracle.	You're wrong.	I know that ...
Brilliant.	You're a liar.	Every day.
I think you're right.	I doubt it.	Nothing personal.
Glad to meet you.	That's your problem.	Keep talking.
I'll do my best.	It won't work.	That was a long time ago.
I'm glad you do.	That's too bad.	Let's do it.
It's nice.	Get lost.	I've seen it.
Good evening.	You don't trust me.	That's ok.

Table 5: Automatic sentiment annotation of example generic responses from a dialogue corpus. We manually checked all 300 instances and found that most of sentences do not contain personal information, while a few contain characters' names in the movie.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
The Limitation section is right after the Conclusion section.

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Our paper's main claims (contribution) are in the last paragraph of the Introduction section.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

This work created a dataset (section 2.2 and section 3.1)

☑ B1. Did you cite the creators of artifacts you used?
section 3.1

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
section 3.1

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
section 3.1

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
please refer to the caption of the appendix A: Most sentences in our dataset do not contain personal information like people's names. However, some sentences contain fake characters' names in the movie.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
abstract section

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
section 3.1

C ☑ Did you run computational experiments?

section 3.1 and section 4.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
section 3.1

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
section 3.1. we select the best model based on the performance on the dev set.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
section 4. For the zero-shot evaluation, everything is deterministic so there are no random factors.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

section 3.1

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Section 3.1

☒ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
no payment

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
From the instruction in section 3.1, the human annotator understands how the data would be used to report the accuracy of machine-produced data.

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Determined exempt. The human annotation is just a binary label 1/0 and this labelled data is not used as training data for any models

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
section 3.1