Sentimatic: Sentiment-guided Automatic Generation of Preference Datasets for Customer Support Dialogue System

Suhyun Lee, Changheon Han
Hanyang University, Seoul, Republic of Korea
{su7561632, datajedi23}@hanyang.ac.kr

Abstract

Supervised Fine-tuning (SFT) and preference optimization (PO) are key methods for enhancing language models and aligning them with human preferences. However, scaling preference datasets for PO training is challenging, leading AI customer support systems to rely on SFT. To address this, we propose the **Senti**ment-guided Auto**matic** Generation of Preference Datasets (**Sentimatic**) methodology to automatically generate customer preference datasets without human intervention using a publicly available dataset constructed for SFT. Our approach classifies responses by sentiment, fine-tunes models on them, and applies advanced sampling and evaluation techniques to ensure diversity and quality. Ultimately, we generated 1,174 customer preference datasets based on 357 test datasets, and through experiments, we confirmed that the AI customer support system trained on these datasets is capable of carefully considering customer emotions and generating professional and appropriate responses.

1 Introduction

Previous studies have used the SFT approach primarily to train AI models for customer service (Xu et al., 2017; Golchha et al., 2019; He et al., 2022). However, SFT focuses solely on the accuracy of individual tokens generated by the model, failing to adequately reflect the overall quality of conversations. This limitation can lead to inefficiencies in performance evaluation and optimization. In contrast, PO addresses these issues by evaluating the quality of the entire response generated by the model (Hua et al., 2024).

However, the preference datasets required for PO training are created through response comparisons, which require the involvement of human annotators. This dependency significantly increases the time and cost of large-scale data collection, posing challenges to the widespread adoption of PO. To address these challenges, AI-based feedback approaches that utilize large language models (LLMs) have been proposed to minimize human intervention (Cui et al., 2024; Bai et al., 2022). However, these approaches still rely on human-authored evaluation criteria for practical application. In the customer service domain, where providing responses that align with customer preferences and mitigate negative emotions is critical, the ambiguity of the evaluation criteria further highlights the limitations of existing methods.

To overcome these challenges, this study proposes a novel methodology for generating customer preference datasets without human intervention. This methodology provides a foundation for the efficient construction and scalability of PO datasets, enabling a wider adoption of PO in AI customer support systems. The proposed methodology consists of the following three key steps:

1. Sentiment Analysis: Model pool is used to analyze emotional changes before and after a response. Responses showing positive emotional changes are considered aligned with customer preferences and included in the positive dataset, while those showing negative emotional changes are included in the negative dataset.

2. Completion Sampling: Positive and negative datasets are used to fine-tune separate models. These models generate pairs of positive and negative responses for the test dataset. To ensure diversity and scalability, N responses are generated for each input by repeating the sampling process.

3. Preference Classification: BERTScore (Zhang et al., 2020) are calculated for the generated response pairs by comparing them with reference responses. High-quality responses are filtered based on a defined threshold.

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop), pages 120–128

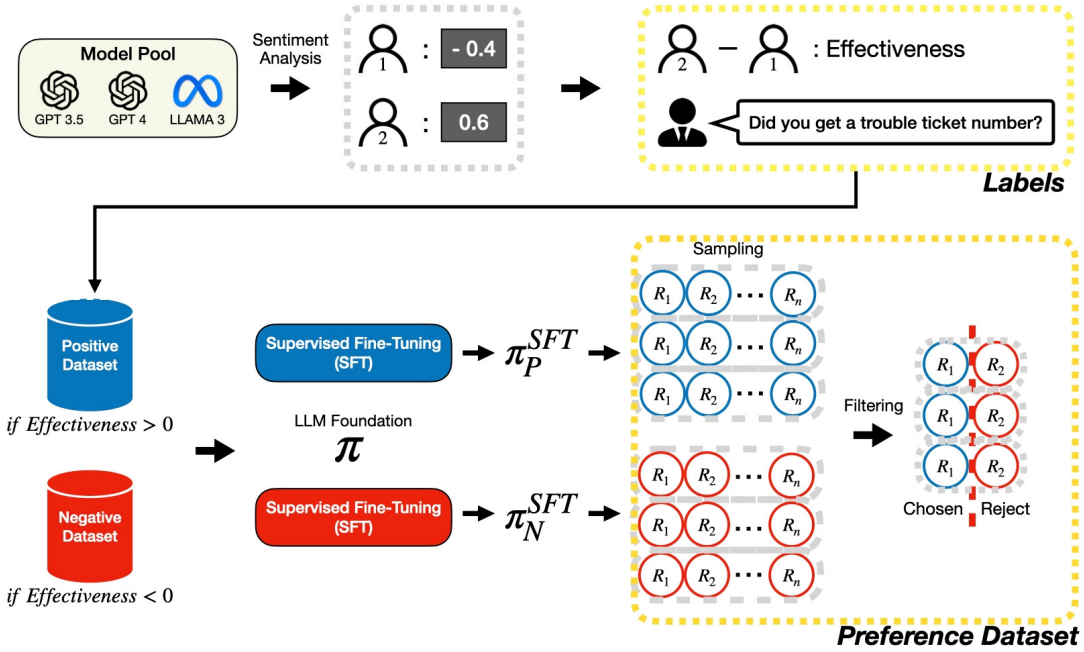

Figure 1: An overview of the Sentimatic methodology. A model pool (GPT-4 (OpenAI et al., 2024), GPT-3.5 (Ouyang et al., 2022), and LLaMA 3 (Grattafiori et al., 2024)) analyzes customer conversations to compute scalar sentiment scores. Responses showing positive emotional shifts are labeled as aligned with customer preferences, while those with negative shifts are not. These labeled datasets are used to fine-tune models for generating aligned and non-aligned responses. Diverse sampling techniques (beam search (Freitag and Al-Onaizan, 2017), top-k (Fan et al., 2018), top-p (Holtzman et al., 2020)) are employed to generate multiple responses per input. BERTScore is then calculated to validate response quality.

2 Methods

2.1 Overview

We adopt an AI-based feedback approach that leverages LLMs with scalability in mind. However, defining "responses aligned with customer preferences" poses a significant challenge. Therefore, instead of following the conventional approach (Cui et al., 2024; Bai et al., 2022) of designing prompts based on human-defined criteria for inference, we opted to fine-tune models separately to learn the responses patterns that align with customer preferences and those that do not.

Specifically, to distinguish between responses aligned with customer preferences and those that are not, we utilized a model pool to obtain sentiment scores for the initial customer conversation and the response, then calculated the difference between them. Responses that demonstrated a positive emotional shift were identified as aligned with customer preferences, while those that showed a negative emotional shift were classified as not aligned with customer preferences, forming the respective datasets.

We then fine-tuned two separate large-language models using the respective datasets. One model was trained on the Positive Dataset to generate responses aligned with customer preferences, while the other was trained on the Negative Dataset to learn patterns of non-aligned responses. Next, we used the fine-tuned models to repeatedly sample responses, generating N responses for the same input to ensure diversity. Finally, we calculate the BERTScore for the generated responses and classify high-quality comparison pairs based on a defined threshold. In the following section, we introduce the Sentimatic methodology in detail.

2.2 Curated Dataset

First, we selected the TWEETSUMM dataset (Feigenblat et al., 2021). This dataset contains real conversations between customer service agents and dissatisfied customers on Twitter, making it suitable for learning linguistic patterns and interaction styles in the customer service domain. Originally, TWEETSUMM is a multi-turn dataset, but we restructured it to focus on initial responses. Conversations were organized based on tweet IDs and transformed into single-turn interactions. Each conversation begins with the initial message from the

Table 1: Dataset Overview. "P" and "N" in the Sentimatic dataset indicate positive and negative preference labels, respectively.

	Sentimatic dataset		TWEETSUMM
Dialog	P	N	Multi-turn
# Train	1,530	1,129	879
# Test	211	146	110
# Valid	192	127	110

customer (c_1), followed by the agent's response, and ends with the customer's reply text after the agent's response (c_2).

Next, we used various models (GPT-4, GPT-3.5, LLaMA3) to perform a sentiment analysis on the customer's initial text (c_1) and the customer's reply text after the agent's response (c_2), assigning the average score as the numerical sentiment score. The prompt used for sentiment analysis can be found in Appendix 6. In this process, if any of the models produced a score of 0, indicating that the model failed to detect positive or negative sentiment tendencies for the given data, the result of that model was excluded. To determine the direction of the change in sentiment, we calculated the difference between the sentiment score of c_1 (s_1) and c_2 (s_2), selecting only responses that showed a positive change (+). Through this process, we collected 1,530 response data points aligned with customer preferences and 1,129 response data points not aligned with customer preferences. A summary of the dataset can be found in Table 1.

2.3 Completion Sampling

LLMs are trained on large-scale datasets to achieve generalization capabilities across various tasks. However, this training approach may not capture the nuances and specific knowledge required in certain domains. Previous studies have shown that fine-tuning in specific domains, such as legal document processing, medical diagnosis, and financial analysis, can lead to significant performance improvements (Dominguez-Olmedo et al., 2025; Ismail et al., 2024; Parker et al., 2022).

Therefore, after fine-tuning the LLMs with a curated dataset, we ensure that the collected responses are diverse and evenly distributed by repeating the various sampling processes (beam-search, top-k, top-p) multiple times to generate N responses for the same input. Specifically, we fine-tune the input-output pairs (x, y) of the selected data set to obtain initial parameters π^{SFT}. Using π^{SFT}, we then generate N responses y_1, y_2,... and y_N:

$$(y_1, \ldots, y_N) \sim \pi^{\text{SFT}}(y|x) \tag{1}$$

The prompt used for fine-tuning can be found in Appendix 7. For inference, only the Instruction and Input parts of the same prompt were used.

Table 2: Quality Evaluation of Completion Sampling (#: Number of samples, C: Chosen average BERTScore, R: Rejected average BERTScore, Δ: Difference average Between Chosen and Rejected Scores)

Sampling	α	β	#	C	R	Δ
Beam Search	0.78	0.2	1174	0.825	0.729	0.096
	0.8	0.2	952	0.834	0.762	0.064
	0.82	0.2	707	0.841	0.730	0.111
Top-K	0.78	0.2	1174	0.825	0.729	0.096
	0.8	0.2	952	0.834	0.762	0.064
	0.82	0.2	707	0.841	0.730	0.111
Top-P	0.78	0.2	1174	0.825	0.729	0.096
	0.8	0.2	952	0.834	0.762	0.064
	0.82	0.2	707	0.841	0.730	0.111

2.4 Quality Evaluation

To ensure contextual relevance and prevent excessive deviation from the dialogue flow, we calculated the BERTScore by comparing the generated responses with the responses from the original data set as references. Specifically, we compute the BERTScore as follows:

$$S_{\text{BERT}}(y, r) = \frac{1}{N} \sum_{i=1}^{N} \cos(\mathbf{h}_i^y, \mathbf{h}_i^r) \tag{2}$$

where y represents the generated response, r is the original reference response from the dataset, and $\mathbf{h}_i^y, \mathbf{h}_i^r$ are the contextual embeddings of each token in y and r, respectively. The final score is obtained by averaging the cosine similarities across all token embeddings.

We use this score to classify high-quality response pairs applying a threshold α, filtering out responses that deviate significantly from the original context. Furthermore, we define a threshold β for the difference in the BERTScore between the chosen and rejected responses to maintain semantic diversity within the dataset. The statistics of the dataset based on α and β are reported in Table 2

3 Experiments

3.1 Response Generation Model

To validate the effectiveness of the Sentimatic methodology, we compare two versions of the Re-

Table 3: Evaluation of different LLM judges on contextual relevance, problem-solving approach, and handling of negative emotions.

Judge	Model	Contextual Relevance	Problem-Solving Approach	Handling Negative Emotions
GPT-4o	T5 + SFT	7.35%	9.45%	15.49%
	T5 + ORPO w/Sentimatic	**67.72%**	**48.29%**	**50.39%**
ChatGPT	T5 + SFT	18.64%	**17.59%**	18.90%
	T5 + ORPO w/Sentimatic	**66.93%**	**62.99%**	**64.30%**
GPT-o3	T5 + SFT	43.83%	46.98%	13.12%
	T5 + ORPO w/Sentimatic	**52.49%**	46.98%	**81.63%**

sponse Generation Model: 1) the SFT version trained on the existing TWEETSUMM dataset based on T5 (Wu et al., 2023) and 2) the Sentimatic version trained with PO using the dataset generated through the proposed methods.

Evaluation Methodology We evaluate the quality of generated responses using the LLM-as-a-judge approach, which follows a win/tie/lose framework judged by multiple LLMs (GPT-4o, ChatGPT, GPT-o3). In particular, we focus on three key aspects that are critical in Customer Support Dialogue Systems: contextual relevance, problem-solving approach, and handling of negative emotions. Each judge compares the responses generated by the two models and selects a preferred one, resulting in the win rate percentages shown in Table 3.

The LLM-as-a-judge methodology has been validated in prior work (Zheng et al., 2023), where strong LLMs such as GPT-4 demonstrated over 80% agreement with human preferences in both controlled and crowdsourced settings. This evaluation framework enables scalable and interpretable estimation of human-like preferences while significantly reducing the cost and effort associated with human evaluation. The used prompt can be found in Appendix 8

Setup We used 1,174 pairs of training data and 319 pairs of validation data, performing 3 fine-tuning iterations. The value of α was set to 0.78 and the value of β was set to 0.2. The ORPO (Hong et al., 2024) method was used as part of the PO approach. For fine-tuning, we utilized the AdamW optimizer with a learning rate of 0.0005 and a linear learning rate scheduler. The batch size per GPU was 8, and the training was performed on a single A6000 GPU.

Result As shown in Table 3, across all three evaluation axes and for all LLM judges, the Sentimatic-enhanced model (T5 + ORPO w/Sentimatic) consistently outperformed the baseline (T5 + SFT). Notably:

- GPT-4o judge: Sentimatic achieved a 67.72% win rate in contextual relevance, 48.29% in problem-solving, and 50.39% in handling negative emotions.
- ChatGPT judge: Sentimatic scored 66.93%, 62.99%, and 64.30% respectively.
- GPT-o3 judge: Sentimatic led with 52.49% for contextual relevance and a striking 81.63% win rate in handling negative emotions.

These results strongly suggest that Sentimatic improves response generation in both contextual understanding and emotional sensitivity, validated by multiple independent LLM judges.

3.2 Qualitative analysis

Table 4 presents representative examples that compare the responses generated by the SFT version and the Sentimatic-enhanced version of the model. We analyze responses along three key axes: contextual relevance, problem solving approach, and handling of negative emotions. Generally,the Sentimatic model demonstrates clear qualitative improvements across all three dimensions, reflecting its effectiveness in producing emotionally intelligent, contextually appropriate, and support-driven responses.

Contextual Relevance The Sentimatic model consistently demonstrates greater contextual awareness in its responses. For example, in row (b-1), the agent acknowledges the user's concern and explicitly refers to the situation ("Can you tell us more about what's going on?"), showing an intention to engage more deeply. In contrast, the SFT version in (a-1) offers a shallow suggestion ("Have you tried restarting your MacBook?") without confirming the actual problem of the user. Additionally, in (b-2), the Sentimatic model tailors the request by asking for specific details, while the SFT version (a-2) provides a generic, less targeted reply. These differences highlight how Sentimatic better aligns its responses with the intent and content of the customer's message.

Table 4: Examples of Model Inference Results

(a) Example 1: SFT version

1.	
Customer:	*@SpotifyCares hey, any explanation why the "Create similar playlist" function doesn't work anymore for me? MacBook, v1.0.64.399.g4637b02a*
Agent:	*@267878 I'm sorry to hear that. Have you tried restarting your MacBook?*
2.	
Customer:	*@AskAmex When is interest charged on Charge Cards with Pay Over Time, Sign and Travel, etc. From the Due Date or from the Closing Date?*
Agent:	*@261778 I'm sorry, I cannot provide an answer to that question as it requires additional information. Please provide the details of the Charge Cards you are interested in.*
3.	
Customer:	*@Delta I just sent it @Delta Thank you, that was excellent resolution, glad I sent that tweet just in time while I still had a window to change the itinerary without a fee.*
Agent:	*@261715 Hi, I'm sorry to hear that. We're glad to hear that you're happy with the resolution. We'll send you a link to the new itinerary.*

(b) Example 2: Sentimatic version

1.	
Customer:	*@SpotifyCares hey, any explanation why the "Create similar playlist" function doesn't work anymore for me? MacBook, v1.0.64.399.g4637b02a.*
Agent:	*@179244 Hi there, thanks for getting back to us. Can you tell us more about what's going on? We're here to help anyway we can.*
2.	
Customer:	*@AppleSupport I did not understood what do you mean by DM my iPhone*
Agent:	*Understood, please provide me with your full name, email address, and postcode. I'll be happy to take a closer look.*
3.	
Customer:	*@airtelcare xxx you, your company, your CEO and even COO. you will fold up and run into loss as u have caused me pains,weeks n counting.*
Agent:	*@179244 Hi there, thanks for getting back to us. We've made a note of this and passed over your feedback to our Developers JB*

Problem-Solving Approach Sentimatic responses generally adopt a more structured and cooperative problem solving approach. For example, (b-2) asks for the customer's full name, email, and postcode in order to assist further—this kind of information gathering is essential in real customer support workflows. However, the SFT response in (a-2) defers the assistance entirely without providing a concrete next step. Similarly, in (b-1), the Sentimatic version attempts to diagnose the problem by requesting more context, whereas the SFT model prematurely offers a potential fix. These examples demonstrate how Sentimatic responses are more proactive and service-oriented in resolving customer issues.

Handling of Negative Emotions Sentimatic shows a significant improvement in emotionally sensitive scenarios. The most striking example is (b-3), where the customer expresses frustration with harsh language. The Sentimatic model remains calm, professional and empathetic, acknowledging the message without escalating the tone, and assuring the user that feedback has been passed to the relevant team. In contrast, the SFT model in similar scenarios, such as (a-3), provides a flat, impersonal response that misses the opportunity to acknowledge the user's sentiment. This suggests that the Sentimatic model is better at defusing negative sentiment and maintaining a respectful tone, even in high-stress conversations.

3.3 AI Feedback Model Specialized for the Customer Support Domain

To develop a scalable method for collecting preference data without relying on public datasets in the customer support domain, we designed an AI feedback model based on LaMini-Flan-T5. This model is configured as a text-to-text task, generating scalar scores representing the quality of responses along with the corresponding textual critiques, allowing a single model to produce both outputs.

The difference between the emotion scores s_1 and s_2, along with c_2, is mapped to a template to generate a scalar score that reflects the change in customer emotion and the expected response.

To validate the effectiveness of the pipeline, two versions of the model were developed. The SFT version was trained using SFT with the mapped text and the initial customer text (y, c_1), while the Sentimatic version was trained using PO on a preference dataset generated through the pipeline. Notably, this process does not aim to create a preference

Table 5: Quality of Text Generation for Customers' Next Response and Score Prediction Error

Model + Method	MSE	BLEU	ROUGE1	ROUGE2	ROUGEL	METEOR
LaMini-Flan-T5-77M + SFT	0.63	32.70	0.46	0.36	0.45	0.49
LaMini-Flan-T5-77M + Sentimatic	**0.55**	28.62	0.52	0.43	0.51	0.49
LaMini-Flan-T5-783M + SFT	0.45	32.92	0.48	0.37	0.46	0.50
LaMini-Flan-T5-783M + Sentimatic	**0.44**	32.48	0.46	0.35	0.44	0.49

dataset itself. Instead of explicitly separating Positive and Negative data, the pipeline expands the dataset using sampling techniques after SFT training. Subsequently, BERTScore is utilized to filter the data, and responses with higher and lower scores are paired to form pairs of 'Chosen' and 'Reject'.

Setup We used 2,852 pairs of training data and performed 30 fine-tuning iterations. The ORPO method was applied as part of the Preference Optimization (PO) approach. For fine-tuning, we employed the AdamW optimizer with a learning rate of 0.0005 and a linear learning rate scheduler. The batch size per GPU was set to 8, and training was performed on a single A6000 GPU.

To evaluate the quality of the Response Generation Model, we used four widely recognized metrics: BLEU(Papineni et al., 2002) , ROUGE(Lin, 2004), and METEOR(Banerjee and Lavie, 2005). Furthermore, the mean squared error (MSE) metric was used to assess the accuracy of the prediction of 'score'.

Result Table 5 presents the performance metrics for different methods in predicting emotion scores and generating customer responses. The LaMini-Flan-T5-77M model, when fine-tuned with the Sentimatic methodology, achieved an MSE of 0.55, indicating a 12.7% improvement compared to the application of SFT alone (MSE 0.63). Similarly, the LaMini-Flan-T5-783M model demonstrated an MSE of 0.44, marking a 2.22% improvement over the SFT-only model (MSE 0.45).

Figure 2 illustrates the distribution of predicted effectiveness scores between models. The SFT-only model shows a high concentration of scores around -0.5, suggesting that the model frequently generates similar emotion scores that deviate from the true values. In contrast, Sentimatic methodology results in a wider distribution of scores, demonstrating the ability to predict a broader range of emotions that align more closely with actual values.

Conclusion

This study proposed a novel methodology for constructing a preference dataset for Preference Optimization (PO) using publicly available customer support data without human intervention. As a result, we generated 1,174 customer preference datasets based on 357 test data instances. The model trained through the proposed data construction pipeline demonstrated effective improvements in the quality of customer support dialogue responses. In particular, we empirically validated that the model can be trained to better meet user expectations without relying on costly human annotations. Across the three key evaluation criteria: contextual relevance, problem solving approach, and handling of negative emotions, the Sentimatic-enhanced model consistently outperformed the baseline model trained by supervised fine-tuning (SFT). These results were reliably validated through the LLM-as-a-judge evaluation framework, involving independent LLM judges including GPT-4o, ChatGPT, and GPT-o3.Overall, the proposed method is scalable, cost-efficient, and readily applicable to real-world customer service scenarios, offering a promising direction for developing emotionally aware and user-centered AI agents.

Limitation

The proposed methodology has certain limitations, depends on multiple LLMs for sentiment detection, which can introduce bias or inaccuracies, and focuses primarily on Twitter-based complaints. To overcome these limitations, future research will evaluate the performance of Sentimatic methodology in general conversation by comparing it with human feedback-based datasets. In addition, ensemble modeling and complementary evaluation techniques will be introduced to minimize bias in large-language models.

References

Yuntao Bai et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *Preprint*, arXiv:2204.05862.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Ganqu Cui et al. 2024. Ultrafeedback: Boosting language models with scaled ai feedback. In *Forty-first International Conference on Machine Learning*.

Ricardo Dominguez-Olmedo, Vedant Nanda, Rediet Abebe, Stefan Bechtold, Christoph Engel, Jens Frankenreiter, Krishna P. Gummadi, Moritz Hardt, and Michael Livermore. 2025. Lawma: The power of specialization for legal annotation. In *The Thirteenth International Conference on Learning Representations*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Guy Feigenblat, Chulaka Gunasekara, Benjamin Sznajder, Sachindra Joshi, David Konopnicki, and Ranit Aharonov. 2021. Tweetsumm–a dialog summarization dataset for customer service. *arXiv preprint arXiv:2111.11894*.

Markus Freitag and Yaser Al-Onaizan. 2017. Beam search strategies for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 56–60, Vancouver. Association for Computational Linguistics.

Hitesh Golchha, Mauajama Firdaus, Asif Ekbal, and Pushpak Bhattacharyya. 2019. Courteously yours: Inducing courteous behavior in customer care responses using reinforced pointer generator network. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 851–860.

Aaron Grattafiori et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Keqing He, Jingang Wang, Chaobo Sun, and Wei Wu. 2022. Unified knowledge prompt pre-training for customer service dialogues. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4009–4013.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. *Preprint*, arXiv:1904.09751.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model. *Preprint*, arXiv:2403.07691.

Ermo Hua, Biqing Qi, Kaiyan Zhang, Yue Yu, Ning Ding, Xingtai Lv, Kai Tian, and Bowen Zhou. 2024. Intuitive fine-tuning: Towards simplifying alignment into a single process. *arXiv preprint arXiv:2405.11870*.

Amelia Ritahani Ismail, Amira Shazleen Aminuddin, Afiqa Nurul, Noor Azura Zakaria, and Wafa Haussain Nasser Fadaaq. 2024. A fine-tuned large language model for domain-specific with reinforcement learning. In *2024 3rd International Conference on Creative Communication and Innovative Technology (ICCIT)*, pages 1–6. IEEE.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

OpenAI, Josh Achiam, et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Brydon Parker, Alik Sokolov, Mahtab Ahmed, Matt Kalebic, Sedef Akinli Kocak, and Ofer Shai. 2022. Domain specific fine-tuning of denoising sequence-to-sequence models for natural language summarization. *arXiv preprint arXiv:2204.09716*.

Minghao Wu, Abdul Waheed, Chiyu Zhang, Muhammad Abdul-Mageed, and Alham Fikri Aji. 2023. Lamini-lm: A diverse herd of distilled models from large-scale instructions. *arXiv preprint arXiv:2304.14402*.

Anbang Xu, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. 2017. A new chatbot for customer service on social media. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pages 3506–3510.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

A Appendix

Instruction	The conversation consists of three sequential segments: {c1} (customer's utterance before the agent's response), {agent} (agent's response), and {c2} (customer's utterance following the agent's response). Please analyze the emotions in the conversation. Calculate the change in emotion using the formula: (c2's emotional score - c1's emotional score). Respond with a single float number only, within the range of -2 to 2. Do not include any explanation or additional text.
Input Data	{c1: [Customer's utterance before agent's response], agent: [Agent's response], c2: [Customer's utterance after agent's response]}

Table 6: Example of Prompt Template used for scoring

Instruction	You are a customer service chatbot. Generate a agent's response to the following customer message.
Inputs	Customer said: `{customer_inquiry}`
Labels	Agent said: `{agent_reply}`

Table 7: Example of Prompt Template used for Completion Sampling

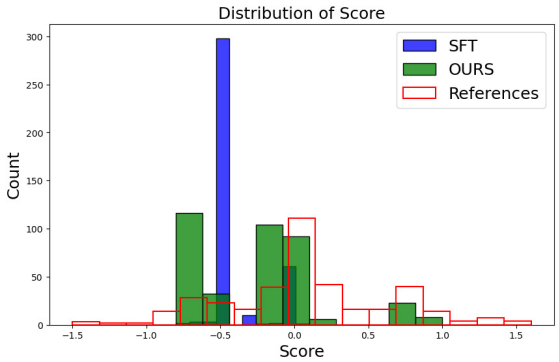

Figure 2: Score Distributions

Instruction	Given a customer message, compare two agent responses. `customer`: {customer} `response_A`: {response_A} `response_B`: {response_B} Evaluate the responses according to the following criteria: 1. Context appropriateness 2. Problem-solving effectiveness 3. Handling of negative emotions Select the better response for each criterion. If one response is clearly superior, label it as "A wins" or "B wins". If both are equivalent, label it as "Draw". Return your judgment in the following JSON format: `{"Context appropriateness": "A wins", "Problem-solving effectiveness": "Draw", "Handling of negative emotions": "B wins"}` No further explanation is required.
Input Data	{customer: [Customer message], response_A: [Response generated by T5 + ORPO w/ Sentimatic], response_B: [Response generated by T5 + SFT]}

Table 8: Prompt template used for comparative response evaluation