ICLEF: In-Context Learning with Expert Feedback for Explainable Style Transfer

Arkadiy Saakyan[1] and **Smaranda Muresan**[1,2]
[1]Department of Computer Science, Columbia University
[2]Data Science Institute, Columbia University
a.saakyan@cs.columbia.edu, smara@columbia.edu

Abstract

While state-of-the-art language models excel at the style transfer task, current work does not address explainability of style transfer systems. Explanations could be generated using large language models such as GPT-3.5 and GPT-4, but the use of such complex systems is inefficient when smaller, widely distributed, and transparent alternatives are available. We propose a framework to augment and improve a formality style transfer dataset with explanations via model distillation from ChatGPT. To further refine the generated explanations, we propose a novel way to incorporate scarce expert human feedback using in-context learning (ICLEF: In-Context Learning from Expert Feedback) by prompting ChatGPT to act as a critic to its own outputs. We use the resulting dataset of 9,960 explainable formality style transfer instances (e-GYAFC) to show that current openly distributed instruction-tuned models (and, in some settings, ChatGPT) perform poorly on the task, and that fine-tuning on our high-quality dataset leads to significant improvements as shown by automatic evaluation. In human evaluation, we show that models much smaller than ChatGPT fine-tuned on our data align better with expert preferences. Finally, we discuss two potential applications of models fine-tuned on the explainable style transfer task: interpretable authorship verification and interpretable adversarial attacks on AI-generated text detectors.

1 Introduction

Attribute style transfer is the task of transforming a given text along a particular style dimension, such as changing its formality, bias, or level of offensiveness (Lample et al., 2019; Sudhakar et al., 2019; Jin et al., 2022). Current state-of-the-art approaches to style transfer typically utilize pre-trained language models (Krishna et al., 2020; Reif et al., 2022; Jin et al., 2022). While highly effective, these methods lack interpretability (Belinkov and Glass, 2019).

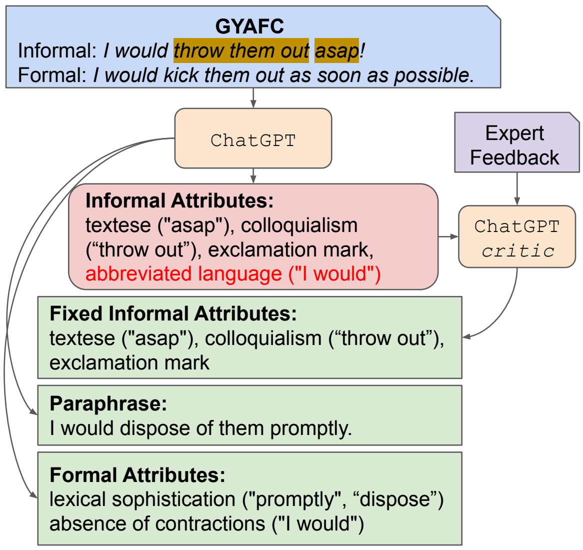


Figure 1: Proposed method to augment formality style transfer dataset GYAFC (Rao and Tetreault, 2018) with structured natural language explanations. ChatGPT is asked to generate the informal attributes of the input sentence, a formal paraphrase, and the formal attributes of the resulting sentence. Expert feedback is provided in the form of a few-shot prompt and a critic model is asked to imitate a human annotator in order to improve generated quality of the informal attributes.

Interpretable tools should improve people's understanding of the AI model and help recognize model uncertainty (Wang and Yin, 2022). Recent progress has shown that large language models (LLMs) are capable of generating faithful natural language explanations comprehensible to an end-user (Camburu et al., 2018; Majumder et al., 2022; Wiegreffe et al., 2021; Lyu et al., 2023). Besides helping the user, the generated explanations could act as a defense against spurious correlations (Ludan et al., 2023; Camburu et al., 2018) and annotation artifacts (McCoy et al., 2019; Poliak et al., 2018) .

We define the task of *explainable* style transfer to improve explainability of pre-trained language models applied to style transfer. Along with the paraphrase, the model needs to provide natural language explanations (Camburu et al., 2018) containing informality and formality attributes (see *Informal Attributes* and *Formal Attributes* boxes in Figure 1). While LLMs like GPT-4 (OpenAI, 2023a) and ChatGPT may be capable of producing plausible explanations for style transfer, this approach is inefficient due to these models' large parameter count[1]. Moreover, it is difficult to fine-tune these models to incorporate specialized expert feedback as they are only available through an API. Even if available publicly, fine-tuning a model of such scale is highly impractical. While elaborate prompt strategies could improve performance (Wei et al., 2022; Kojima et al., 2022, *inter alia*), this leads to increase in context length and corresponding variable costs that can be avoided by fine-tuning the model once.

A popular solution is to train smaller models on outputs from large models like GPT-4, PALM (Google, 2022), and ChatGPT; this approach is known as distillation (Ho et al., 2022; Magister et al., 2023). We propose to use distillation as a means to augment existing style transfer corpora with natural language explanations. Our approach is visualized in Figure 1. We start with the GYAFC (Rao and Tetreault, 2018) formality style transfer dataset, which is a parallel dataset of informal and formal sentences. We use ChatGPT to generate semi-structured natural language explanations elaborating on the attributes of source and target styles.

Due to the high chance of hallucinations (confabulations) (Ji et al., 2023), we ask expert linguists to evaluate the explanations and provide feedback. However, due to scarcity and high cost of expert feedback, it would not be possible to annotate the entire corpora in such a way. Inspired by the self-critique ability of LLMs (Madaan et al., 2023; Bai et al., 2022b; Saunders et al., 2022; Scheurer et al., 2023, *inter alia*), we prompt ChatGPT with a small number of expert corrections and ask it to act as an annotator criticizing its own outputs. We use this approach (ICLEF: In-Context Learning from Expert Feedback) to improve the generated instances before incorporating them in our explainable GYAFC dataset (e-GYAFC).

We validate the quality of e-GYAFC with expert evaluation. We then use this dataset to compare instruction-tuned models (in one-shot setting), ChatGPT, and fine-tuned models in their ability to perform the explainable style transfer task. Our investigation shows that task-specific fine-tuned models can outperform models significantly larger in size or fine-tuned on thousands of instructions.

Finally, we explore the applications of our fine-tuned models capable of providing explanations, i.e. self-rationalizing models (Wiegreffe et al., 2021; Alvarez Melis and Jaakkola, 2018), in two settings. First, we show that informality explanations could provide interpretable features for the author verification task. Second, we show that our formal-to-informal explainable style transfer model produces an interpretable adversarial attack on AI-generated text detection.

To summarize, our contributions are:

- A high-quality dataset for explainable formality style transfer (e-GYAFC) containing explanations for style attributes as well as improved formality paraphrases (§3).
- A framework to augment a style transfer dataset with semi-structured natural language explanations and improved paraphrases (ICLEF). Our method leverages model distillation and incorporates scarce yet high-quality expert feedback via in-context learning and self-critique (§3.2).
- Rigorous evaluation of current instruction-tuned models on the resulting dataset (§4). Via both automatic and human evaluation, we show that a smaller, specialized, fine-tuned model outperforms one-shot instruction-tuned models larger in size (§5).
- A discussion of practical importance of models fine-tuned on the explainable style transfer task in two downstream applications (§6).

We will release the data, models, and scripts to encourage further research in explainability, learning from human feedback, and style transfer. [2]

2 Related Work

Model distillation Model or knowledge distillation is a process of fine-tuning a smaller student model to imitate the behaviour of a more competent teacher model. (Beyer et al., 2022; Buciluundefined et al., 2006; Hinton et al., 2015). With the advance of large language models such as GPT-3 (Brown et al., 2020), knowledge distillation became

[1]We do not know the exact parameter count of ChatGPT, but for this work we will assume it is in the order of 175B.

[2]github.com/asaakyan/explain-st

an especially popular technique, allowing to generate datasets of similar quality to the crowd-sourced data (West et al., 2022), especially when combined with a model-in-the-loop approach (Wiegreffe et al., 2022; Bartolo et al., 2022; Chakrabarty et al., 2022). Unlike these approaches, we do not rely on a large number of crowdworkers but instead incorporate scarce expert feedback. Recent work explores learning from natural language explanations (Wang et al., 2023a; Ho et al., 2022; Magister et al., 2023), showing that large language models are capable of generating acceptable enough reasoning steps and explanations to fine-tune improved smaller models. We take this further by providing a task-specific detailed prompt to facilitate correct explanation generations.

Human feedback Reinforcement learning from human feedback (RLHF) (Stiennon et al., 2020), while effective, is difficult to implement and requires a lot of annotated data. Alternative approaches, such as Chain-of-Hindsight (Liu et al., 2023), Sequence Likelihood Calibration (Zhao et al., 2023), and Direct Preference Optimization (Rafailov et al., 2023) also require large corpora of paired feedback to fine-tune on. Imitation learning from human feedback (ILF) (Scheurer et al., 2023) utilizes human feedback to improve model-generated instances, and then fine-tunes on that data.

In this work, we focus on incorporating *expert* feedback which is naturally scarce and expensive to collect. We propose to do this by teaching a language model to generate synthetic feedback based on few examples of expert feedback. Motivated by the success of self-critiquing approaches (Madaan et al., 2023; Bai et al., 2022b; Saunders et al., 2022, *inter alia*), we condition the model on expert corrections and prompt it to criticize its outputs.

Explainability Natural language explanations have been utilized for a variety of tasks, such as natural language inference (Camburu et al., 2018), commonsense (Rajani et al., 2019; Aggarwal et al., 2021), figurative language understanding (Chakrabarty et al., 2022), social norm entailment (CH-Wang et al., 2023) (see more in Wiegreffe and Marasovic (2021)). In this work, we focus on creating natural language explanations for the style transfer task, which to our knowledge has not been addressed. Recently, there has been an increase in the number of works focusing on of structured explanations (Wiegreffe and Marasovic, 2021). Structured explanations provide well-defined templates for the model as opposed to free-text explanations for which it is hard to collect reliable human annotations and compare against them in automatic evaluation. Since the inherent structure of the style transfer task allows it, we collect semi-structured explanations. See more in §3.

Style transfer Style transfer approaches range from instruction-based methods (Reif et al., 2022) and paraphrasing (Krishna et al., 2020), to approaches focused on learning in low-resource settings (Patel et al., 2022). Much of style transfer work focuses on style representations that decouple style and content (Wegmann et al., 2022; Wegmann and Nguyen, 2021), however most of these methods are not interpretable. Interpretable approaches rely on constructing interpretable embeddings, such as LIWC (Tausczik and Pennebaker, 2010) or LISA (Patel et al., 2023). Unlike these approaches, we propose to use natural language explanations to further enhance model interpretability.

3 e-GYAFC: An Explainable Style Transfer Dataset

We build an explainable style transfer dataset by first augmenting GYAFC with natural language explanations generated by an LLM (§3.1), and then improving the generated data with ICLEF: in-context learning from expert feedback (§3.2).

3.1 Augmenting Style Transfer Datasets with Natural Language Explanations

The GYAFC (Rao and Tetreault, 2018) formality style transfer dataset contains parallel formal and informal expressions. The informal sentences are collected from Yahoo Answers, and formal paraphrases were crowdsourced using Amazon Mechanical Turk (AMT). Our initial goal was to generate structured natural language explanations for formality and informality attributes absent from the dataset. However, we noticed that the formal paraphrase generated by ChatGPT rivals or sometimes exceeds the quality of the crowdsourced paraphrase in the GYAFC data (see the paraphrase in Figure 1). Hence, we also use the model to generate a paraphrase and use it as part of our dataset.

Inspired by the observation that LLMs can generate intermediate reasoning steps (Wei et al., 2022; Kojima et al., 2022), we formulate the following multi-step generation task: given a GYAFC informal sentence s_i, generate a structured explanation of its informal attributes e_i, then generate a formal paraphrase s_f based on these attributes, then generate the formal attributes of the resulting paraphrase e_f jointly. In other words, we are modelling the conditional probability $p(e_i, s_f, e_f|s_i)$. We chose the Informal→Formal direction for data generation for two reasons. First, the informal sentences in GYAFC are not crowdsourced with AMT but are user-generated. Second, we hypothesize that it is easier for ChatGPT to generate formal text rather than diverse informal text that typically would have high perplexity.

We use a semi-structured format for the explanations. Specifically, we ask the model to generate a list of attributes followed by an excerpt from the sentence as the evidence: attribute ("evidence"), see examples in Figure 1. These explanations are more helpful to the user than traditional free-text explanations since they have a consistent format with textual evidence, making it easier to verify. Moreover, it is easier to evaluate structured explanations using automatic metrics since there is less variation of possible answers compared to free-text explanations.

We design a prompt taking advantage of the long context of ChatGPT. The prompt explains the task in detail and provides examples of formality style attributes and informality style attributes. In addition, we leverage the existing instances in the GYAFC dataset and provide "candidate" informality attributes by appending the set difference of words between informal and formal GYAFC sentences. This helps the model to more accurately identify the informal attributes, but does not bias the model into generating a formal sentence similar to the one in GYAFC. Instead, we encourage the model to faithfully generate a formal sentence based on the informal attributes it identifies. Details on our prompt of 3,391 tokens in length and can be found in Appendix A.

The resulting data allows us to train models in two directions: Formal→Informal (given s_f, generate e_f, s_i, e_i) and Informal→Formal (given s_i, generate e_i, s_f, e_f).

3.2 In-Context Learning from Expert Feedback (ICLEF)

We observe that ChatGPT generations can contain hallucinations and inaccuracies (e.g., the generated style attribute "abbreviated language" with the evidence "I would" in Figure 1). To improve the quality of the data, we turn to expert feedback, since previous work has identified that crowdworkers on platforms such as Amazon Mechanical Turk could be unreliable for open-ended generation tasks (Karpinska et al., 2021), and recently even rely on chatGPT to provide their answers (Veselovsky et al., 2023), amplifying the low-quality generations.

To improve generated explanations, we turn to expert linguists (defined as having a formal educational background in linguuistics). We hire 3 experts with formal education in linguistics: one with a bachelors degree in linguistics (E1), one with a bachelors degree in linguistics and a masters degree in education (E2), and one with a bachelor and master degrees in linguistics (E3) on Upwork[3]. We pay the experts 15-30 USD per hour depending on their asking rate and qualifications. Our annotation protocol provides a non-exhaustive reference to formality and informality attributes (also used in the ChatGPT prompt) and asks the annotators to be very critical and provide feedback on what attributes in e_i, e_f are missing or are incorrect.

We provide 50 generated examples for annotation, which takes each expert around 3 hours. In our preliminary investigations, we found that providing feedback from multiple experts hurts the identification of all the incorrect attributes as the model learns to imitate a certain experts' style instead of the content of the annotation. Due to the objective nature of the task, there is little concern of expert bias (as the attributes are either correct from a linguistics perspective or not), so we select a single expert's feedback. Qualitatively, we select the expert with most thorough feedback. Quantitatively, we select the expert with the largest number of corrections provided. We selected expert E3.

Given the high price and scarcity of expert feedback, we cannot use traditional fine-tuning pipelines or techniques to incorporate human feedback like Reinforcement learning from human feedback (RLHF) (Stiennon et al., 2020), Chain-of-Hindsight (Liu et al., 2023), Sequence Likelihood Calibration (Zhao et al., 2023), Direct Preference Optimization (Rafailov et al., 2023), or Imitation Learning from Language Feedback (Scheurer et al., 2023) that all rely on large-scale non-expert annotations. Instead, inspired by the recent obser-

[3] Upwork.com

vations that models can self-critique their outputs (Bai et al., 2022b; Saunders et al., 2022), we provide the model with the expert human feedback corrections in-context and prompt it to act as an annotator on the new instances (see Appendix A). We refer to the resulting model as GPT-Critic.

We observed that the rate of incorrectly generated attributes was higher for e_i than e_f ($\approx 29\%$ vs. $\approx 15\%$), perhaps since e_f is generated for the model's own paraphrase, and it is easier for the model to "comprehend" formal text. We also find that it is easier for GPT-Critic to identify incorrect instances rather than generate the missing ones, since generation of novel attributes may induce new hallucinations. We query the model to identify incorrect attributes in e_i, if there are any, and then ask it to provide a fixed version $\hat{e}_i$. In this way, we fixed $\approx 30\%$ of the generated data (2853 instances). The resulting data (which we refer to as e-GYAFC) contains 9,960 $s_i \leftrightarrow s_f$ instances with corresponding $\hat{e}_i, e_f$ attribute explanations. We randomly split the data into 8,000 training instances and 1,960 held-out test instances.

3.3 Dataset quality

Automatic evaluation We estimate paraphrase quality automatically using Mutual Implication Score and Formality Score (see §4.2) between our formal paraphrases and the ones in GYAFC. We find that our paraphrases are of comparable quality with a MIS of 81.32 vs. 83.08 for GYAFC, yet we achieve a much higher formality score of 91.62 vs. 80.14 for GYAFC (see example in Figure 1, where the GYAFC formal instance contains *kick them out*).

Human evaluation We re-hire the 2 expert annotators who performed the feedback annotations (A1 and A3) as well as an independent expert annotator with a masters degree in linguistics who did not perform the feedback annotations (A2). We ask their preferences on 100 randomly sampled instances with respect to the explanations (generated e_i vs. fixed $\hat{e}_i$) and paraphrases (GYAFC s_f vs. generated s_f). In addition, we ask for acceptability judgments for the preferred instance and separately for e_f. We report preference or equal rates (pref.)[4] and acceptability rates (accept.) in Table 1. Overall, we found that our dataset instances are considered acceptable (the average acceptability rate for e_i, s_f, e_f is 87%, 77%, 98% respectively) and that generated paraphrases are generally preferred to the ones in the GYAFC corpus (average 77%), and fixed explanations are preferred or are equal in quality with original generations on average in 90% of cases. Annotator A2 expressed concerns that the paraphrases may sound unnatural due to excessive formality (we believe it is due to the context in which the informal expression would be uttered) and that explanations sometimes miss punctuation errors (which, while important, is not critical for model-generated explanations). Due to high prevalence of the positive class, there is a high chance of random agreement, hence we provide a more granular look into the expert annotations than the inter-rater agreement in Table 1. Pairwise accuracy between annotator responses for all categories of e-GYAFC evaluation, and found that it averages at 81% across all categories.

[4]We compute preference or equal preference among acceptable instances. For acceptability, we compute dispreferred instances as unacceptable.

	e_i pref.	e_i accept.	s_f pref.	s_f accept.	e_f accept.
A1	91%	95%	64%	64%	98%
A2	87%	84%	76%	75%	96%
A3	91%	83%	91%	93%	100%

Table 1: Expert evaluation of e-GYAFC dataset quality. We report percentage of time each item was preferred, as well as acceptability judgements.

4 Experiments

We evaluate several models on their ability to generate generate e_f, s_i, e_i given s_f (Formal→Informal) and e_i, s_f, e_f given s_i (Informal→Formal) on a held-out test set. We evaluate how closely generated e_i, e_f match e-GYAFC explanations, and we evaluate the semantic closeness and paraphrase quality for s_i, s_f with reference-free metrics. We note that we focus on evaluating the performance on the *explainable* style transfer task rather than the style transfer performance itself.

4.1 Models

We test general-purpose instruction-tuned models (in a 1-shot scenario) of various sizes to show that they are not adequately equipped to deal with the explainable style transfer task. Additionally, we test the models fine-tuned on our domain-specific data. See hyperparameters and instructions in Appendix B. We do not fine-tune larger models as that contradicts the purpose of our paper to introduce more efficient models for explainable style transfer.

Instruction-tuned Models All instruction-tuned models are provided with the same one-shot prompt (modulo special token requirements) and generation parameters.

- MPT-7B-Instruct: built by finetuning MPT-7B (Team, 2023) on a dataset derived from the Databricks Dolly-15k (Databricks, 2023) and the Anthropic Helpful and Harmless (Bai et al., 2022a) datasets.

- Vicuna-13B (Chiang et al., 2023): an open-source chatbot trained by fine-tuning LLaMA on user-shared conversations collected from ShareGPT[5]. It places first in the Hugging-face Open LLM Leaderboard (Beeching et al., 2023) based on human and GPT-4 evaluation as of writing this paper.

- Falcon-40B (Almazrouei et al., 2023) causal decoder-only model trained on 1,000B tokens of RefinedWeb (Penedo et al., 2023) enhanced with curated corpora.

- Tülu-65B (Wang et al., 2023b) a 65B LLaMa model finetuned on a mixture of instruction datasets (FLAN V2 (Longpre et al., 2023), CoT (Wei et al., 2022), Dolly (Databricks, 2023), Open Assistant 1 (Köpf et al., 2023), GPT4-Alpaca (Peng et al., 2023), Code-Alpaca (Chaudhary, 2023), and ShareGPT).

ChatGPT As most of our data was generated after March 1st, 2023, we use `gpt-3.5-turbo-0301`. However, automatic evaluation still favors this model due to architectural similarities. Note that we do not provide long context, dataset hints, or human feedback-based critiques in this setting.

Fine-tuned models We fine-tune below models on e-GYAFC. $\rightarrow$ indicates fine-tuning two models in each direction, and $\leftrightarrow$ indicates fine-tuning on combined data in both directions.

- FLAN-T5-XL$_{\leftrightarrow}$ (Chung et al., 2022) approximately 3B parameter instruction-tuned model based on the T5 architecture (Raffel et al., 2020).

- LLaMA-7B$_{\rightarrow}$ (Touvron et al., 2023) model by Meta trained on 1 trillion tokens.

- Alpaca-7B$_{\rightarrow,\leftrightarrow}$ (Taori et al., 2023) a model fine-tuned from the LLaMA-7B model on 52K instruction-following demonstrations. In addition, we test Alpaca-7B$_{noexpl}$ as the model fine-tuned for Formal to Informal style transfer with no explanations provided in the fine-tuning data or in the output.

4.2 Automatic Metrics

- BLEU (Papineni et al., 2002): We would like to measure the amount of exactly matched formal and informal attributes and evidences between the generated structured explanation and e-GYAFC reference, which makes BLEU a fitting metric.

- Mutual Implication Score (MIS) (Babakov et al., 2022) is a symmetric measure of text semantic similarity based on a RoBERTa (Liu et al., 2019) model fine-tuned for natural language inference and on a paraphrase detection dataset. We chose this over alternative metrics like P-SP (Wieting et al., 2022) due to ease of implementation in the huggingface library and use in prior work (e.g., Patel et al., 2022).

- Formality/Informality Score[6]: RoBERTa (Liu et al., 2019) trained to predict for English sentences, whether they are formal or informal using GYAFC and Online Formality Corpus (OFC) (Pavlick and Tetreault, 2016). This classifier achieves 0.97 AUC on GYAFC making it highly reliable. We apply the sigmoid function on the logits to obtain a score between 0 and 1.

4.3 Human Evaluation: Preference Judgments

We select the best fine-tuned model, the best instruction-tuned model, and ChatGPT outputs (100 random samples) and evaluate them in terms of human preferences. We hire an expert with a masters degree in linguistics (same as A2 above) on Upwork and ask which model generation among the three do they prefer in terms of correctness and completeness of explanations as well as semantic preservation of paraphrase. We also perform a crowdworker evaluation on AMT hiring 3 workers per instance. To mitigate ChatGPT usage, we require to provide answer justifications and reject entries that clearly used ChatGPT (for Upwork, we also block copy-pasting with javascript).

[5] sharegpt.com

[6] huggingface.co/s-nlp/roberta-base-formality-ranker

5 Results and Analysis

Automatic evaluation Table 2 shows performance of models of various types (instruction-tuned one-shot, ChatGPT, fine-tuned models), sizes (7B-175B), and task direction combinations ($\rightarrow$, $\leftrightarrow$). We find that the Formal$\rightarrow$Informal direction is especially challenging, with only fine-tuned models being able to achieve informality scores above 50 (see Vicuna and ChatGPT examples in Table 3). We hypothesize it is due to the high perplexity of user-generated informal speech. Interestingly, smaller model size does not prevent a 13B Vicuna model from outperforming other instruction-tuned models even as large as 40B and 65B, indicating the importance of instruction tuning data for downstream performance. Moreover, we find that fine-tuning helps smaller models reach and surpass the performance of the teacher model (our best model achieves 55.13 average score while ChatGPT is at 47.53), demonstrating the benefits of fine-tuning domain-specific models on data with incorporated expert feedback.

The model fine-tuned without explanations (Alpaca$_{noexpl}$) achieves comparable performance, indicating that generating explanations does not significantly hurt performance on the standard style transfer task.

Many of the errors in instruction-tuned models are caused by not being able to follow a complex instruction that requires three steps to perform a task. For example, for all instruct models except Vicuna and ChatGPT the performance from Formality Attributes identification (step 1 of the task) to Informality Attribute identification (step 3 of the task) drops by 85-99%. This can be explained by the model not fully following the instruction and not generating all the 3 steps, as well as compounding errors due to autoregressive generation. Meanwhile, for fine-tuned models this drop is only 48-52%. For Vicuna and ChatGPT this percentage drop is in line with fine-tuned models at 53% and 48% respectively. Based on these findings, we posit that fine-tuning instruct models for better understanding of multi-step instructions could be an interesting area of future work.

Human evaluation Alpaca$_{\leftrightarrow}$ generations (the best fine-tuned model on our data according to average scores) are preferred to ChatGPT and Vicuna 53% of the time by the expert linguist (ChatGPT is preferred 42%) and 36% of the time by the AMT crowdworkers (compared to 34% for ChatGPT), indicating fine-tuned models are more aligned with both expert and crowdworker preferences. See a qualitative example in Table 3.

6 Discussion: downstream applications of models fine-tuned for explainable style transfer

Informality attribute explanations provide interpretable features for authorship verification We turn to the task of Authorship Verification (Martindale and McKenzie, 1995; Coulthard, 2004; Neal et al., 2017) utilizing PAN 2022 (Bevendorff et al., 2022) data. This is a binary classification task of deciding if two texts belong to the same author. We run Alpaca$_{\mathrm{IF}\rightarrow\mathrm{F}}$ model and extract explanations containing informality attributes and evidence (see Table 4).

In a preliminary evaluation of the usefulness of these features, we compute the similarity between authors by measuring the percentage of overlapping attributes. At most 15 sentences per author are considered. We then use the percentage as a classification score. We find that this simplistic approach, which does not even consider the evidences, achieves a classification performance of 0.60 AUC, indicating a non-random signal. This indicates a potential application of the explanations generated by our model to be used as interpretable authorship features that can be explored in future work.

Explainable formal $\rightarrow$ informal style transfer is an interpretable adversarial attack on AI-generated text detection methods, including retrieval Krishna et al. (2023) established that paraphrasing easily evades the detection of AI-generated text, and proposed a retrieval-based defense. However, we hypothesize that retrieval-based metrics will degrade as similarity between generations becomes more ambiguous, as is the case for formality style transfer. For example, an adversarial agent might generate a post containing misinformation in typical “formal" language generated by a language model like ChatGPT. This text is relatively detectable by current classifiers and 100% detectable by retrieval-based methods. However, the agent might apply a style transfer model to lower the formality of the message. Alarmingly, not only this accomplishes the goal of spreading the AI-generated message more effectively as the result looks more like user-generated text, but, as

Model	Size	Formal → Informal				Informal → Formal				Average
		Form.Attrs. BLEU	MIS	Informality	Inform.Attrs. BLEU	Inform.Attrs. BLEU	MIS	Formality	Form.Attrs. BLEU	
MPT	7B	24.59	51.84	12.35	2.10	23.26	46.26	53.33	0.86	26.82
Vicuna	13B	23.16	<u>83.24</u>	<u>36.52</u>	<u>10.97</u>	27.31	<u>61.22</u>	**92.26**	<u>9.88</u>	<u>43.07</u>
Falcon	40B	8.38	28.12	12.78	1.23	20.80	38.01	58.40	7.13	21.86
Tülu	65B	<u>24.90</u>	19.60	6.68	0.02	<u>27.76</u>	26.69	25.50	0.28	16.43
ChatGPT	175B*	25.48	**86.85**	43.69	13.18	30.12	77.21	92.06	11.62	47.53
FLAN-T5$_{\rightarrow}$	3B	0.00	8.54	0.02	0.00	0.00	9.82	1.05	0.00	2.43
LLaMA$_{\rightarrow}$	7B	39.64	85.31	56.10	19.86	38.02	81.80	90.91	25.10	54.59
Alpaca$_{noexpl}$	7B	-	85.34	54.75	-	-	**83.20**	91.10	-	-
Alpaca$_{\rightarrow}$	7B	39.98	84.70	56.74	19.22	**40.56**	81.69	91.12	24.71	54.84
Alpaca$_{\leftrightarrow}$	7B	**40.42**	81.76	**60.38**	**21.11**	40.34	79.43	91.81	**25.75**	**55.13**

Table 2: Performance of different models across subdivisions of Formal to Informal and Informal to Formal style transfer. Models in the top 4 rows are instruction-tuned evaluated in a one-shot setting, row 5 is ChatGPT (was used to generate the data), the bottom rows are fine-tuned models. Best overall performance is in bold, best performance among non-GPT instruction models is underlined.

Formal (Input) If it is feasible, allow love to prevail.
Vicuna
`Formal Attributes:` passive voice ("it is feasible", "allow love to prevail"), use of verb "to be" ("is feasible", "prevail") `Informal:` It's okay to let love rule. `Informal Attributes:` active voice ("it's okay", "let love rule"), informal verb use ("rule")
ChatGPT
`Formal Attributes:` conditional clause ("if it is feasible"), elevated vocabulary ("allow love to prevail") `Informal:` If it's possible, just go with love. `Informal Attributes:` simple language ("if it's possible"), imperative form ("just go"), contraction ("it's")
Alpaca$_{\leftrightarrow}$
`Formal Attributes:` precise language ("feasible"), use of conditional clause ("if it is feasible"), lexical sophistication ("prevail") `Informal:` let love win if it's possible. `Informal Attributes:` contraction ("it's"), casual tone ("let love win")

Table 3: Comparison between generations from a one-shot instruction-tuned model (Vicuna, ChatGPT, and our best fine-tuned model). Vicuna outputs wrong attributes. ChatGPT paraphrase does not sound natural.

we show, it also decreases the chances of being detected as AI-generated by current methods.

We test this in the following setting: we use an online dataset of political tweets [7], and sample 30 of them. We ask ChatGPT to generate a political commentary post on the topic of the tweet (GPT-F), as well as an informal paraphrase of the said post (GPT-Inf). We manually annotate the resulting summaries and select those that look like they could be legitimate political messages posted on social media and have valid paraphrases. We then use our Alpaca$_{\mathrm{F}\rightarrow\mathrm{IF}}$ model to generate an informal paraphrase of the GPT-F posts sentence-by-sentence. We also verify that these paraphrases are semantically valid and close to the original GPT-Formal post and select 24 high-quality generations. We choose a relatively small sample since we want to verify the paraphrase was still close to the original sentence manually to ensure semantic control for the experiment.

Attribute	**Evidence**
Colloquialism	"assumed they all started off low!?", "typing it out"
Textese	"xx"
Informal Vocabulary	"give you a call", "arrange something"
Informal Tone	"hoping to borrow a couple of charging leads"

Table 4: Informality features could be used for authorship identification: on the left, informality attributes identified by our model, on the right, textual evidence from the author's text provided by the model.

We report detection scores[8] from 4 methods surveyed by Krishna et al. (2023): GPTZero (Tian, 2023), OpenAI classifier (OpenAI, 2023b), DetectGPT (Mitchell et al., 2023), and their proposed

[7] kaggle.com

[8] Since we do not have the human baseline text, we do not report the performance at 1% FPR, but for our study it is sufficient to show the scores decrease in absolute terms.

retrieval methods based on BM25 (Robertson and Zaragoza, 2009) or P-SP (Wieting et al., 2022) retrievers. As can be seen in Table 5, the formal-to-informal transfer model significantly decreases detection scores of all AI-generated text detection methods, including the retrieval-based one (despite the fact that the retrieval corpus is significantly smaller than it would be in real-world). Interestingly, for the BM25 retrieval method, the ChatGPT paraphrases are slightly harder to detect than $\text{Alpaca}_{\text{F} \rightarrow \text{IF}}$, whereas it is easier for all other methods. Since we used ChatGPT to generate the original posts, we could not use the watermarking methods (Kirchenbauer et al., 2023), but this can be explored in future work.

This result highlights the need to investigate new methods of detecting style transferred AI-generated text. As formality style transfer remains an effective attack, informality features produced by our model could help improve such classifiers. We leave this investigation for future work.

Models	GPTZero	OpenAI	GPTDetect	BM25	P-SP
GPT-F	85.92	70.64	104.88	100	100
GPT-Inf	69.58	54.24	65.42	**48.15**	74.99
F → IF	**6.11**	**44.86**	**54.92**	58.68	**74.08**

Table 5: Performance of various AI-generated text detectors on informal paraphrases from our model. Even retrieval methods perform poorly in this setting.

7 Conclusion and Future Work

We propose a framework to augment a formality style transfer dataset with semi-structured natural language explanations. We leverage long context and existing data cues to guide the teacher model to produce high-quality generations. To further improve quality and incorporate expert feedback, we propose ICLEF (In-Context Learning from Expert Feedback), a method to incorporate a small amount of expert feedback via in-context learning and self-critique. We provide a rigorous evaluation of contemporaneous instruction-tuned models on the resulting dataset and find promising areas of future work, such as multi-step instruction following. Future work could use the generated explanations for authorship verification and AI-generated text detection.

Limitations

The GYAFC dataset does not contain all types of informal and formal language, namely, they mostly focus on interpersonal relationships (the subset used for this paper) and entertainment. Future work could consider extending our approach to other style transfer datasets, including ones more encompassing of formality.

As OpenAI may deprecate the model at any time, some ChatGPT results may not be fully reproducible in the future. However, we preserved the generations used for experiments.

While our methods are intended to produce faithful explanations, there can still be instances when a model does not rely on the attributes in order to complete the paraphrase. We also observed that hallucinations can still be present in our fine-tuned models' explanations and hope that future work will try to address these issues.

One limitation of our method is that we used a relatively small number of experts to conduct our study. However, we believe that this setting mirrors real-life conditions where experts are usually scarcely available. We hope our approach provides a more general framework for incorporating expert feedback that can be adjusted to experts' needs (e.g., a forensic linguist may require a different style transfer explanation than a literary critic).

Fine-tuning and running inference on large models requires expensive computational resources. However, we hope that our study presents a convincing argument that fine-tuning a smaller model once may be more efficient and accurate than running a large general-purpose model with elaborate long-context prompts.

Ethics Statement

Our model is intended for explainable style transfer. In our study, we show how style transfer can be used to evade AI-text detectors. Similarly to Krishna et al. (2023), we reiterate that this is not to provide a way to attack such systems, but to bring awareness to the community that current detectors are easy to evade. Moreover, we bring to attention the issue of detecting text on which style transfer paraphrase has been applied. We hope that future work develops systems capable of defending against such attacks, perhaps utilizing explanations generated by our system.

Acknowledgements

This research is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via the HIATUS Program contract #2022-22072200005. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Shourya Aggarwal, Divyanshu Mandowara, Vishwajeet Agrawal, Dinesh Khandelwal, Parag Singla, and Dinesh Garg. 2021. Explanations for CommonsenseQA: New Dataset and Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3050–3065, Online. Association for Computational Linguistics.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. Falcon-40B: an open large language model with state-of-the-art performance.

David Alvarez Melis and Tommi Jaakkola. 2018. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Nikolay Babakov, David Dale, Varvara Logacheva, and Alexander Panchenko. 2022. A large-scale computational study of content preservation measures for text style transfer and paraphrase generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 300–321, Dublin, Ireland. Association for Computational Linguistics.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. 2022a. Training a helpful and harmless assistant with reinforcement learning from human feedback.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022b. Constitutional ai: Harmlessness from ai feedback.

Max Bartolo, Tristan Thrush, Sebastian Riedel, Pontus Stenetorp, Robin Jia, and Douwe Kiela. 2022. Models in the loop: Aiding crowdworkers with generative annotation assistants. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3754–3767, Seattle, United States. Association for Computational Linguistics.

Edward Beeching, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. 2023. Open llm leaderboard. `https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard`.

Yonatan Belinkov and James Glass. 2019. Analysis Methods in Neural Language Processing: A Survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Janek Bevendorff, Berta Chulvi, Elisabetta Fersini, Annina Heini, Mike Kestemont, Krzysztof Kredens, Maximilian Mayerl, Reynier Ortega-Bueno, Piotr Pęzik, Martin Potthast, et al. 2022. Overview of pan 2022: Authorship verification, profiling irony and stereotype spreaders, and style change detection. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 13th International Conference of the CLEF Association, CLEF 2022, Bologna, Italy, September 5–8, 2022, Proceedings*, pages 382–394. Springer.

Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. 2022. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10925–10934.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens

Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Cristian Buciluundefined, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 535–541, New York, NY, USA. Association for Computing Machinery.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Sky CH-Wang, Arkadiy Saakyan, Oliver Li, Zhou Yu, and Smaranda Muresan. 2023. Sociocultural norm similarities and differences via situational alignment and explainable textual entailment.

Tuhin Chakrabarty, Arkadiy Saakyan, Debanjan Ghosh, and Smaranda Muresan. 2022. FLUTE: Figurative language understanding through textual explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7139–7159, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sahil Chaudhary. 2023. Code alpaca: An instruction-following llama model for code generation. `https://github.com/sahil280114/codealpaca`.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Malcolm Coulthard. 2004. Author identification, idiolect, and linguistic uniqueness. *Applied linguistics*, 25(4):431–447.

Databricks. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm. Blog post.

Google. 2022. Pathways language model (palm): Scaling to 540 billion parameters for breakthrough performance.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2022. Large language models are reasoning teachers. *ArXiv*, abs/2212.10071.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. 2022. Deep learning for text style transfer: A survey. *Computational Linguistics*, 48(1):155–205.

Marzena Karpinska, Nader Akoury, and Mohit Iyyer. 2021. The perils of using Mechanical Turk to evaluate open-ended text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1265–1285, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *arXiv preprint arXiv:2303.13408*.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 737–762, Online. Association for Computational Linguistics.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. 2023. Openassistant conversations – democratizing large language model alignment.

Guillaume Lample, Sandeep Subramanian, Eric Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-attribute text rewriting. In

International Conference on Learning Representations.

Hao Liu, Carmelo Sferrazza, and Pieter Abbeel. 2023. Chain of hindsight aligns language models with feedback.

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.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*.

Josh Magnus Ludan, Yixuan Meng, Tai Nguyen, Saurabh Shah, Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2023. Explanation-based finetuning makes models more robust to spurious cues.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching small language models to reason.

Bodhisattwa Prasad Majumder, Oana Camburu, Thomas Lukasiewicz, and Julian Mcauley. 2022. Knowledge-grounded self-rationalization via extractive and natural language explanations. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 14786–14801. PMLR.

Colin Martindale and Dean McKenzie. 1995. On the utility of content analysis in author attribution: "the federalist". *Computers and the Humanities*, 29(4):259–270.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature.

Tempestt Neal, Kalaivani Sundararajan, Aneez Fatima, Yiming Yan, Yingfei Xiang, and Damon Woodard. 2017. Surveying stylometry techniques and applications. *ACM Comput. Surv.*, 50(6).

OpenAI. 2023a. Gpt-4 technical report.

OpenAI. 2023b. New ai classifier for indicating ai-written text.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ajay Patel, Nicholas Andrews, and Chris Callison-Burch. 2022. Low-resource authorship style transfer with in-context learning.

Ajay Patel, Delip Rao, and Chris Callison-Burch. 2023. Learning interpretable style embeddings via prompting llms.

Ellie Pavlick and Joel Tetreault. 2016. An empirical analysis of formality in online communication. *Transactions of the Association for Computational Linguistics*, 4:61–74.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. Association for Computational Linguistics.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140, New Orleans, Louisiana. Association for Computational Linguistics.

Emily Reif, Daphne Ippolito, Ann Yuan, Andy Coenen, Chris Callison-Burch, and Jason Wei. 2022. A recipe for arbitrary text style transfer with large language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 837–848, Dublin, Ireland. Association for Computational Linguistics.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. Self-critiquing models for assisting human evaluators.

Jérémy Scheurer, Jon Ander Campos, Tomasz Korbak, Jun Shern Chan, Angelica Chen, Kyunghyun Cho, and Ethan Perez. 2023. Training language models with language feedback at scale.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc.

Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. "transforming" delete, retrieve, generate approach for controlled text style transfer. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3269–3279, Hong Kong, China. Association for Computational Linguistics.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54.

MosaicML NLP Team. 2023. Introducing mpt-7b: A new standard for open-source, commercially usable llms. Accessed: 2023-06-20.

Edward Tian. 2023. Ai content detector and writing captcha for chat gpt, openai, bard, education.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Veniamin Veselovsky, Manoel Horta Ribeiro, and Robert West. 2023. Artificial artificial artificial intelligence: Crowd workers widely use large language models for text production tasks.

PeiFeng Wang, Aaron Chan, Filip Ilievski, Muhao Chen, and Xiang Ren. 2023a. PINTO: Faithful language reasoning using prompt-generated rationales. In *The Eleventh International Conference on Learning Representations*.

Xinru Wang and Ming Yin. 2022. Effects of explanations in ai-assisted decision making: Principles and comparisons. *ACM Trans. Interact. Intell. Syst.*, 12(4).

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023b. How far can camels go? exploring the state of instruction tuning on open resources.

Anna Wegmann and Dong Nguyen. 2021. Does it capture STEL? a modular, similarity-based linguistic style evaluation framework. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7109–7130, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Anna Wegmann, Marijn Schraagen, and Dong Nguyen. 2022. Same author or just same topic? towards content-independent style representations. In *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 249–268, Dublin, Ireland. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: from general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of* *the Association for Computational Linguistics: Human Language Technologies*, pages 4602–4625, Seattle, United States. Association for Computational Linguistics.

Sarah Wiegreffe, Jack Hessel, Swabha Swayamdipta, Mark Riedl, and Yejin Choi. 2022. Reframing human-AI collaboration for generating free-text explanations. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 632–658, Seattle, United States. Association for Computational Linguistics.

Sarah Wiegreffe and Ana Marasovic. 2021. Teach me to explain: A review of datasets for explainable natural language processing. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran.

Sarah Wiegreffe, Ana Marasović, and Noah A. Smith. 2021. Measuring association between labels and free-text rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10266–10284, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Wieting, Kevin Gimpel, Graham Neubig, and Taylor Berg-kirkpatrick. 2022. Paraphrastic representations at scale. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 379–388, Abu Dhabi, UAE. Association for Computational Linguistics.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. 2023. Slic-hf: Sequence likelihood calibration with human feedback.

A Prompts

ChatGPT Explanation Generation Prompt We provide an instruction as a system prompt ("You are an expert forensic linguist...") and 6 examples of the task in the OpenAI ChatML format.[9]

Instruction: You are an expert forensic linguist. Your task is to identify infromal attributes in a setnece, modify them to create a formal sentence, and then output the attributes of the generated formal sentence. Use the following format: attribute (excertp from text in quotation marks). Make sure to provide a complete list of informal and formal atttributes. Focus on what has changed between formal and informal sentences. Informal writing tends to be more casual, personal, and conversational than formal writing. Here are some common features of informal writing: Contractions: Informal writing often uses contractions, such as "I'm," "can't," "won't," and "they've," which are generally avoided in formal writing. ...

Examples: Informal greetings and sign-offs: Informal writing often uses casual greetings, such as "Hi" or "Hey," and sign-offs like "Cheers" or "Take care." Informal: if ur under 18 u have a BIG PROBLEM. Informal Candidates: '18', 'BIG', 'PROBLEM.', 'if', 'u', 'ur Attributes of Informal Style: textese ("ur", "u"), capitalization ("BIG PROBLEM"), colloquialism ("BIG PROBLEM")...

Prompt for ChatGPT ICLEF generation You are an extremely attentive and critical annotator with background in stylometry and linguistics. You will be provided with an informal sentence. You will also be provided with an explanation of its informality attributes. Decide whether the explanation is incorrect, and if so, describe what attributes were listed incorrectly.

EXAMPLES: Informal Sentence: Look, If you really like this person, just tell her. Informal Attributes: colloquialism ("just tell her"), contraction ("If you"), simple sentence structure. Attributes Listed Incorrectly: contraction ("If you" is not a contraction)...

B Instructions and hyperparameters

Fine-tuning hyperparameters We fine-tune all models using the script provided in the Stanford Alpaca repository.[10] We use exact same hyperparameters, except for batch size which we adjust to 1 due to memory constraints. We fine-tune our models on 4 A100 NVIDIA 40GB GPUs. We train our models for 3 epochs with learning rate 2e-05 with cosine rate scheduler and warmup ratio of 0.03.

Inference parameters We use the same hyperparameters for generation across all models, that is temperature=0.7, top p=0.9, max new tokens = 256. We use the huggingface library.

One-shot instruction is provided below: Identify informal attributes in a given sentence, modify them to create a formal sentence, and then output the attributes of the generated formal sentence.

For example:

Informal: how can you tell if a girl likes you or not?

Informal Attributes: direct question form ("how can you tell"), informal language ("girl", "likes you") Formal: What are some indications that a woman may be interested in you? Formal Attributes: indirect question form ("what are some indications"), lexical sophistication ("woman", "interested in you")

For the following sentence, identify informal attributes in a given sentence, modify them to create a formal sentence, and then output the attributes of the generated formal sentence.

For Tulu, we add the <asistant> and <user> tokens as advised by model developers.

C Annotation protocols

The screenshots form explanation interfaces are provided below in Figures 2, 3, 4.

[9] github.com/openai/openai-python/blob/main/chatml.md

[10] Alpaca GitHub

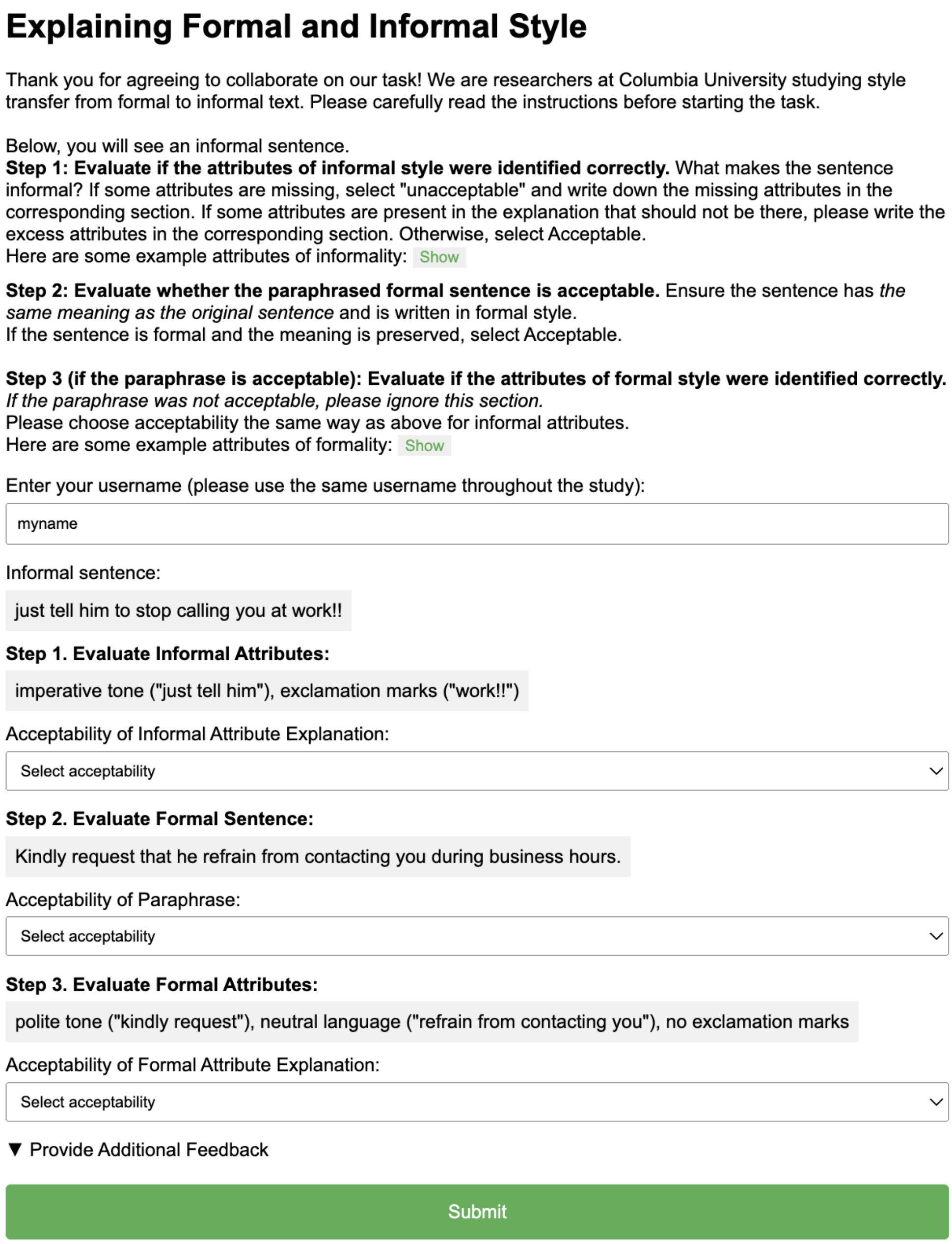
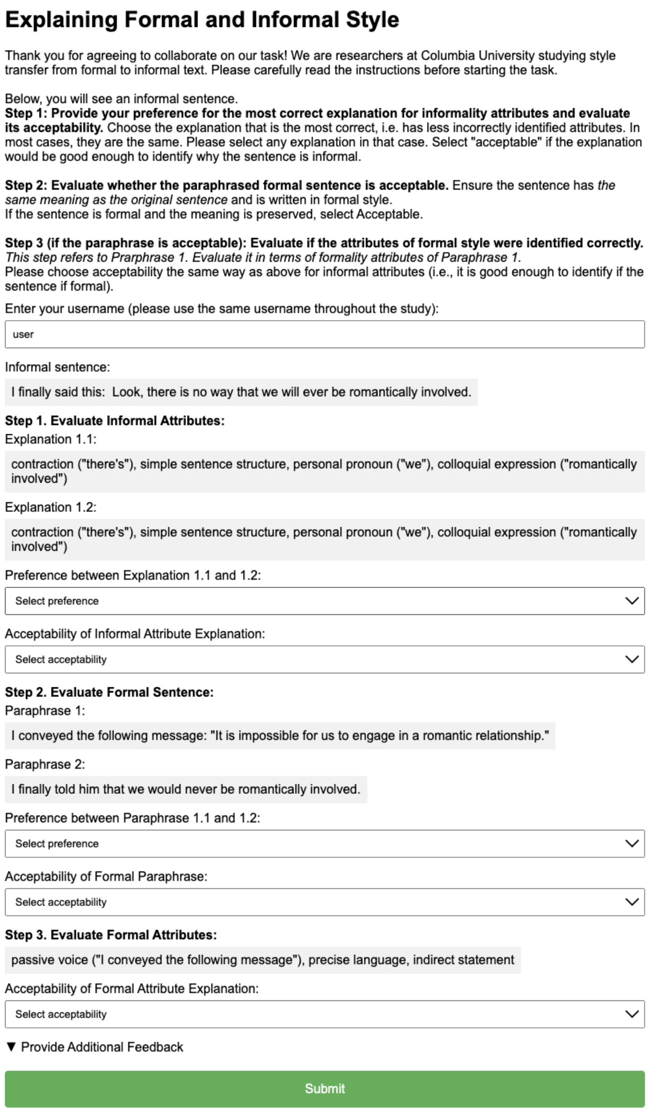


Figure 2: Annotation to gather feedback for ICLEF.



Explaining Formal and Informal Style

Thank you for agreeing to collaborate on our task! We are researchers at Columbia University studying style transfer from formal to informal text. Please carefully read the instructions before starting the task.

Below, you will see an informal sentence.
Step 1: Provide your preference for the most correct explanation for informality attributes and evaluate its acceptability. Choose the explanation that is the most correct, i.e. has less incorrectly identified attributes. In most cases, they are the same. Please select any explanation in that case. Select "acceptable" if the explanation would be good enough to identify why the sentence is informal.

Step 2: Evaluate whether the paraphrased formal sentence is acceptable. Ensure the sentence has *the same meaning as the original sentence* and is written in formal style.
If the sentence is formal and the meaning is preserved, select Acceptable.

Step 3 (if the paraphrase is acceptable): Evaluate if the attributes of formal style were identified correctly.
This step refers to Prarphrase 1. Evaluate it in terms of formality attributes of Paraphrase 1.
Please choose acceptability the same way as above for informal attributes (i.e., it is good enough to identify if the sentence if formal).

Enter your username (please use the same username throughout the study):

user

Informal sentence:

I finally said this: Look, there is no way that we will ever be romantically involved.

Step 1. Evaluate Informal Attributes:
Explanation 1.1:

contraction ("there's"), simple sentence structure, personal pronoun ("we"), colloquial expression ("romantically involved")

Explanation 1.2:

contraction ("there's"), simple sentence structure, personal pronoun ("we"), colloquial expression ("romantically involved")

Preference between Explanation 1.1 and 1.2:

Select preference

Acceptability of Informal Attribute Explanation:

Select acceptability

Step 2. Evaluate Formal Sentence:
Paraphrase 1:

I conveyed the following message: "It is impossible for us to engage in a romantic relationship."

Paraphrase 2:

I finally told him that we would never be romantically involved.

Preference between Paraphrase 1.1 and 1.2:

Select preference

Acceptability of Formal Paraphrase:

Select acceptability

Step 3. Evaluate Formal Attributes:

passive voice ("I conveyed the following message"), precise language, indirect statement

Acceptability of Formal Attribute Explanation:

Select acceptability

▼ Provide Additional Feedback

Submit

Figure 3: Annotation for eGYAFC data acceptability and preferences.

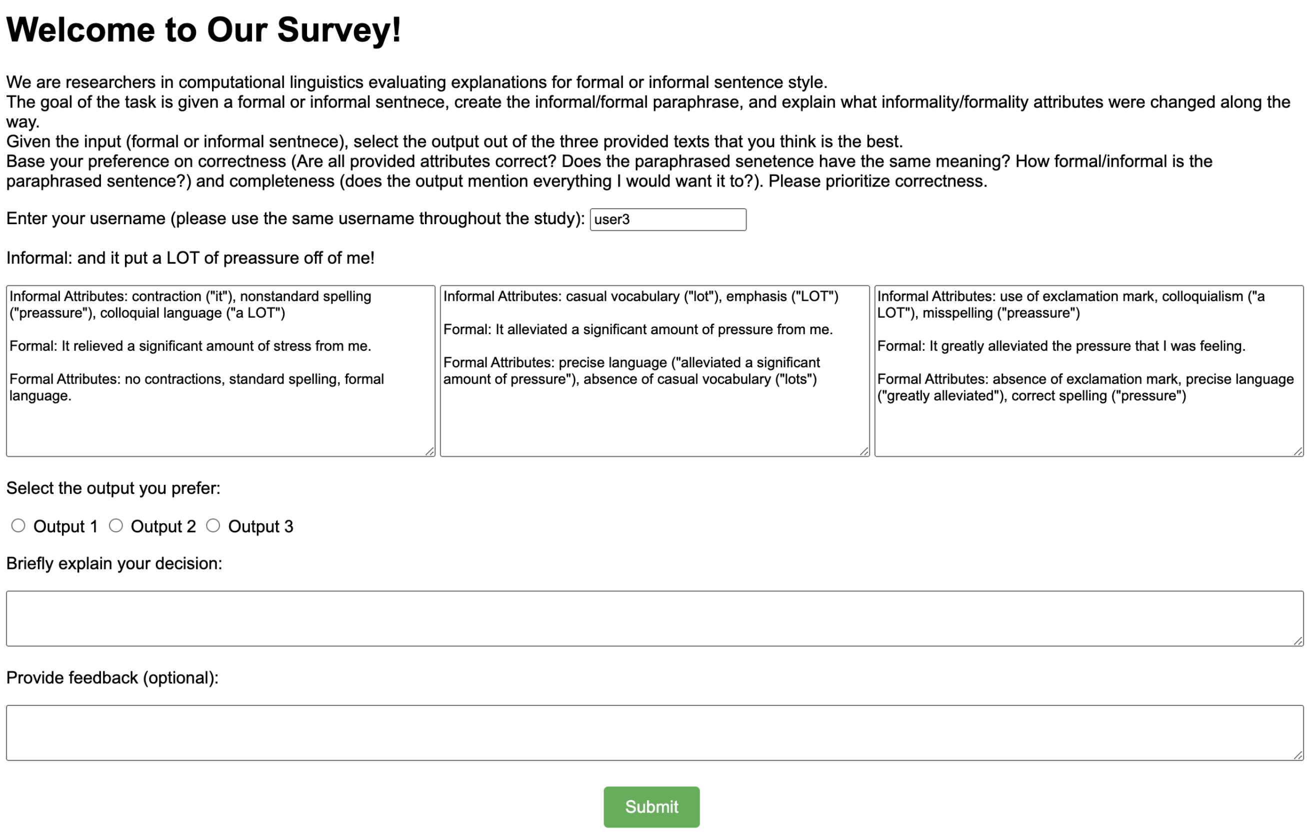


Figure 4: Annotation for model preferences.