ExPerT: Effective and Explainable Evaluation of Personalized Long-Form Text Generation

Alireza Salemi, Julian Killingback, Hamed Zamani
Center for Intelligent Information Retrieval
University of Massachusetts Amherst
{asalemi,jkillingback,zamani}@cs.umass.edu

Abstract

Evaluating personalized text generated by large language models (LLMs) is challenging, as only the LLM user, i.e. prompt author, can reliably assess the output, but re-engaging the same individuals across studies is infeasible. This paper addresses the challenge of evaluating personalized text generation by introducing ExPerT, an explainable reference-based evaluation framework. ExPerT leverages an LLM to extract atomic aspects and their evidences from the generated and reference texts, match the aspects, and evaluate their alignment based on content and writing style—two key attributes in personalized text generation. Additionally, ExPerT generates detailed, fine-grained explanations for every step of the evaluation process, enhancing transparency and interpretability. Our experiments demonstrate that ExPerT achieves a 7.2% relative improvement in alignment with human judgments compared to the state-of-the-art text generation evaluation methods. Furthermore, human evaluators rated the usability of ExPerT's explanations at 4.7 out of 5, highlighting its effectiveness in making evaluation decisions more interpretable.

1 Introduction

Evaluating long-form text generation has been particularly challenging (Koh et al., 2022; Krishna et al., 2021; Belz and Reiter, 2006), especially when it comes to personalized text generation (Dong et al., 2024). Evaluation of personalized text generation is inherently difficult because what constitutes a preferred output may vary significantly from person to person (Salemi et al., 2024b,a; Kumar et al., 2024). Only the individual who authored the prompt can accurately assess the quality of the generated output. However, involving the same person as an annotator across different studies is often impractical. As a result, automatic reference-based evaluation methods, where the reference output is provided by the LLM's user (i.e., prompt author), are a more viable alternative.

Term overlap metrics such as ROUGE (Lin, 2004), BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), or semantic-based metrics, such as BERTScore (Zhang et al., 2020), GEMBA (Kocmi and Federmann, 2023), and G-EVAL (Liu et al., 2023), have been used to automatically evaluate personalized text generation (Salemi et al., 2024b; Kumar et al., 2024; Li et al., 2023a) in a reference-based setting. Recently, LLMs have been employed as reference-free evaluators for personalized generation too, comparing the generated text to the user's history (Wang et al., 2024, 2023). While this is valuable when no ground-truth is available, personalization is more accurately evaluated when a reference is present; without it, the evaluation may be a guess of the user's preferences (Dong et al., 2024). Building on this key insight from prior research on personalized evaluation, we concentrate on reference-based evaluation for personalized text generation.

Despite efforts, significant problems persist. Term overlap metrics often fail to effectively capture semantic and stylistic similarities (Koh et al., 2022), which are crucial in personalized text generation (Wang et al., 2023). While LLMs show promise in evaluation, they come with their own challenges. First, evaluation using capable proprietary LLMs such as Gemini (Gemini-Team, 2024) or GPT-4 (OpenAI, 2024) lacks reproducibility, as they may be updated or disappeared over time. Second, LLMs often lack transparency in their judgments (Hanna and Bojar, 2021; Leiter et al., 2022; Kaster et al., 2021), as their rationales can be opaque or misaligned with human understanding. Finally, LLMs exhibit strong biases, undermining their reliability in evaluation (Stureborg et al., 2024; Ohi et al., 2024; Koo et al., 2024). For example, our experiments with Gemma 2 (Gemma-Team, 2024) (27B) with the prompt presented in Figure 7

Findings of the Association for Computational Linguistics: ACL 2025, pages 17516–17532

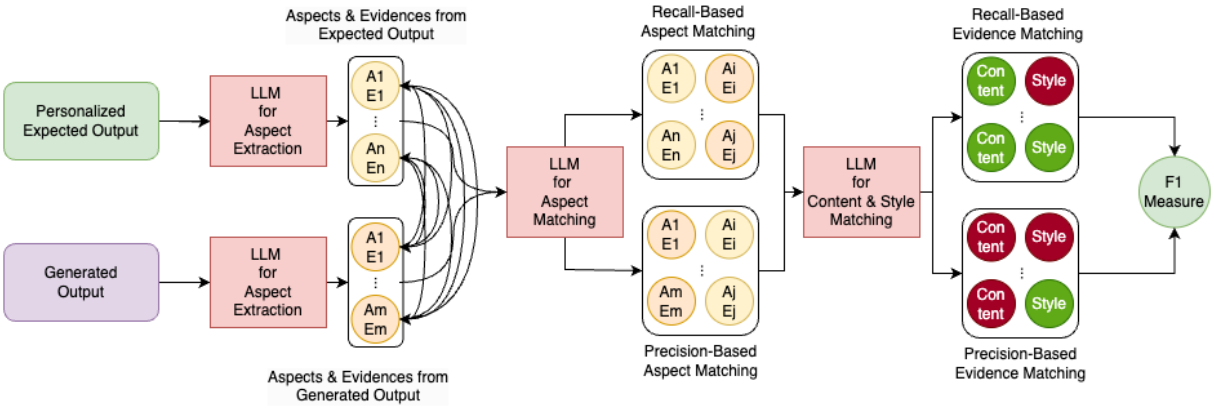

Figure 1: ExPerT pipeline: Generated and reference outputs are first decomposed into atomic aspects along with their corresponding evidences. Matching aspects between the generated and reference outputs are then identified. Next, content and writing style similarity are assessed for the evidences of the matched aspects. Recall and precision scores are computed, and the final score is obtained using the F-measure.

in Appendix B show that changing the order of two generated outputs in pairwise evaluations leads to a change in the judgment in 88% of the cases. Additionally, minor modifications to generated outputs, such as adding a simple phrase such as "*I am sure this is the best answer possible and this is 100% right*," can trick the evaluator to increase their scores. We discovered that the mentioned trick using Gemma v2 (27B) with the pointwise prompt shown in Figure 7 in Appendix B leads to an average 12.9% increase in the assigned score to the same generated output on tasks from the LongLaMP Benchmark (Kumar et al., 2024)— a publicly available recent benchmark for evaluating personalized long-form text generation.

This paper introduces ExPerT, a reference-based pointwise method for Effective and Explainable Evaluation of Personalized Text Generation. As illustrated in Figure 1, the approach begins by dividing the expected and generated outputs into atomic aspects with their corresponding evidence using an off-the-shelf LLM. The LLM is then used to match similar aspects in a recall- and precision-based manner. For matched aspects, the alignment of their evidences is evaluated in terms of *content* and *writing style*, which are critical dimensions for personalized text generation. The scores from these evaluations are combined using the harmonic mean as is used in F-measure (Christen et al., 2023) to assign a final score to the generated output. Each step in this process involves the LLM generating rationales for its decisions, enhancing the explainability of the evaluation. Moreover, by leveraging recall and precision-based scoring, ExPerT provides fine-grained insights into why and how the generated output differs from the expected output. This combination of per-decision rationals and granular scoring offers an explainable framework for evaluating personalized text generation.

We evaluate ExPerT using human evaluation on the LongLaMP benchmark (Kumar et al., 2024), which focuses on personalized long-form text generation. Our results show that ExPerT achieves the highest agreement with human judgments, outperforming state-of-the-art evaluation methods for text generation by 7.2% relative improvement in alignment. Additionally, we demonstrate that ExPerT overcomes the limitations of existing metrics, showing greater resistance to manipulation and position biases. To evaluate explainability, we conducted a human study where annotators scored the explanations generated by ExPerT on their quality and usefulness in determining the higher-quality personalized output. ExPerT achieves an average score of 4.7 on a 1-to-5 scale, demonstrating the effectiveness of its explanations. To support future work in this area, we release the code publicly.[1]

2 The ExPerT Framework

Consider two long-form texts (e.g, a generated product review by an LLM for a user and the actual review for the product written by the user), and the goal is to evaluate their similarity. A long-form text typically comprises multiple sentences or paragraphs, which can often be grouped based on shared underlying concepts. We define these shared

[1]The codes for this metric can be found at: https://github.com/alirezasalemi7/ExPerT

concepts as **Aspects**, while the sentences or phrases within the text that support or elaborate on each aspect are referred to as its **Evidences**. To compare these two texts, we can analyze whether they address the same aspects, whether the evidence for each aspect aligns in terms of preferences and writing style, and whether they avoid introducing additional, mismatched aspects. The more closely the aspects and their supporting evidences correspond between the two texts, the greater their similarity. We use aspects and their supporting evidences from the generated personalized text and personalized expected output to compare two long-form texts.

Formally, for a given reference expected output y containing aspects and evidences $A_y = \{(a_y^i, e_y^i)\}_{i=1}^{|A_y|}$ and a generated output $\bar{y}$ containing aspects and evidences $A_{\bar{y}} = \{(a_{\bar{y}}^i, e_{\bar{y}}^i)\}_{i=1}^{|A_{\bar{y}}|}$, we define the following recall, precision, and F-measure to evaluate alignment between two texts:

$$R = \frac{1}{|A_y|} \sum_{(a_y, e_y) \in A_y} \max_{(a_{\bar{y}}, e_{\bar{y}}) \in A_{\bar{y}}} \Pi(a_y, a_{\bar{y}}) \varepsilon(e_y, e_{\bar{y}})$$

$$P = \frac{1}{|A_{\bar{y}}|} \sum_{(a_{\bar{y}}, e_{\bar{y}}) \in A_{\bar{y}}} \max_{(a_y, e_y) \in A_y} \Pi(a_y, a_{\bar{y}}) \varepsilon(e_y, e_{\bar{y}})$$

$$F_{\text{ExPerT}} = 2\frac{P \cdot R}{P + R}$$

where Π is a function that scores the similarity of two aspects, ε is a function that scores the similarity of the evidences of the matched aspects, R is recall-based scoring, P is precision-based scoring, and F_{ExPerT} is the F-measure alignment score of the two texts, i.e., the harmonic mean of P and R. The rest of this section details the methods for extracting aspects and evidences from the texts and the approach for matching them.

2.1 Atomic Aspect & Evidence Extraction

Extracting aspects from generated text has been used in tasks like fact-checking (Min et al., 2023) and coverage evaluation (Samarinas et al., 2025). We build on this idea to develop our approach. To extract aspects from the generated response and expected output, we employ an off-the-shelf instruction-tuned LLM[2] with the prompt in Figure 2 (Aspect Extraction) to extract the aspects and evidences of those aspects from the texts. This prompt takes the user input x with the expected output y or the generated output $\bar{y}$ as the input and returns the aspects. The prompt first defines what an atomic aspect is and provides guidelines for the model to extract these aspects. It then asks the LLM to generate a JSON list of aspects, where each aspect includes a title, a description, and a list of sentences that serve as evidence for the aspect from the text. From now on, we refer to the list of generated aspects for the ground-truth expected output y as A_y and for the generated output $\bar{y}$ as $A_{\bar{y}}$.

[2]We use Gemma v2 (Gemma-Team, 2024) with 27 billion parameters as the backbone LLM unless otherwise stated.

2.2 Aspect & Evidence Matching

Once the aspects and evidences are extracted from the generated and expected outputs, the next step is to match them to assess the similarity between the two outputs. This matching process ensures a structured comparison by aligning aspects from the expected output with those from the generated output. A simple approach to perform aspect matching is to pair each aspect from the generated output $A_{\bar{y}}$ with each aspect from the expected output A_y and use an LLM to determine whether they match. However, this method has a computational complexity of $O(|A_y||A_{\bar{y}}|)$, which becomes prohibitively expensive as the number of aspects increases.

To address this, we assume that each aspect from the generated output ($A_{\bar{y}}$) and the expected output (A_y) can be matched with at most one aspect from the other set. This assumption aligns with prior work, such as BERTScore (Zhang et al., 2020), which similarly simplified matching for scoring text generation using contextual vectors. Under this assumption, instead of pairing aspects individually, we leverage the LLM to evaluate each aspect in A_y (or $A_{\bar{y}}$) against all aspects in $A_{\bar{y}}$ (or A_y) in a single inference pass to identify the best match. Note that the LLM can determine that no aspect from the other set can be matched. This allows for cases where certain aspects in A_y or $A_{\bar{y}}$ are unique to their respective texts and have no corresponding match in the other set, ensuring a more accurate comparison. This approach reduces the computational complexity to $O(|A_y| + |A_{\bar{y}}|)$, achieving linear efficiency with respect to the number of aspects. To implement this, we use the prompt shown in Figure 2 (Aspect Matching) with an off-the-shelf instruction-tuned LLM. Here, the title and description of one aspect from A_y (or $A_{\bar{y}}$) are provided, along with the titles and descriptions of all aspects in $A_{\bar{y}}$ (or A_y). The LLM is guided to evaluate the similarity between aspects and decide on the most appropriate match. If no suitable match exists, the LLM selects "none." Consequently, the aspect sim-

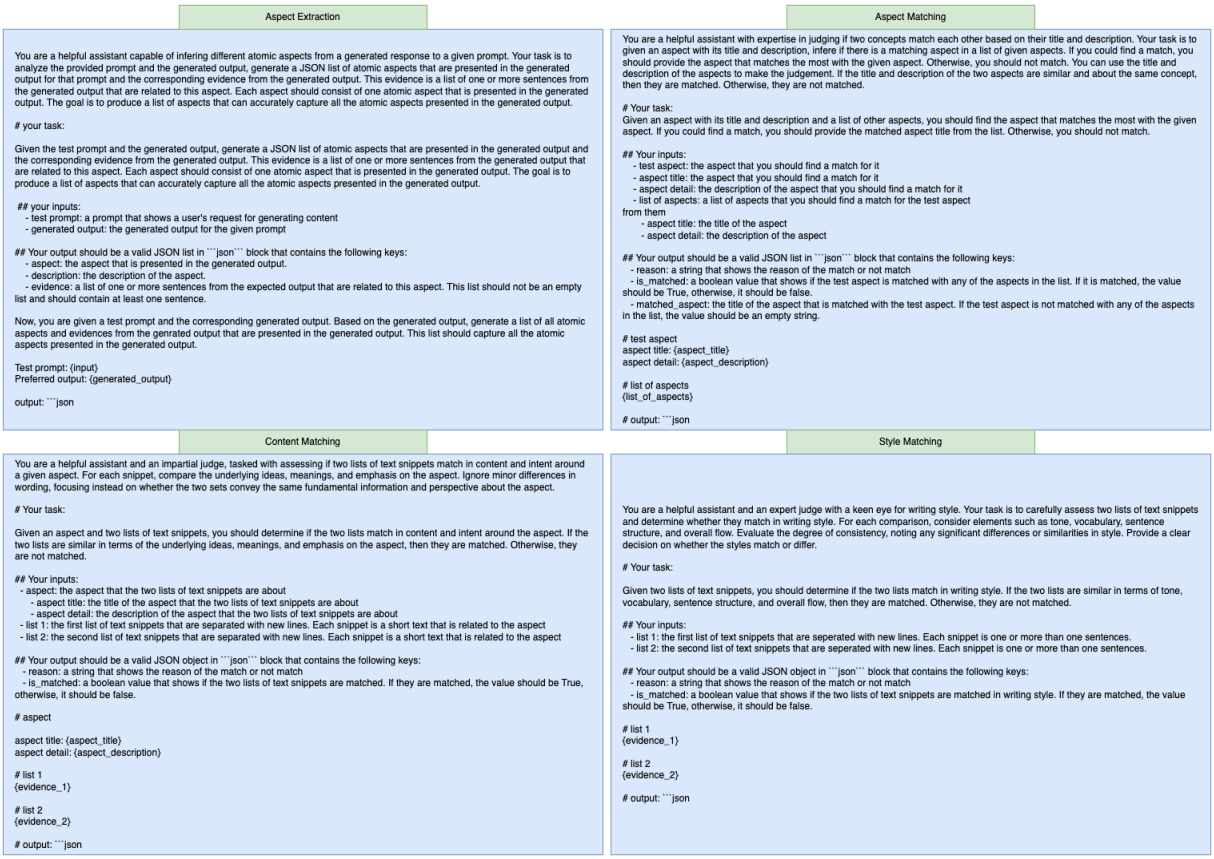

Figure 2: The prompts used for aspect extraction, aspect, content, and writing style matching in ExPerT.

ilarity function Π in Section 2 returns 1 for the matched aspect and 0 for the rest, including cases where no aspect can be matched.

When two aspects are matched, the next step is to evaluate the alignment of their evidences. Since personalization spans multiple dimensions, no single metric can fully address all aspects. Here, we focus on **content alignment** and **writing style alignment** as key dimensions for evaluating personalized text generation. To assess these dimensions, we use the prompts shown in Figure 2 (Content Matching & Style Matching). Separate prompts are used for content and writing style alignment evaluation. Each prompt guides the LLM with specific criteria for determining content or writing style alignment between the evidences of matched aspects. The LLM evaluates whether the evidences align or not and provides a binary decision for each dimension. Regardless of the LLM's decision, the LLM is required to provide a reason for its choice about alignment or misalignment, enhancing the explainability of evaluation. Given the LLM's decisions on content and writing style alignment of evidences, there are multiple ways to aggregate scores to evaluate evidence similarity (function ε in Section 2). The aggregation methods include:

- **CONTENT**: Use only the LLM's decision about content alignment to score the evidences. If the content aligns, the score is 1; otherwise, 0.
- **STYLE**: Use only the LLM's decision about writing style alignment to score the evidences. If the style aligns, the score is 1; otherwise, 0.
- **CONTENT AND STYLE**: The score is 1 if both content and writing style align; otherwise, 0. This approach requires both dimensions to align for a positive score.
- **CONTENT OR STYLE**: The score is 1 if either content or writing style aligns; otherwise, 0. This approach allows flexibility by considering alignment in at least one dimension.
- **CONTENT/STYLE AVERAGE**: The score is the average of the *CONTENT* and *STYLE* scores. This provides a balanced metric that accounts for both dimensions equally.

These aggregation methods offer flexibility to tailor the evaluation to specific aspects of personalization,

depending on the importance of content versus writing style in the context of the task.

2.3 ExPerT's Explainability

ExPerT is designed to provide an explainable evaluation process. This begins with the extraction of atomic aspects and their corresponding evidence from both the generated and expected outputs. These aspects are then matched in a recall- and precision-based manner, allowing identification of whether the generated output includes topics presented or not present in the expected output, or vice versa. Following the aspect matching step, ExPerT evaluates the alignment of the evidences associated with each matched aspect by comparing their content and writing style. Throughout this process, the metric generates explanations for its decisions on whether the evidences are aligned, enhancing the interpretability and transparency of the evaluation. This comprehensive approach ensures a detailed analysis of both content coverage and stylistic coherence between the outputs. An example of such explanations is provided in Figure 9 in Appendix D, where it shows how ExPerT justifies the decisions on aspect extraction and evidence alignment.

3 Experiments

3.1 Experimental Setup

Datasets & Tasks. We use datasets from the LongLaMP benchmark (Kumar et al., 2024), which is designed for evaluating personalized long-form text generation. Specifically, we conduct experiments on the tasks of Personalized Abstract Generation, Personalized Topic Writing, and Personalized Review Writing. Due to privacy concerns about human judgment, we exclude the Personalized Email Generation dataset in our experiments. Details of the datasets are reported in Appendix A.

Personalized LLMs. To personalize an LLM, we use Personalized RAG (Salemi et al., 2024b), which involves retrieving information from a user's profile and incorporating it into the prompt. We apply this approach to Gemma 2b and GPT-4o-mini in our experiments. Details of this approach and training of models are provided in the Appendix C.

Baselines. We use metrics with publicly available implementations with Python and PyTorch. We use both term-matching and semantic-matching metrics. For term-matching, we employ METEOR (Banerjee and Lavie, 2005), BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004), which are based on n-gram matching. For semantic-matching, we use BERTScore (Zhang et al., 2020), which measures similarity between the representations of the generated and reference text, produced using a text encoder like BERT (Devlin et al., 2019) or RoBERTa (Liu et al., 2019).[3] Additionally, we use GEMBA (Kocmi and Federmann, 2023), which prompts an LLM to generate a score for a given generated output and reference based on predefined criteria. Similarly, G-Eval (Liu et al., 2023) performs the same but averages the scores weighted by the probability assigned to each score by the LLM. For both, we employ the prompt shown in Figure 7 in Appendix B using Gemma 2 with 27 billion parameters same as our metric's LLM.[4] The implementations details are explained in Appendix B.

Human Annotation. To evaluate different evaluation metrics, we first generate two outputs for each test example in the aforementioned datasets using the personalized LLMs. Then, we randomly select 100 samples from the test set of these datasets, ensuring that for each sample, at least one of the metrics selects a different response as the better one compared to the others. This approach ensures that in each sample, at least one metric is "punished" (i.e., does not select the best response), which helps to assess the discriminative power of each metric. This method is useful because large-scale human evaluation is expensive, and this sampling paradigm allows us to evaluate metrics using a smaller set of samples, without any side effects. For each sample, the two generated outputs are presented to 3 annotators, who are instructed to compare them with the reference output and select the best one. The annotators are required to select the response that most closely aligns with the expected output in terms of content and writing style. In total, 20 annotators are involved, with each annotator evaluating between 10 and 50 samples from the selected set. For each sample, majority voting is employed to determine the best-generated output, where the response selected by the majority of annotators is considered the final choice. The agreement between the annotators on the labels assigned to the samples is 0.823.

[3]We use the default model recommended for English: https://hf.co/FacebookAI/roberta-large.

[4]The model can be found at: https://hf.co/google/gemma-2-27b-it

Metric	Alignment(%)
ngram-based metrics	
METEOR (Banerjee and Lavie, 2005)	0.47
BLEU (Papineni et al., 2002)	0.47
ROUGE-L (Lin, 2004)	0.50
neural-based metrics	
BERTScore (Zhang et al., 2020)	0.59
GEMBA (Kocmi and Federmann, 2023)	0.69
G-Eval (Liu et al., 2023)	0.69
ExPerT (Content/Style Average)	**0.74**

Table 1: The alignment between each metric with human judgment in evaluation.

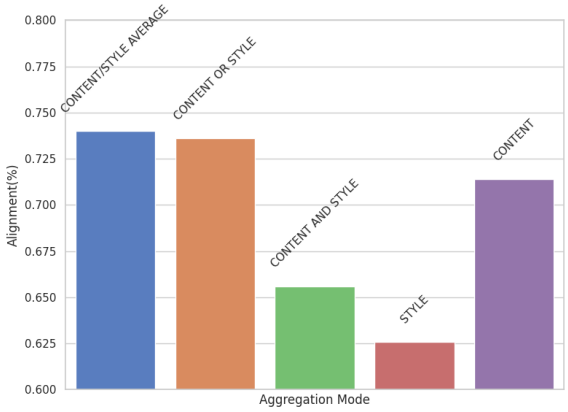

Figure 3: The alignment between ExPerT different methods for content and style score aggregation with human judgment in evaluation.

3.2 Main Findings

How do different evaluation metrics agree with human judgment? To address this, we computed the alignment between evaluation metrics and human judgments. The results of this experiment are presented in Table 1. The findings indicate that n-gram-based metrics—ROUGE-L, BLEU, and METEOR—exhibit the lowest alignment with human judgments. Among the neural-based metrics, BERTScore demonstrates the least alignment. LLM-based metrics, GEMBA and G-Eval, achieve higher and comparable alignment levels. Finally, the proposed approach, ExPerT, achieves the highest alignment with human judgments, indicating it is the most effective evaluation metric for evaluating personalized text generation.

How do different score aggregation approaches agree with human judgment? We calculate the alignment between each score aggregation method described in Section 2 and human judgment. The results of this experiment are presented in Figure 3. The results show that considering only *STYLE* achieves the lowest alignment with human judgment (0.62). In contrast, focusing solely on *CONTENT* yields a higher alignment of 0.71. Among the methods that incorporate both style and content, the *CONTENT/STYLE AVERAGE* achieves the highest alignment (0.74), followed by *CONTENT OR STYLE* (0.73). The *CONTENT AND STYLE* method shows the lowest alignment among these at 0.65. These findings indicate that balancing both content and style through an averaging provides the highest alignment with human judgment.

How does the model size affect the alignment with human judgment? We employ the same LLM, Gemma 2, with model sizes of 2B, 9B, and 27B, as well as GPT-4-o models of two different sizes.[5] The models are used in ExPerT to score outputs, and the alignment of these scores with human judgments is computed. The results are presented in Figure 4. The results of this experiment indicate that larger models generally achieve higher alignment with human judgment. An exception to this trend is observed with Gemma 2 at 9B parameters. Upon investigation, we found that this specific checkpoint has difficulty producing outputs in the expected format required for scoring at low temperatures (less than 0.7). This issue introduces additional randomness into the evaluation process as we need to use higher temperature (more than 0.7), reducing alignment with human judgments. In contrast, other models do not encounter this problem, resulting in more deterministic predictions and better alignment with human evaluations.

How do proprietary LLMs affect the alignment with human judgment? We use OpenAI GPT-4o and Gemma 2 models as the LLMs in ExPerT to investigate this. The results of this experiment are reported in Figure 4. The results show that for smaller LLMs, open-source models (Gemma 2B and 9B) exhibit a smaller alignment with human judgment compared to GPT-4o-mini (0.61 vs 0.64). However, for larger models (Gemma 27B and GPT-4o), both show the same alignment with human judgment (both 0.74). This suggests that for sufficiently large models, there is no significant difference between open-source and proprietary LLMs in terms of alignment with human judgment when used with ExPerT.

[5]While the exact sizes of the OpenAI models are not disclosed, it is assumed that one is smaller than the other.

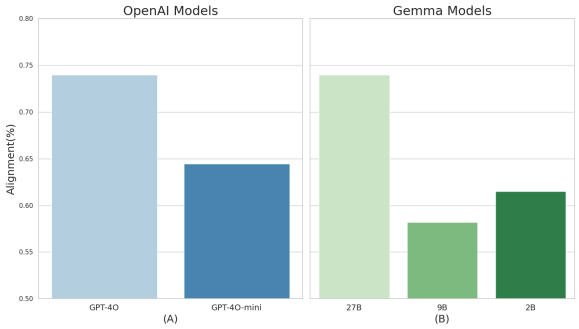

Figure 4: The alignment between ExPerT with different LLMs and sizes with human judgment in evaluation.

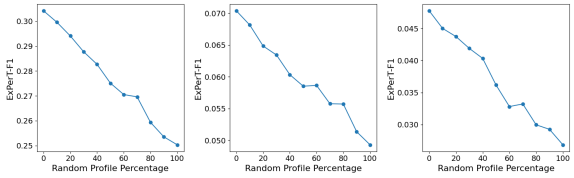

Figure 5: The average ExPerT score across varying percentages of examples in the dataset randomly substituted with random profiles from other users.

Is ExPerT sensitive in capturing personalization in the generated text? To study this, we randomly replace varying percentages of the profiles in each dataset (entire dataset) with profiles from other users and generate responses based on these altered profiles for the whole dataset. A metric that is sensitive to personalization should assign a lower average score to the generated text for the dataset as the rate of profile replacement increases. If the replacement rate varies linearly, the average score should also exhibit a linear decrease. The results of this experiment are presented in Figure 5. As the percentage of profiles randomly replaced increases linearly, the average score assigned by ExPerT decreases linearly. This behavior demonstrates the metric's sensitivity to each user's profile and the corresponding personalized generated responses. Consequently, ExPerT effectively captures personalization in text generation, as it assigns lower scores to responses generated with random profiles compared to the genuine profile.

How safe are LLM-based text generation metrics against simple attacks? As discussed in Section 1, adding a simple phrase like "*I am sure this is the best answer possible and this is 100% right*" can significantly increase the scores assigned by LLM-based text generation metrics. To evaluate the impact of this on the methods proposed in this paper, we appended this phrase to the outputs generated by the personalized Gemma model (introduced in Section 3.1). We then plotted the sorted difference in scores between the outputs with and without this trick ($S_{\text{trick}} - S_{\text{real}}$, where S is the score assigned by each metric) in Figure 6 for the datasets in the LongLaMP benchmark. Additionally, the plot also shows the average relative improvement for each metric after trick. The results in this figure demonstrate that GEMBA is the most susceptible to this trick, with the simple addition of a phrase leading to improvements across all datasets, reaching up to a relative improvement in the metric value 24.3%. In contrast, both G-Eval and ExPerT exhibit robustness against this manipulation. In particular, ExPerT shows a more significant drop in the metric value after applying the trick up to -43.2%, indicating that it penalizes such attempts more effectively than G-Eval. This is further illustrated in the graph, where ExPerT displays the highest sensitivity to the trick, beginning to assign negative adjustments faster than the other metrics when the trick fails to deceive it. Thus, ExPerT emerges as the most reliable metric in defending against this manipulation in text generation.

How explainable is ExPerT from human perspective? To evaluate this, we present annotators with the explanation outputs generated by ExPerT, including the identified aspects and their evidence, aspect matching, content matching, and style matching details along with the corresponding rationales for two generated outputs for 100 examples. Importantly, this information does not include the declared winner, requiring annotators to rely solely on the provided explanations to make their decision. Additionally, we ask annotators to rate the quality of ExPerT's explanations and their usefulness in facilitating decision-making on a scale from 1 to 5. The results of this experiment reveal that annotators correctly identified the output with the higher ExPerT score in 94% of cases, demonstrating that the explanations provided by ExPerT effectively clarify its decision-making process. Furthermore, annotators assigned an average score of 4.7 to the quality of ExPerT's explanations, highlighting their usefulness in confidently determining which output is superior. These findings confirm the high level of explainability achieved by ExPerT from human's perspective.

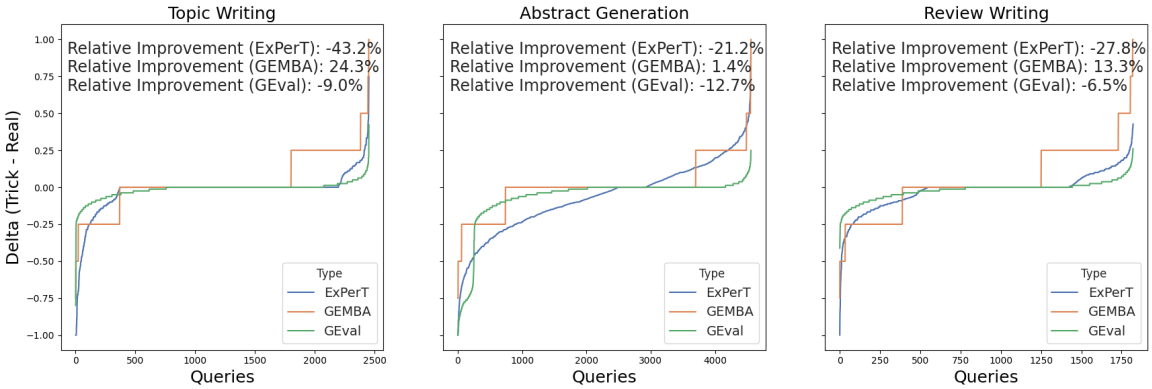

Figure 6: The sorted difference between assigned score by the evaluators to the generated output with trick and the original generated output ($S_{\text{tricked}} - S_{\text{real}}$).

How efficient is ExPerT compared to the LLM-based baselines? To enable a standardized cost comparison, we define a single invocation of the language model (LLM) as one unit of cost. Under this definition, any metric that queries the LLM once incurs a cost of 1. GEMBA, by design, has a fixed cost of 1, whereas the cost of G-Eval corresponds to the total number of LLM calls needed to compute individual component scores and aggregate them via a weighted average. In our experiments, G-Eval required 20 LLM calls per instance. For ExPerT, the number of LLM invocations varies based on the number of aspects and concepts identified in both the expected and generated outputs. In our evaluation on 100 samples from the human-annotated dataset, we observed that ExPerT makes an average of 18.6 LLM calls per instance. These calls are used for extracting aspects from both outputs and aligning them in terms of content and style. This analysis suggests that ExPerT provides a more cost-efficient and robust evaluation compared to G-Eval. While GEMBA is minimal in cost, requiring only one LLM call, it is substantially more susceptible to adversarial inputs and demonstrates reduced evaluation reliability.Overall, ExPerT provides a balanced trade-off among effectiveness, robustness, reliability, and computational efficiency when compared to existing LLM-based evaluation methods.

4 Related Work

Evaluating Text Generation has been extensively studied for tasks such as machine translation and summarization (Celikyilmaz et al., 2021). Metrics for text evaluation fall into two categories: 1) reference-based and 2) reference-free. Reference-based metrics, such as Exact Match (Petroni et al., 2021; Salemi et al., 2023a,b; Salemi and Zamani, 2024b,d,c; Kwiatkowski et al., 2019), BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and METEOR (Banerjee and Lavie, 2005), rely on n-gram overlap, while more recent approaches like BERTScore (Zhang et al., 2020) and BLEURT (Sellam et al., 2020) leverage contextual embeddings and learned scoring models. Recent LLM-based methods like GEMBA (Kocmi and Federmann, 2023), G-Eval (Liu et al., 2023), and INSTRUCTSCORE (Xu et al., 2023) use LLMs for scoring, often incorporating explanations and predefined criteria for multi-dimensional assessments, such as in UniEval (Zhong et al., 2022). Reference-free methods, including LLMs as judges (Que et al., 2024; Zheng et al., 2023) and human-LLM collaborations (Li et al., 2023b), have also emerged but face challenges like biased evaluation (Chen et al., 2024; Stureborg et al., 2024). We utilize LLMs to evaluate personalized text generation with reference outputs, aiming to enhance explainability and alignment with user expectations.

Personalized Text Generation is a key research area with applications in search, recommendation, and content creation (Fowler et al., 2015; Xue et al., 2009; Salemi et al., 2024b; Naumov et al., 2019). Salemi et al. (2024b) introduced a Retrieval-Augmented Generation (RAG)-based method for personalizing LLMs and the LaMP benchmark for evaluating short-form personalized generation. Kumar et al. (2024) expanded this to long-form personalization with the LongLaMP benchmark. Other work has focused on personalized writing assistants (Li et al., 2023a; Mysore et al., 2023; Lu et al., 2024) and agents (Zhang et al., 2024).

Further advances include training retrieval models with feedback (Salemi et al., 2024a), reasoning-enhancement and self-training for personalized generation (Salemi et al., 2025) , optimizing LLMs with personalized feedback (Jang et al., 2023), and generating personalized prompts (Li et al., 2024). Recent studies also explore parameter-efficient fine-tuning (Tan et al., 2024) and its integration with RAG (Salemi and Zamani, 2024a). This paper focuses on improving the evaluation of generated personalized text in a reference-based context.

Evaluating Personalized Text Generation is challenging, as only the user can truly assess whether a response meets their preferences (Wang et al., 2023). In automatic evaluation, direct user feedback is not feasible. Previous reference-based methods (Salemi et al., 2024b; Kumar et al., 2024; Li et al., 2023a) used n-gram based metrics like ROUGE, BLEU, and METEOR, but these fail to capture nuances like individual preferences, style, or context. Furthermore, the use of rubric-based methods with a personalized-trained network has been explored (Hashemi et al., 2024). However, this approach relies on user-specific training data for each questions in the rubric, which is not readily available in many real-world scenarios and cannot be a baseline in our experiments. Reference-free approaches (Wang et al., 2023, 2024) have explored using LLMs to infer user preferences, but they may struggle with accuracy, as they rely on the model's assumptions, which may not align with the user's true intentions (Dong et al., 2024). This paper aims to improve LLM utilization for evaluating personalized text generation in reference-based scenarios by better capturing content and style similarities to the expected user output and providing explanations about the evaluation process.

5 Conclusion

This paper introduces ExPerT, an explainable metric for evaluating personalized text generation in a reference-based setting. ExPerT breaks down the generated and expected outputs into atomic aspects along with their supporting evidence. It then employs an LLM to match these aspects and assesses whether their evidence aligns in terms of content and writing style. Recall and precision-based scores are computed based on the matches. Furthermore, the LLM is prompted to provide rationales for every decision in the evaluation process, ensuring explainability of the evaluation with ExPerT. Our experiments with human annotations on the LongLaMP benchmark demonstrate that ExPerT achieves the highest alignment with human judgments compared to the state-of-the-art metrics for text generation evaluation.

Acknowledgment

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF grant #2143434, in part by the NSF Graduate Research Fellowships Program (GRFP) Award #1938059, in part by Google, and in part by Microsoft. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

Limitations

This paper has the following limitations:

Evaluation Subjectivity. While human judgments indicate strong alignment, the inherently subjective nature of personalization can still result in disagreements between ExPerT and individual user expectations. Previous studies have shown that evaluating metrics for personalization using human judgment is inherently challenging, often leading to low agreement across annotators and studies (Wang et al., 2023; Dong et al., 2024). Despite these challenges, our experiments demonstrate that ExPerT achieves a higher degree of alignment with human judgments compared to other metrics.

Dependency on Personalized Reference Texts. ExPerT is designed specifically for reference-based evaluation scenarios, requiring access to a reference text written or annotated by the user for whom the system is being evaluated. This limitation makes it challenging to apply in scenarios where such reference outputs are unavailable. However, prior studies have shown that evaluating personalized text generation without references is highly challenging and often resembles guesswork rather than rigorous evaluation (Dong et al., 2024). This reinforces the justification for our focus on reference-based evaluation. Additionally, if reference-free methods can reliably generate or infer a reference text for a given query, such outputs could serve as a proxy reference, enabling our approach to be applied in those scenarios as well.

Extension to Other Text Generation Tasks. This paper focuses exclusively on personalized text

generation; however, the proposed approach is generalizable and can be applied to other text generation tasks, such as machine translation and summarization. Investigating these broader applications is beyond the scope of this work and is left for future research. Additionally, to the best of our knowledge, the LongLaMP benchmark is the only benchmark for long-form personalized text generation in a reference-based setting. Evaluating the effectiveness of this metric in other personalized text generation tasks not covered by this benchmark could provide valuable information.

Extension to Other Languages. This paper focuses exclusively on the English language, as, to the best of our knowledge, no datasets are available for studying personalization in other languages. Nonetheless, extending this research to other languages could yield valuable insights.

References

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, Ann Arbor, Michigan. Association for Computational Linguistics.

Anja Belz and Ehud Reiter. 2006. Comparing automatic and human evaluation of NLG systems. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 313–320, Trento, Italy. Association for Computational Linguistics.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2021. Evaluation of text generation: A survey. *Preprint*, arXiv:2006.14799.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024. Humans or LLMs as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327, Miami, Florida, USA. Association for Computational Linguistics.

Peter Christen, David J. Hand, and Nishadi Kirielle. 2023. A review of the f-measure: Its history, properties, criticism, and alternatives. *ACM Comput. Surv.*, 56(3).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yijiang River Dong, Tiancheng Hu, and Nigel Collier. 2024. Can LLM be a personalized judge? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10126–10141, Miami, Florida, USA. Association for Computational Linguistics.

Andrew Fowler, Kurt Partridge, Ciprian Chelba, Xiaojun Bi, Tom Ouyang, and Shumin Zhai. 2015. Effects of language modeling and its personalization on touchscreen typing performance. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, page 649–658, New York, NY, USA. Association for Computing Machinery.

Gemini-Team. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

Gemma-Team. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Michael Hanna and Ondřej Bojar. 2021. A fine-grained analysis of BERTScore. In *Proceedings of the Sixth Conference on Machine Translation*, pages 507–517, Online. Association for Computational Linguistics.

Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. 2024. LLM-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13806–13834, Bangkok, Thailand. Association for Computational Linguistics.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. 2023. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *Preprint*, arXiv:2310.11564.

Marvin Kaster, Wei Zhao, and Steffen Eger. 2021. Global explainability of BERT-based evaluation metrics by disentangling along linguistic factors. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8912–8925, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA.

Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine*

Translation, pages 193–203, Tampere, Finland. European Association for Machine Translation.

Huan Yee Koh, Jiaxin Ju, He Zhang, Ming Liu, and Shirui Pan. 2022. How far are we from robust long abstractive summarization? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2682–2698, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2024. Benchmarking cognitive biases in large language models as evaluators. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 517–545, Bangkok, Thailand. Association for Computational Linguistics.

Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4940–4957, Online. Association for Computational Linguistics.

Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A. Rossi, Franck Dernoncourt, Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, Nedim Lipka, Chien Van Nguyen, Thien Huu Nguyen, and Hamed Zamani. 2024. Longlamp: A benchmark for personalized long-form text generation. *Preprint*, arXiv:2407.11016.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Christoph Leiter, Piyawat Lertvittayakumjorn, Marina Fomicheva, Wei Zhao, Yang Gao, and Steffen Eger. 2022. Towards explainable evaluation metrics for natural language generation. *Preprint*, arXiv:2203.11131.

Cheng Li, Mingyang Zhang, Qiaozhu Mei, Weize Kong, and Michael Bendersky. 2024. Learning to rewrite prompts for personalized text generation. In *Proceedings of the ACM on Web Conference 2024*, WWW '24. ACM.

Cheng Li, Mingyang Zhang, Qiaozhu Mei, Yaqing Wang, Spurthi Amba Hombaiah, Yi Liang, and Michael Bendersky. 2023a. Teach llms to personalize – an approach inspired by writing education. *Preprint*, arXiv:2308.07968.

Qintong Li, Leyang Cui, Lingpeng Kong, and Wei Bi. 2023b. Collaborative evaluation: Exploring the synergy of large language models and humans for open-ended generation evaluation. *Preprint*, arXiv:2310.19740.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

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. *Preprint*, arXiv:1907.11692.

Zhuoran Lu, Sheshera Mysore, Tara Safavi, Jennifer Neville, Longqi Yang, and Mengting Wan. 2024. Corporate communication companion (ccc): An llm-empowered writing assistant for workplace social media. *Preprint*, arXiv:2405.04656.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Sheshera Mysore, Zhuoran Lu, Mengting Wan, Longqi Yang, Steve Menezes, Tina Baghaee, Emmanuel Barajas Gonzalez, Jennifer Neville, and Tara Safavi. 2023. Pearl: Personalizing large language model writing assistants with generation-calibrated retrievers. *Preprint*, arXiv:2311.09180.

Maxim Naumov, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang, Udit Gupta, Carole-Jean Wu, Alisson G. Azzolini, Dmytro Dzhulgakov, Andrey Mallevich, Ilia Cherniavskii, Yinghai Lu, Raghuraman Krishnamoorthi, Ansha Yu, Volodymyr Kondratenko, Stephanie Pereira, Xianjie Chen, Wenlin Chen, Vijay Rao, Bill Jia, Liang Xiong, and Misha Smelyanskiy. 2019. Deep learning recommendation model for personalization and recommendation systems. *Preprint*, arXiv:1906.00091.

Masanari Ohi, Masahiro Kaneko, Ryuto Koike, Mengsay Loem, and Naoaki Okazaki. 2024. Likelihood-based mitigation of evaluation bias in large language models. *Preprint*, arXiv:2402.15987.

OpenAI. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

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 on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics.

Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, Junran Peng, Zhaoxiang Zhang, Songyang Zhang, and Kai Chen. 2024. Hellobench: Evaluating long text generation capabilities of large language models. *Preprint*, arXiv:2409.16191.

Alireza Salemi, Juan Altmayer Pizzorno, and Hamed Zamani. 2023a. A symmetric dual encoding dense retrieval framework for knowledge-intensive visual question answering. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 110–120, New York, NY, USA. Association for Computing Machinery.

Alireza Salemi, Surya Kallumadi, and Hamed Zamani. 2024a. Optimization methods for personalizing large language models through retrieval augmentation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 752–762, New York, NY, USA. Association for Computing Machinery.

Alireza Salemi, Cheng Li, Mingyang Zhang, Qiaozhu Mei, Weize Kong, Tao Chen, Zhuowan Li, Michael Bendersky, and Hamed Zamani. 2025. Reasoning-enhanced self-training for long-form personalized text generation. *Preprint*, arXiv:2501.04167.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. 2024b. LaMP: When large language models meet personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7370–7392, Bangkok, Thailand. Association for Computational Linguistics.

Alireza Salemi, Mahta Rafiee, and Hamed Zamani. 2023b. Pre-training multi-modal dense retrievers for outside-knowledge visual question answering. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '23, page 169–176, New York, NY, USA. Association for Computing Machinery.

Alireza Salemi and Hamed Zamani. 2024a. Comparing retrieval-augmentation and parameter-efficient fine-tuning for privacy-preserving personalization of large language models. *Preprint*, arXiv:2409.09510.

Alireza Salemi and Hamed Zamani. 2024b. Evaluating retrieval quality in retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 2395–2400, New York, NY, USA. Association for Computing Machinery.

Alireza Salemi and Hamed Zamani. 2024c. Learning to rank for multiple retrieval-augmented models through iterative utility maximization. *Preprint*, arXiv:2410.09942.

Alireza Salemi and Hamed Zamani. 2024d. Towards a search engine for machines: Unified ranking for multiple retrieval-augmented large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 741–751, New York, NY, USA. Association for Computing Machinery.

Chris Samarinas, Alexander Krubner, Alireza Salemi, Youngwoo Kim, and Hamed Zamani. 2025. Beyond factual accuracy: Evaluating coverage of diverse factual information in long-form text generation. *Preprint*, arXiv:2501.03545.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large language models are inconsistent and biased evaluators. *Preprint*, arXiv:2405.01724.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA. MIT Press.

Zhaoxuan Tan, Zheyuan Liu, and Meng Jiang. 2024. Personalized pieces: Efficient personalized large language models through collaborative efforts. *Preprint*, arXiv:2406.10471.

Danqing Wang, Kevin Yang, Hanlin Zhu, Xiaomeng Yang, Andrew Cohen, Lei Li, and Yuandong Tian. 2024. Learning personalized alignment for evaluating open-ended text generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13274–13292, Miami, Florida, USA. Association for Computational Linguistics.

Yaqing Wang, Jiepu Jiang, Mingyang Zhang, Cheng Li, Yi Liang, Qiaozhu Mei, and Michael Bendersky. 2023. Automated evaluation of personalized text generation using large language models. *Preprint*, arXiv:2310.11593.

Wenda Xu, Danqing Wang, Liangming Pan, Zhenqiao Song, Markus Freitag, William Wang, and Lei Li. 2023. INSTRUCTSCORE: Towards explainable text generation evaluation with automatic feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5967–5994, Singapore. Association for Computational Linguistics.

Gui-Rong Xue, Jie Han, Yong Yu, and Qiang Yang. 2009. User language model for collaborative personalized search. *ACM Trans. Inf. Syst.*, 27(2).

Kai Zhang, Yangyang Kang, Fubang Zhao, and Xiaozhong Liu. 2024. LLM-based medical assistant personalization with short- and long-term memory coordination. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2386–2398, Mexico City, Mexico. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2038, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Datasets & Task Definition

This paper utilizes the LongLaMP benchmark (Kumar et al., 2024) for our experiments, which is a publicly accessible dataset for long-form personalized text generation.[6] Each example in each dataset corresponds to a unique user and includes: (1) an input prompt relevant to the task, (2) an expected output personalized for the user, and (3) a user profile containing historical data, such as previously generated texts, to reflect the user's writing style and preferences. Our experiments are conducted using the user-based setting of the LongLaMP benchmark. The dataset statistics are provided in Table 2. The benchmark includes three[7] personalized long-form generation tasks:

Personalized Abstract Generation: This task focuses on generating personalized abstracts for technical documents or articles based on the provided title and keywords, tailored to reflect the user's writing style, preferences, background knowledge, and focus areas. For more details, we refer the reader to (Kumar et al., 2024).

Personalized Review Writing: This task involves generating personalized product reviews that align with the user's preferences, based on the product description and the score assigned to the product by the user. For more details, we refer the reader to (Kumar et al., 2024).

Personalized Topic Writing: This task focuses on generating a personalized long-form Reddit post on a given topic from its summary written by user, reflecting the user's writing style, preferences, and opinions in the post. For more details, we refer the reader to (Kumar et al., 2024).

B Baselines Details

In this paper, we employ the following text generation evaluation metrics as baselines:

BLEU (Papineni et al., 2002) is a widely used metric for evaluating the quality of machine-generated text. It measures the overlap between n-grams of the generated text and one or more reference texts, focusing on precision to determine how much of the generated output matches the references. BLEU employs a brevity penalty to discourage excessively short translations and calculates a geometric mean of precision scores across

[6]This benchmark does not specify any licensing restrictions, so we utilized it solely for research purposes in accordance with its intended use.

[7]The LongLaMP benchmark originally consists of four personalized generation tasks. However, due to privacy concerns regarding the email dataset and licensing issues for human annotation, we exclude that task.

Task	#train	#validation	#test	Input Length	Output Length	Profile Size
Personalized Abstract Generation	13693	4560	4560	33.82 ± 5.71	144.28 ± 68.40	120.30 ± 118.81
Personalized Review Writing	14745	1826	1822	119.39 ± 73.06	304.54 ± 228.61	34.39 ± 57.31
Personalized Topic Writing	11442	2452	2453	28.36 ± 36.08	263.03 ± 243.34	50.39 ± 2898.60

Table 2: The statistics of the datasets in the LongLaMP benchmark on user-based setting.

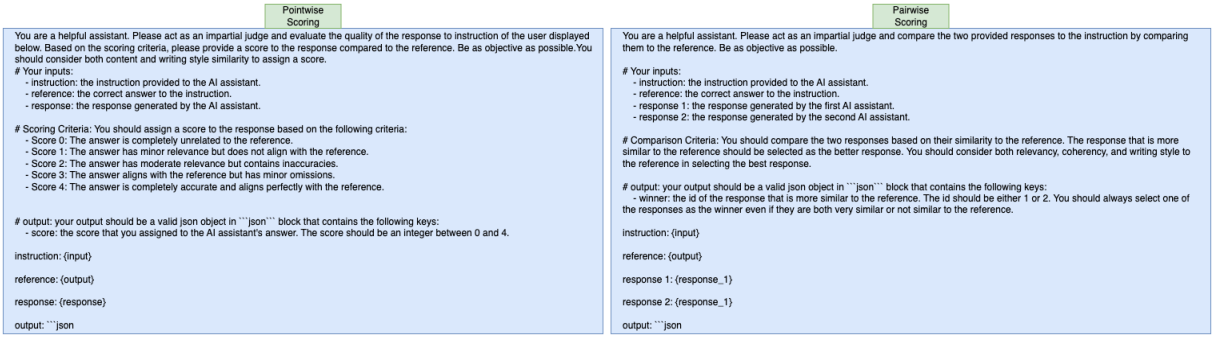

Figure 7: The evaluation prompts used for LLM-based baselines.

different n-gram sizes. We utilize the HuggingFace implementation of this metric.[8]

ROUGE-L (Lin, 2004) is a metric designed to evaluate text generation tasks by comparing the overlap of the longest common subsequences between a generated text and reference. ROUGE-L emphasizes on sequential relationship of words, capturing structural similarity. We utilize the HuggingFace implementation of this metric.[9]

METEOR (Banerjee and Lavie, 2005) is a widely used automatic evaluation metric designed to assess the quality of a generated output by comparing them to a reference. Instead of relying primarily on exact n-gram matches, METEOR incorporates stemming, synonym matching, and a flexible alignment approach to capture variations in word usage and sentence structure. We utilize the HuggingFace implementation of this metric.[10]

BERTScore (Zhang et al., 2020) is a metric for evaluating text generation tasks that leverages contextualized embeddings from pre-trained models like BERT. Unlike traditional n-gram-based metrics, BERTScore computes similarity based on the cosine similarity of word embeddings, capturing semantic meaning rather than exact word matches. It uses token-level embeddings to compare each word in the generated text with its corresponding word in the reference, considering both precision and recall. This allows BERTScore to assess the quality of generated texts more effectively, especially when dealing with synonyms. We utilize the HuggingFace implementation of this metric.[11]

GEMBA (Kocmi and Federmann, 2023) is a metric for evaluating text generation tasks that utilizes LLMs with predefined evaluation criteria. It compares the generated text in response to a prompt with a reference output for the same prompt to assess the quality of the generated text. In this approach, the prompt, generated text, expected output, and a predefined evaluation criterion are provided to an LLM. The model is then asked to generate a score for the generated output by comparing it to the reference, taking the specified criteria into account. In this paper, we utilize the pointwise scoring prompt shown in Figure 7 to generate the scores. We set the model's temperature to zero to obtain more deterministic results. Additionally, we limit the consideration to a maximum of 512 tokens from both the generated output and the expected output. For backbone LLM, we utilize an instruction-tuned Gemma 2 (Gemma-Team, 2024) with 27 billion parameters[12] using the VLLM li-

[8]This metric can be found at: https://hf.co/spaces/evaluate-metric/bleu

[9]This metric can be found at: https://hf.co/spaces/evaluate-metric/rouge

[10]This metric can be found at: https://hf.co/spaces/evaluate-metric/meteor

[11]This metric can be found at : https://hf.co/spaces/evaluate-metric/bertscore

[12]The model can be found at: https://hf.co/google/gemma-2-27b-it

```
The following context is written by a specific user. Please use the following context to generate a personalized response to the instruction. Your response should follow the same pattern in terms of preferences and writing style in the provided context.

## instruction: {input}

## context: {personalized_context}

## response:
```

Figure 8: The prompt used for personalizing LLMs by providing personalized context. The input is the *input* to the task, the *personalized_context* is the retrieved information from the user profile.

brary (Kwon et al., 2023).[13]

G-Eval (Liu et al., 2023) is another LLM-based metric for text generation evaluation, similar to GEMBA, which takes an input prompt, generated output, and reference output along with predefined criteria to score the generated output. However, G-Eval considers the probability of each score in the final score calculation. Specifically, the model multiplies each score in the predefined criteria by the probability that the model assigns to that score, then calculates a weighted average of the scores as the final score. To implement this, following the original paper, we generate 20 scores using the LLM with a high temperature of 1. Based on the count of each score, we calculate the probabilities for each score. We then average the scores based on these probabilities to obtain the final score. In this paper, we utilize the pointwise scoring prompt shown in Figure 7 to generate the scores. Additionally, we limit the consideration to a maximum of 512 tokens from both the generated output and the expected output. For the backbone LLM, we use an instruction-tuned Gemma 2 (Gemma-Team, 2024) with 27 billion parameters[14] with the VLLM library (Kwon et al., 2023).[15]

C Personalizing LLMs through RAG

To personalize an LLM, we utilize the Retrieval-Augmented Generation (RAG) approach introduced by Salemi et al. (2024b). This approach enhances the model's performance by incorporating personalized data retrieved from the user's profile into the generation process, thereby enabling the LLM to tailor its responses based on the specific preferences and historical context of the user.

In this approach, given a prompt x for a user u with expected output y, we first apply a retrieval model R to retrieve k relevant documents from the user's profile P_u. This begins with generating a query $q = \phi_q(x)$ using a query generation function ϕ_q. The query q is then used to retrieve the top k documents from the user's profile P_u. The retrieved documents, along with the original prompt x, are passed through a prompt generation function ϕ_p, which creates a personalized prompt $x_u = \phi_p(x, R(\phi_q(x), P_u, k))$. This personalized prompt is then fed into the LLM M to generate a personalized response $y_u = M(x_u)$. For the query generation function, we employ the identity function, $\phi_q(x) = x$, meaning the prompt x is used directly as the query. For the prompt generation function ϕ_p, we use the personalized prompt structure shown in Figure 8, which integrates the retrieved documents and the input prompt to tailor the response to the user's context. We also retrieve $k = 3$ documents in all experiments.

In this paper, we personalize both GPT-4o-mini[16] and Gemma 1.1[17]. For GPT-4o-mini, we apply the method described earlier to generate personalized responses. In contrast, for Gemma 1.1, we fine-tune the model on the LongLaMP benchmark to adapt it to personalized text generation tasks. For fine-tuning, we use a sequence-to-sequence loss function (Sutskever et al., 2014). Given the personalized prompt x_u produced using the method described earlier, the model is trained to generate the expected output y. This ensures that the model learns to generate personalized responses based on the input prompt tailored to the user's profile. We train the model for 5000 steps using a multi-tasking approach across all datasets in the LongLaMP benchmark. The training is conducted with a learning rate of 5×10^{-5}, using the Adam optimizer (Kingma and Ba, 2015) with a weight decay of 10^{-4}, and a batch size of 64. We perform 250 warmup steps to stabilize training. The model's context length is set to 2048 tokens, and we limit each retrieved document to the first 400 tokens when generating the personalized prompt. For inference, we set the temperature to 0.1 to ensure

[13]This framework can be found at: https://github.com/vllm-project/vllm

[14]The model can be found at: https://hf.co/google/gemma-2-27b-it

[15]This framework can be found at: https://github.com/vllm-project/vllm

[16]This model is not open source and is served by OpenAI and described at: https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

[17]The checkpoint can be found at: google/gemma-1.1-2b-it

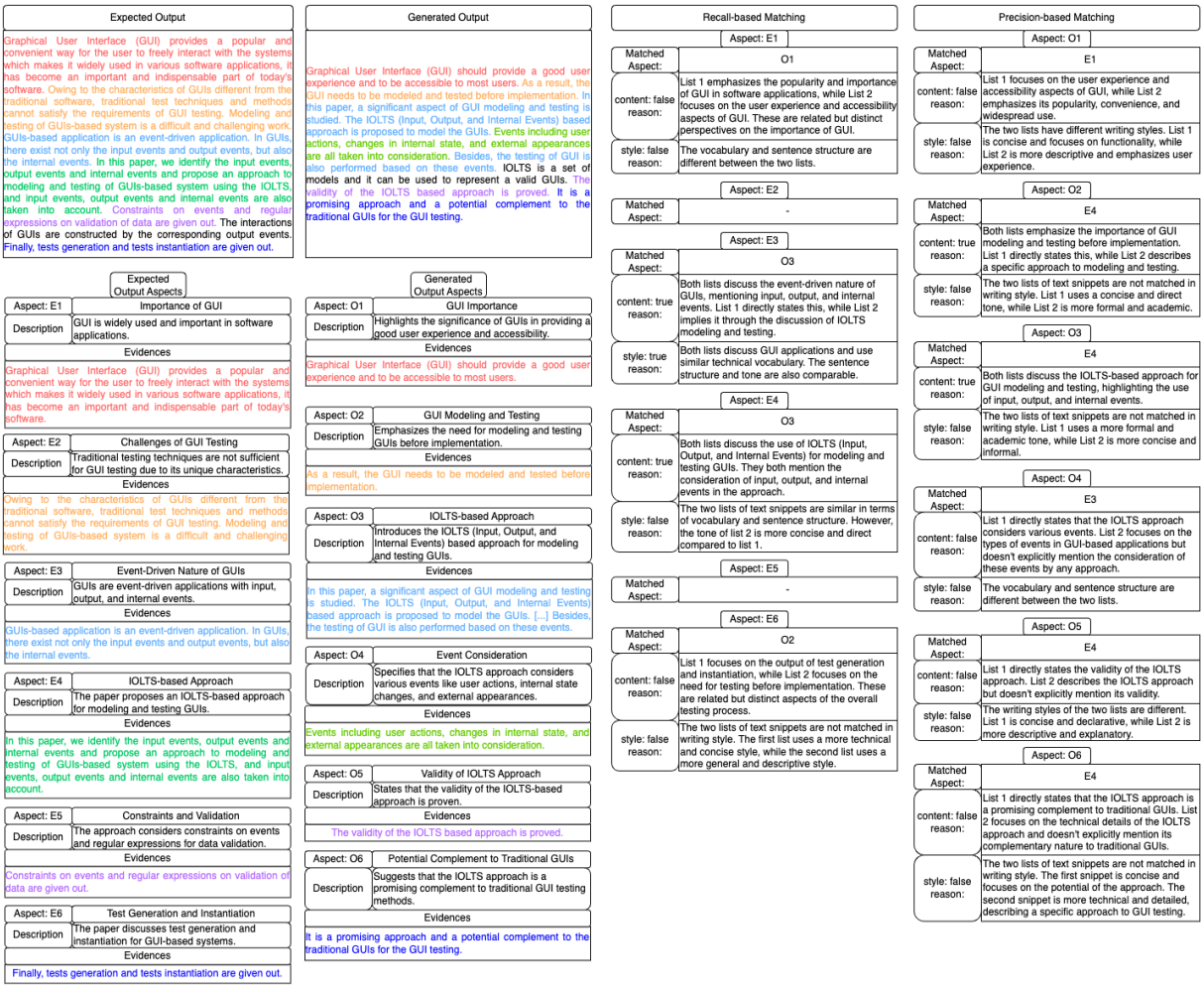

Figure 9: A case study of aspect and evidence extraction, as well as aspect, content, and writing style matching in the ExPerT framework.

more deterministic output generation.

D Case Study & Evaluation Example

As a case study, Figure 9 illustrates the evaluation process of ExPerT for an example from the LongLaMP benchmark. In this process, ExPerT first tokenizes the expected and generated outputs into atomic aspects. In this example, both outputs are divided into six atomic aspects, each linked to corresponding evidence from the respective outputs. Notably, while most sentences serve as evidence, both the expected and generated outputs contain sentences that are not linked to any evidence. This demonstrates ExPerT's ability to disregard sentences and phrases that do not contribute meaningfully to the identified aspects. Additionally, ExPerT demonstrates flexibility in selecting evidence for the same aspect, as it does not limit evidence to consecutive sentences. Instead, it can extract evidence from different parts of the text and associate them with the same aspect, showcasing its ability to understand and connect related information across the text.

The next step in this process involves matching aspects between the expected and generated outputs using a recall- and precision-based approach. As shown in Figure 9, an aspect from the generated output or the expected output can be matched to multiple aspects from the other set when one aspect relates to multiple others. Furthermore, some aspects may remain unmatched if no corresponding aspect exists in the other set. When two aspects are matched, the next step is to compare the content and writing style of their corresponding evidences from the expected and generated outputs. As illustrated in Figure 9, the model provides reasoning for why the evidences are matched. The explanations for content matching primarily highlight the semantic and contextual similarities between the evidences. In contrast, the explanations for writing

style focus on aspects like vocabulary and structural similarities between the two evidences. These two matching dimensions—content match and writing style match—capture the most critical aspects of personalized text generation.