Comparative Personalization for Multi-document Summarization

Anonymous ACL submission

Abstract

Personalized multi-document summarization (MDS) is essential for meeting individual user preferences of writing style and content focus for summaries. In this paper, we propose that for effective personalization, it is important to identify fine-grained differences between users' preferences by comparing the given user's preferences with other users' preferences. Motivated by this, we propose `ComPSum`, a personalized MDS framework. It first generates a structured analysis of a user by comparing their preferences with other users' preferences. The generated structured analysis is then used to guide the generation of personalized summaries. To evaluate the performance of `ComPSum`, we propose `AuthorMap`, a fine-grained reference-free evaluation framework for personalized MDS. It evaluates the personalization of a system based on the authorship attribution between two personalized summaries generated for different users. For robust evaluation of personalized MDS, we construct `PerMSum`, a personalized MDS dataset in the review and news domain. We evaluate the performance of `ComPSum` on `PerMSum` using `AuthorMap`, showing that it outperforms strong baselines.

1 Introduction

Multi-document summarization (MDS) aims to generate a summary with the salient information from multiple documents on a certain topic, such as multiple news articles about an event (Fabbri et al., 2019) or reviews of a product (Bražinskas et al., 2020). However, different users often have different or even conflicting *preferences* of *writing styles* or *content focuses* for summaries (Jang et al., 2023). While writing style refers to the manner or tone in which the summaries are written, content focus refers to which aspects are emphasized when presenting a certain topic. Users can have different preferences for writing style. For example, for product reviews, some users may prefer a formal tone, while others may prefer a conversational tone. User preferences can also differ on *content focus*. Some users may focus on price and utility of the product while others might focus on quality and durability. Therefore, to meet these individual user preferences, personalized MDS is essential.

Personalized MDS is related to personalized text generation. Recent works on personalized text generation use Large Language models (LLMs) and assume access to the *profile* of individual users–set of documents previously authored by the user. They then either retrieve related documents from a user's profile (Salemi et al., 2024; Li et al., 2023a), include a summary of the user's profile (Richardson et al., 2023), or tune different models for different users based on their profiles (Tan et al., 2024). However, most of these works only include general features of the user and ignore finer differences between users. To identify the finer differences, it is important to compare a user's profile documents with comparable profile documents written by other users. Ideally, the profile documents of two users can be comparable if they are on the same topic but differ on personal preferences. In general personalized text generation, identifying such comparable profile documents of different users can be difficult since the differences between profile documents of different users can stem from either personal preferences or topic differences. Contrarily, for MDS, since all input documents are about the same topic (e.g. reviews about the same product), their differences are more likely to stem from differences in personal preferences of their authors (users).

Motivated by this, we propose `ComPSum`(**Com**parative **P**ersonalization for Multi-Document **Sum**marization), a personalized MDS framework. Specifically, `ComPSum` considers two key preference dimensions: writing style and content focus (Zhang et al., 2024). `ComPSum` first generates a structured analysis of these two dimensions for a user by comparing profile

Style analysis: User X's writing style is characterized by its concise and straightforward narrative. Unlike other users who favor long and complex analysis, user X uses simple and straightforward language...
Content analysis: User X's profile texts tend to focus on personal reflections and positive developments. Unlike other users, User X avoids sensationalism but focus more on practical ...

Table 1: Structured analysis of a user on dimensions of writing style and content focus generated by ComPSum.

documents with other documents authored by different users on the same topic as shown in Tab. 1. The generated structured analysis is then used to guide the generation of personalized summaries that capture the user's individual preference.

Apart from generating personalized summaries, their evaluation is also a major challenge since there is no reference summary and conventional metrics like ROUGE (Lin, 2004) cannot be applied. To address this issue, we propose AuthorMap, a fine-grained reference-free evaluation framework for personalized MDS that independently assesses writing style and content focus. To control the inherent differences between different systems, AuthorMap evaluates the personalization of a system based on the authorship attribution between two personalized summaries generated for different users by the same system. We perform automatic and human evaluation of AuthorMap and show that it achieves reasonable accuracy.

For robust evaluation of personalized MDS, we construct PerMSum, an MDS dataset spanning reviews and news domains. For evaluation, we need documents labeled with their authors (users). While there are MDS datasets with user labels in the review domain (Ni et al., 2019), the news domain still lacks such datasets. To construct such a dataset, we use news articles from the All The News dataset [1]. We collect and process 14K document sets and 1.4K users each of whom authored at least 10 documents. These document sets and corresponding author information are then combined with the document sets sampled from the Amazon dataset (Ni et al., 2019) to form PerMSum, which results in 45K document sets and 5.3K users.

Using AuthorMap, we find that ComPSum achieves consistent improvement on PerMSum with different LLMs while maintaining other critical qualities of summaries, such as relevance and factuality.

Our contributions are four-fold:

[1]https://components.one/datasets/all-the-news-2-news-articles-dataset/

- We propose ComPSum, a personalized MDS framework based on comparative personalization;
- We propose AuthorMap, a fine-grained reference-free evaluation framework for personalized MDS;
- We propose PerMSum, a personalized MDS dataset in the news and review domain;
- Evaluation of ComPSum on PerMSum using AuthorMap shows that it outperforms strong baselines.

2 Related Work

Personalized text generation aims to generate a personalized text for a user using his profile, making it the task most closely related to personalized MDS. Specifically, Salemi et al. (2024) retrieve related documents from a user's profile and Li et al. (2023a) train models to summarize and synthesize the retrieved documents. To mitigate the information loss in retrieval, Richardson et al. (2023) additionally include a summary of user profile. However, these works model user individually. Recently, Sun et al. (2025) use other similar users' profile to infer a user's profile when existing data about the user is sparse. Qiu et al. (2025) improve personalized review generation by comparing a user's review with other user's review. However, their design is specific to the review domain and may not generalize well to other domains. We perform experiments to show that ComPSum outperforms their method in Sec. 7.4. Another line of works (Daumé III, 2009; Padmakumar et al., 2025) generates personalized summaries using user-provided queries, which differs from personalized MDS in this work that uses user profiles.

Most previous works on personalized text generation (Salemi et al., 2024; Ao et al., 2021) use reference-based metrics that evaluate the similarity between generated texts and human-written references, such as ROUGE (Lin, 2004) and BERTScore (Zhang et al., 2019). Recently, Salemi et al. (2025) propose to evaluate personalized text generation from writing style and content, which are similar to the dimensions used by AuthorMap, but it assesses how well the generated text aligns with the reference text, which differs from the reference-free AuthorMap. For reference-free evaluation, Wang et al. (2023) propose AuPEL, which evaluates personalization based on pairwise authorship attribution (Bozkurt et al., 2007) between two personalized texts from different systems for the same user. However, AuPEL overlooks various dimensions of personalization and does not control the inherent differences in writing styles or content focuses between different systems. Zhang et al. (2025) uses aspect and sentiment similarity between personalized summaries and user profiles to evaluate personalization. However, their design is specific to the review domain and may not generalize well to other domains.

3 Problem Statement

The input of personalized MDS is a document set D containing multiple documents on the same topic to be summarized. For personalization for user u, the input also contains a profile P_u containing multiple profile documents $p_u^i \in P_u$ authored by the user u. Given these inputs, the output of personalized MDS is a personalized summary s_u of the document set D that capture the individual preference of user u as expressed in their profile P_u.

4 `ComPSum`

In this section, we describe our proposed framework for personalized MDS, `ComPSum`. We first describe how `ComPSum` generates a structured analysis a_u of a user u that captures their distinctive features of writing styles and contents focuses by comparing with documents written by other users. We then describe how it uses the structured analysis a_u to generate a personalized summary s_u.

Generating structured analysis: `ComPSum` uses an LLM to generate structured analysis, a_u, of a user u from two dimensions: writing style and content focus. `ComPSum` explicitly focuses on these two dimensions so that it only captures preferences but not unrelated information, like a general summary of each profile document. To generate the structured analysis, following Salemi et al. (2024), `ComPSum` first retrieves the top k documents from user u's profiles P_u using a retrieval model $\mathcal{R}$. For the retrieval query, `ComPSum` uses the concatenation of all documents belonging to document set D and retrieves k profile documents most similar to the query: $\mathcal{R}(D, P_u, k)$.

However, generating the structured analysis a_u only based on the retrieved profile documents can make the analysis focus on general features of the user u but ignore finer differences compared to other users. To address this issue, comparing documents on the same topic written by other users can be useful. Therefore, for each retrieved profile document $p_u^i \in \mathcal{R}(D, P_u, k)$, `ComPSum` identifies a set of documents, $C_{\neg u}^{p_i}$, that belong to the same document set as p_u^i (and hence are on the same topic) but are written by different users. `ComPSum` then retrieves one comparative document $p_{\neg u}^i \in C_{\neg u}^{p_i}$ that is most dissimilar to p_u^i using the retrieval model $\mathcal{R}$. By comparing every pair of profile document p_i and its comparative document $p_{\neg u}^i$, `ComPSum` then instructs an LLM to generate the structured analysis a_u consisting of two distinct components, each highlighting the differences in writing style and content focus that distinguish user u from others:

$$a_u = LLM(p_u^1, p_{\neg u}^1, ..., p_u^k, p_{\neg u}^k) \quad (1)$$

An example structured analysis of `ComPSum` is shown in Tab. 1. As illustrated, the structured analysis captures finer differences between the users.

Generating personalized summary: Using the structured analysis a_u and the retrieved profile documents $p_u^1, ...p_u^k$, `ComPSum` generates a personalized summary s_u for the document set D:

$$s_u = LLM(p_u^1, ..., p_u^k, a_u, D) \quad (2)$$

Specifically, `ComPSum` instructs the LLM to generate a summary s_u that cover the main information from the document set D while mimics the writing style and content focus based on retrieved profile documents $p_u^1, ..., p_u^k$ and structured analysis a_u. To prevent information leakage, `ComPSum` explicitly instructs the LLM to only include contents from the document set D but not from the profile texts. The prompts for `ComPSum` are in App. A.1.

5 `AuthorMap`

In this section, we describe our proposed fine-grained reference-free evaluation framework, `AuthorMap`. `AuthorMap` evaluates personalization along two key dimensions: writing style and content focus. The underlying idea behind `AuthorMap` is that if the generated summary is well-personalized, it will be possible to infer the user's preferences from the summary and use them for the task of authorship attribution. `AuthorMap` considers two profiles, P_{u_1} and P_{u_2}, and two personalized summaries, s_{u_1} and s_{u_2}, of the same input document set, D, generated for user u_1 and u_2 respectively. `AuthorMap` then evaluates whether each profile P_{u_*} can be correctly attributed to its user based on personalized summaries.

`AuthorMap` performs such evaluation separately for writing style and content focus as they are independent dimensions (Jafaritazehjani et al., 2020).

Algorithm 1 Pseudo Code for AuthorMap

Input: personalized summaries s_{u_1}, s_{u_2}, user profile P_{u_1}, P_{u_2}, number of retrieved profiles n
$P_{u_1} = \mathcal{R}(s_{u_1} \circ s_{u_2}, P_{u_1}, n)$
$P_{u_2} = \mathcal{R}(s_{u_1} \circ s_{u_2}, P_{u_2}, n)$
count = 0 ▷ count number of correct judgements
for $(s_a, s_b) \in \{(s_{u_1}, s_{u_2}),(s_{u_2}, s_{u_1})\}$ **do**
    count += $(u_1 == LLM_{judge}(P_{u_1}, s_a, s_b))$
    count += $(u_2 == LLM_{judge}(P_{u_2}, s_a, s_b))$
end for ▷ Mitigate position bias of LLM judge
Return (count>=3) ▷ Aggregate judgments

However, user preferences for these dimensions may vary across topics. When evaluating for a certain dimension, AuthorMap first retrieves the top n profile documents from each user's profile P_{u_*} most similar to the concatenation s_{u_1} and s_{u_2} using the retrieval model $\mathcal{R}$: $\mathcal{R}(s_{u_1} \circ s_{u_2}, P_{u_*}, n)$, where $\circ$ denotes the concatenation. AuthorMap uses concatenated summaries instead of just one summary to ensure that no summary has inherent advantages. For simplicity, we use P_{u_1} and P_{u_2} to denote the retrieved profiles. For each retrieved profile P_{u_*}, AuthorMap instructs a judge LLM to predict which user, u_1 or u_2, is more likely to be the author of the retrieved profile P_{u_*} given their personalized summaries, s_{u_1} and s_{u_2} for the given dimension:

$$\hat{u}_1 = LLM_{judge}(P_{u_1}, s_{u_1}, s_{u_2}) \tag{3}$$

$$\hat{u}_2 = LLM_{judge}(P_{u_2}, s_{u_1}, s_{u_2}) \tag{4}$$

where $\hat{u}_*$, the predicted author of the profile P_{u_*}, can be u_1, u_2 or tie. If the summary s_{u_1} is well personalized for user u_1, the LLM judge will be able to attribute the profile P_{u_1} to user u_1. The same should also apply to user u_2.

To mitigate positional bias (Huang et al., 2023), AuthorMap performs such prediction twice with different orders of s_{u_1} and s_{u_2}, which results in four predictions in total. AuthorMap evaluates the personalization capability of a personalized MDS system as the percentage of samples where the judge LLM correctly predicts the author of the retrieved profile in the majority of four predictions. A larger percentage value indicates better personalization capability on the corresponding dimension of the personalized MDS system. The pseudo code of AuthorMap is shown in Alg. 1. The prompts for AuthorMap are shown in App. A.1.

6 PerMSum

In this section, we describe the construction of the PerMSum dataset. We first describe how PerMSum obtains document sets labeled with a user (its author). We then describe how to select samples from PerMSum for the evaluation using AuthorMap.

Obtaining Document Set with User Label: For the news domain, PerMSum uses the All the News dataset[2], which includes metadata of author names, publishing media, and publishing dates. However, the news articles from it have two issues for direct application in personalized MDS. First, some news articles contain explicit mentions of the author or media outlet (e.g., "XXX reports in New York"), which are undesirable shortcuts for personalized summarization. To address this, PerMSum removes all sentences containing author names or publishing media. Second, some news articles have more than three authors or media organizations as authors. Since these news articles might not truly reflect the preference of their individual authors, PerMSum excludes them from being using as profile documents. After labeling documents with users, PerMSum clusters documents into document sets based on token overlap, named entities, and publishing dates as Liu et al. (2022) so that documents in each document set are about the same event.

For the review domain, PerMSum uses book reviews of the Amazon dataset (Ni et al., 2019) following Wang et al. (2023). PerMSum preprocesses the reviews and obtains document sets following Bražinskas et al. (2019) by only keeping reviews that are between 50 to 150 words and are written in English. PerMSum also filters out reviews written by users who write more than 200 reviews.

For both domains, PerMSum only considers users that write at least 10 documents and splits the users into training, validation, and test sets using the user split motivated by Salemi et al. (2024). Document sets are also split to prevent information leakage, ensuring no overlap across splits. More details of data curation process are in App. A.3.

Sample selection for AuthorMap: Each sample for AuthorMap consists of a document set D and two personalized summaries for users u_1 and u_2. However, randomly sampling two users for evaluation can face sparsity in personalization (Dong et al., 2024). For example, for a user whose profile documents are all entertainment news, it can be difficult to get enough information from his profile for generating a personalized summary of international news. To address this issue, for each document set D, PerMSum only selects pairs of users

[2]https://components.one/datasets/all-the-news-2-news-articles-dataset/

u_1 and u_2 who write documents belonging to the document set D. To prevent the system from copying user-authored contents, we remove all documents written by the users u_1 or u_2 from the input document set. The generated pairs of personalized summaries are then used for evaluation using AuthorMap. To prevent users from dominating the evaluation, PerMSum limits each user to appear in at most 100 samples. Statistics of PerMSum are reported in Tab. 2.

7 Experiments

In this section, we describe experiments on AuthorMap and ComPSum.

7.1 Implementation Details

For AuthorMap, we use Llama3.3-70b-Instruct (AI@Meta, 2024) as the LLM judge. AuthorMap can also use other LLMs as the LLM judge. We perform evaluation using Gemma-3-27b-it (Team, 2025) in App. A.6 and find that AuthorMap is independent of the choice of the LLM judge. For evaluation, AuthorMap retrieves $n = 5$ profile documents using BM25 (Robertson et al., 1995). Motivated by Huang et al. (2024), the LLM is instructed to additionally focus on linguistic features when evaluating writing style.

For ComPSum, we experiment with Llama3.1-8b-Instruct (AI@Meta, 2024), Qwen2.5-14B-Instruct (Yang et al., 2024), and Llama3.3-70b-Instruct. For personalization, ComPSum retrieves $m = 5$ profile documents and comparative documents using BM25. The token limit for personalized summaries is 100 words. All LLMs used in experiments use the default sampling parameter. Hyperparameters and prompts are tuned on the validation set. More implementation details are in App. A.1.

7.2 Automatic Evaluation of AuthorMap

In this section, we perform large-scale automatic evaluation to evaluate the accuracy of AuthorMap. Since there is no existing reference personalized summaries, we use documents from the same document set but written by two users of interest for this evaluation. To mimic the setting of AuthorMap, the documents are truncated to 100 words, matching the length of a typical generated personalized summary. We report the accuracy of AuthorMap on human-written documents in the test sets of PerMSum. For comparison, we also report the accuracy of AuthorMap when profile documents P_{u_*} are not retrieved but randomly sampled from user profiles. This setup resembles the setting of Wang et al. (2023). The results are in Tab. 3.

From the table, we observe that AuthorMap shows reasonable accuracy on the documents, suggesting that it can reliably evaluate personalization in different dimensions. Besides, AuthorMap outperforms its variant without retrieval, which is used by Wang et al. (2023), showing that retrieval is useful to capture the varying preferences of users on different topics.

Writing style and content focus are independent dimensions (Jafaritazehjani et al., 2020) and AuthorMap should be able to evaluate them independently. To enable controlled evaluation, we test the accuracy of AuthorMap on paraphrased documents with altered writing styles. Specifically, we evaluate two types of document pairs : d_{u_1} vs $para(d_{u_2})$, and d_{u_1} vs $para(d_{u_1})$, where d_{u_*} denotes the original document written by user u_*, $para(d_{u_*})$ denotes a paraphrased document following the writing style of the other user. The first type of document pair evaluates from the perspective of the content focus with similar writing styles, while the second evaluate from the perspective of the writing style with similar content focuses. We use Llama3.3-70b-Instruct for paraphrasing (prompt in App. A.4). Please note that the paraphrasing may not be perfect as it may not completely mimic the writing style of the given user. However, it is sufficient for us to test the independence based on changes in AuthorMap's accuracies. If AuthorMap evaluates writing style and content focus independently, AuthorMap should show higher accuracy on d_{u_1} vs $para(d_{u_2})$ when evaluating content focus since the document pair has differ in content focus but not much in style. Conversely, when evaluating writing style, AuthorMap should show higher accuracy on d_{u_1} vs $para(d_{u_1})$ since the document pairs differ in style but not much in content focus. The results are shown in Tab. 4.

From the table, we observe the desirable pattern: AuthorMap shows higher accuracy on d_{u_1} vs $para(d_{u_2})$ when evaluating content focus and higher accuracy on d_{u_1} vs $para(d_{u_1})$ when evaluating writing style. The result shows that AuthorMap can evaluate personalization based on either writing style or content focus independently.

7.3 Human Evaluation of AuthorMap

We perform a human evaluation to further evaluate the correlation between AuthorMap and human judgment. Following Wang et al. (2023), each hu-

	#User	#Doc. Set	#Sample	Prof. Size	Doc. Set. Size	Doc. Len.
News	828/293/296	10730/1393/1463	-/2085/2360	39.72	3-10	216.72
Review	2400/763/766	27725/1878/1795	-/2774/2757	19.77	8	86.40

Table 2: Statistics of `PerMSum`. We report numbers of users (#User), document sets (#Doc. Set), and evaluation samples (#Sample) for the training, validation, and test sets, respectively, separated by '/'. We also report average profile size per user, size of input document sets, and length of documents.

	News		Review	
	style	content	style	content
`AuthorMap`	76.65	71.64	89.00	82.69
w/o retrieval	75.10	68.71	87.88	81.27

Table 3: Automatic Evaluation of `AuthorMap`. `AuthorMap` shows high accuracy on human-written documents and outperforms its variant without retrieval.

	News		Review	
	style	content	style	content
d_{u_1} vs $para(d_{u_2})$	61.34	58.76	70.34	70.58
d_{u_1} vs $para(d_{u_1})$	68.77	31.68	77.93	54.10

Table 4: Accuracy of `AuthorMap` on paraphrased human written documents. It shows that `AuthorMap` can independently evaluate writing style and content focus.

man evaluation sample contain two personalized summaries generated for different users and a set of retrieved profile documents written by one of the user. For each sample, human annotators are instructed to select which summary better aligns with the writing style or content focus expressed by the profile documents. We then calculate the accuracy of `AuthorMap` by measuring proportion of samples where `AuthorMap` and human annotation based on majority select the same summary.

We randomly sample 25 samples from each domain, news and reviews, yielding 50 samples in total. Personalized summaries of each sample are generated by `ComPSum` with Llama3.1-8b, which shows the medium-level performance in Tab. 5. Each sample is annotated by three annotators recruited from Amazon Mechanical Turk. More details of human evaluation are in App. A.2.

Among 50 samples, the Randolph's Kappa (Randolph, 2005) between three annotators is 0.40, showing a moderate correlation. On the news domain, `AuthorMap` achieves 80 percent accuracy for writing styles and 72 percent for content focus. On the review domain, `AuthorMap` achieves 72 percent accuracy for writing style and 88 percent for content focus. The results show that `AuthorMap` achieves high accuracy when compared to human annotated labels.

7.4 Evaluation of `ComPSum`

In this section, we evaluate the qualities of personalized summaries generated by `ComPSum`. Specifically, we evaluate both personalization and general qualities of summaries. For personalization, we consider two dimensions: writing style and content focus using `AuthorMap`. For general qualities, we consider factuality, which measures whether summaries only contain information supported by the input, and relevance, which measures whether summaries only include important information from document sets (Fabbri et al., 2021). We use FactScore (Min et al., 2023) for factuality and G-Eval (Liu et al., 2023) for relevance. To match the scales of other measures, we map G-Eval's score to 1-100. We also report an overall score, which is the arithmetic average of the four measures. A higher value indicates better overall quality.

We compare `ComPSum` with the following baselines: (i) General, which generates general summaries without using user profiles; (ii) RAG (Salemi et al., 2024) that generates a personalized summary based on retrieved profile documents; (iii) CICL (Gao and Das, 2024), which extends RAG by additionally retrieving comparative documents authored by other users; (iv) RAG+Summ (Li et al., 2023a), which generates summaries of the retrieved profile documents and uses them to guide generation of personalized summaries; (v) DPL (Qiu et al., 2025), which first generates an analysis for each retrieved profile document by comparing its comparative documents and then aggregate these analyses into a profile summary to generate the personalized summary; (vi) Rehearsal, which (Zhang et al., 2025) iteratively refines a general summary using a user agent and a supervisor agent. More details for implementation of these baselines are shown in App. A.5. The results of `ComPSum` and baselines on the test set of `PerMSum` are in Tab. 5.

From the table, we observe that `ComPSum` generally outperforms all baselines on personalization and general summary qualities and also achieves the best overall scores. All differences between `ComPSum` and the second-best performing methods

	News					Review				
	style	content	fact.	rele.	overall	style	content	fact.	rele.	overall
Llama3.1-8b-Instruct										
General	43.60	36.48	97.86	98.51	62.58	42.79	38.69	98.58	93.94	62.58
RAG	53.35	49.61	98.00	96.05	70.65	52.52	54.12	98.16	92.71	71.32
CICL	53.22	48.89	97.34	95.47	70.12	54.56	55.83	96.87	88.91	71.57
RAG+Summary	54.58	50.89	98.03	97.17	71.72	54.56	57.10	97.67	92.02	72.74
DPL	53.94	47.90	97.91	96.30	70.26	54.70	59.13	97.05	87.68	72.43
Rehersal	99.36	99.49	23.16	28.75	50.65	97.97	98.40	57.28	37.62	67.51
ComPSum	59.75	53.94	98.01	95.32	**74.07**	59.13	57.89	98.03	91.99	**74.54**
Qwen2.5-14B-Instruct										
General	44.92	40.42	98.21	98.71	64.77	42.03	42.61	97.89	93.55	63.64
RAG	52.88	48.26	98.07	96.77	70.15	48.78	51.18	97.76	91.76	68.79
CICL	51.69	48.64	97.11	96.68	69.71	54.66	55.90	96.95	88.91	71.64
RAG+Summary	52.75	49.32	98.17	97.93	70.72	54.88	59.75	97.40	90.97	73.42
DPL	50.97	46.31	98.04	97.37	68.90	50.71	57.53	96.83	89.42	70.89
Rehersal	98.69	99.41	22.86	26.94	49.58	95.79	98.69	53.20	33.63	64.13
ComPSum	57.92	57.08	97.96	96.58	**74.78**	60.36	63.92	96.51	89.98	**76.08**
Llama3.3-70b-Instruct										
General	41.10	32.03	98.75	99.03	59.90	45.15	38.73	98.36	94.17	63.44
RAG	48.64	39.96	98.59	97.98	65.83	50.82	46.13	98.76	93.16	68.15
CICL	50.04	44.41	98.55	98.03	68.07	49.36	52.09	98.70	92.79	69.66
RAG+Summary	49.96	41.57	98.70	98.08	66.96	52.99	47.80	98.68	93.43	69.52
DPL	49.32	41.40	98.81	98.17	66.71	50.96	46.21	98.50	93.10	68.17
Rehersal	99.49	100.00	21.74	24.90	48.17	97.53	98.55	62.03	33.81	67.00
ComPSum	53.90	45.38	98.64	98.03	**69.74**	59.27	51.87	98.70	93.35	**72.95**

Table 5: Evaluation of ComPSum. A higher value indicates better performance. The best-performing method based on overall score is **bolded**. ComPSum shows the best overall performance.

are statistically significant using paired bootstrap resampling ($p < 0.05$) (Koehn, 2004).

Interestingly, Llama3.1-8B-Instruct generally outperforms Llama3.3-70B-Instruct, despite having fewer parameters. One possible explanation is that Llama3.3-70B-Instruct tends to produce more concise summaries, even though both LLMs share the same token limit in their instructions. Since shorter summaries generally contain fewer cues about personal preferences, they can be difficult to be classified accurately.

Although Rehearsal performs well on personalization, its performance on general qualities is pretty low. We find that many summaries generated by Rehearsal resemble user profiles rather than faithful summaries of input document sets. Further analysis of the 'modification suggestions' generated by Rehearsal shows that they often suggest adding information that only exists in the user profile but not input document sets. This can be caused by the fact that neither the user agent nor the supervisor agent has access to input document sets when generating the suggestions. The finding also shows the importance of considering both personalization and general qualities during evaluation.

Furthermore, ComPSum outperforms DPL especially in the news domain while both methods use comparative documents when generating analysis of users. This can be caused by two reasons. First, DPL is designed specifically for reviews and instructs the LLM to focus on aspects like emotional style that do not generalize to domains beyond reviews (and in this sense this comparison is not fair to DPL). Second, DPL's analysis is initially generated from a single profile document, which is generally not enough to infer the preference of a user. Even though DPL later summarizes multiple analyses, information loss is still inevitable. Contrarily, ComPSum generates the structured analysis of a user directly conditioned on multiple profile documents. A fairer comparison with DPL using the same aspect as ComPSum is provided in Sec. 7.5.

Finally, all other methods outperform General, which does not use user profiles when generating summaries. The result shows that user profiles in PerMSum capture individual user preferences useful for personalized summarization, even though the profile texts (news articles or reviews) differ slightly in genre from summaries. It can be because authors tend to retain consistent stylistic choices or content focus across different document types as shown in previous works on cross-genre authorship attribution (Ma et al., 2025).

7.5 Ablation Study of ComPSum

In this section, we validate the design of ComPSum by comparing it with the following ablated variants: (i) **w/o comp. doc.**, which generates structured analysis of a user without comparative documents; (ii) **w/o structure**, which instructs LLM to generate a profile summary of a user instead of separate analysis of writing style and content focus; (iii) **w/ sim. comp.**, which generates the structured analysis based on comparative

	Llama 3.1-8b News	Llama 3.1-8b Review	Qwen2.5-14b News	Qwen2.5-14b Review	Llama 3.3-70b News	Llama 3.3-70b Review	Avg.
ComPSum	**74.08**	74.54	**74.78**	76.08	**69.74**	72.95	**73.70**
w/o comp. doc.	72.34	73.24	72.34	76.10	67.40	72.05	72.24
w/o structure	71.43	**78.43**	69.34	75.88	69.69	**75.69**	73.41
w/ sim. comp.	72.07	74.00	71.84	**76.85**	69.22	72.86	72.81
w/ multi. stage	69.87	75.15	68.62	74.43	67.14	72.63	71.31

Table 6: Ablation study of ComPSum. The best-performing method is **bolded**. All differences between the best and the second-best performing methods are statistically significant using paired bootstrap resampling ($p < 0.05$). ComPSum outperforms its ablated version, showing the effectiveness of ComPSum design.

	Llama3.1-8b News	Llama3.1-8b Review	Qwen2.5-14b News	Qwen2.5-14b Review	Avg.
ComPSum	80.93	82.63	78.81	81.18	80.89
w/o comp. doc.	83.18	82.74	81.73	82.42	82.52

Table 7: Average similarity between structured analysis of different users. Comparative documents can make structured analysis more diverse for different users.

documents that are most similar to profile documents; (iv) **w/ multi. stage**, which generates the structured analysis in multiple stages, similar to DPL, but focusing on writing style and content focus instead of dimensions used by DPL. For these ablated variants, we report their overall scores in Tab. 6 which evaluates personalization and general qualities as described in Sec. 7.4 on the test set of PerMSum. The detailed scores for each dimension and implementation details are shown in App. A.7.

From the table, we observe that ComPSum outperforms w/o comp. doc. and w/o structure, which shows the effectiveness of comparative documents and structure constraints. Besides, ComPSum also outperforms w/ multi. stage, which shows that directly generating analysis based on multiple profile documents is more effective than generating multiple analyses based on one profile document and summarizing them afterward. Overall, ComPSum outperforms all of its ablated version based on average performance across LLMs and datasets. We show examples of summaries generated by ComPSum in App. A.9. We also perform ablation study on number of retrieved documents for ComPSum in App. A.10 and find that retrieve 5 profile documents performs the best.

7.6 Analysis Generated by ComPSum

In this section, we examine whether using comparative documents leads to more diverse structured analysis for different users. To evaluate this, for each sample in the test set, we measure the average similarity between the structured analysis generated for two users for the same input document set. We then compare the similarity scores produced by ComPSum with those from its ablated variant, **w/o comp. doc.**, which generates structured analysis without using comparative documents. To measure the similarity, we report cosine similarity of structured analysis's embedding generated by gte-Qwen2-1.5B-instruct (Li et al., 2023b) in Tab. 7.

ComPSum: User X's writing style is characterized by a clear and concise narrative voice, often incorporating direct quotes and specific details... **Unlike other users, who may rely on sensational language or emotional appeals**, User X's tone is measured and informative, making their content feel more authoritative and trustworthy.
w/o comp. doc.: User X's writing style is characterized by a conversational tone and a focus on storytelling. They often use anecdotes and quotes from celebrities to illustrate their points, making their content feel more relatable and engaging. The text is also well-structured and easy to follow, with a clear and concise writing style...

Table 8: Structured analysis for writing style generated by ComPSum and w/o comp. doc. The structured analysis generated by ComPSum includes comparison with other users (in **bold**), which helps in better personalization.

From the table, we observe that ComPSum has lower similarity score than w/o comp. doc., suggesting that using comparative documents leads to more diverse structured analysis for different users. We also show examples of structured analysis for writing style generated by ComPSum and w/o comp. doc. in Tab. 8. From the examples, we can observe that the structured analysis generated by ComPSum additionally includes comparison with other users, which helps the MDS system to better differentiate different users. We show additional examples of structured analysis in App. A.8.

8 Conclusion

We propose ComPSum, a personalized MDS framework. It captures the finer differences between users by comparing profile documents with other documents authored by different users. We also propose AuthorMap, a reference-free fine-grained evaluation framework. We perform automatic and human evaluation to evaluate AuthorMap and show that it achieves reasonable accuracy. For robust evaluations of ComPSum, we construct PerMSum, a personalized MDS dataset in the news and review domain. We evaluate the performance of ComPSum on PerMSum using AuthorMap, showing that it outperforms strong baselines.

9 Limitation

One limitation of `CompSum` is its reliance on comparable documents that share the same topic as the profile documents. Otherwise, the differences between profile documents and comparable can stem from topic difference but not individual preference differences. Therefore, `CompSum` cannot be directly applied to general personalized text generation as identifying such comparable documents can be difficult for other tasks. Future work could explore methods for automatically identifying or generating comparable documents to broaden the applicability of `CompSum` to more general personalized text generation

When constructing the `PerMSum` dataset, we define the profile documents for a user as documents written by the user. However, for the news domain, defining the profile documents as documents clicked or liked by the user seems to be more similar to the real-world application. Unfortunately, existing publicly available news datasets with user interaction data are not suitable for personalized MDS. For example, the PENS dataset (Ao et al., 2021) lacks publishing dates of news articles, which makes it difficult to efficiently create input document sets where all documents are about the same event. MIND dataset (Wu et al., 2020) lacks both publishing dates and full texts of news articles. Therefore, we construct `PerMSum` using articles from All the News dataset, which is the only large-scale dataset that includes publishing dates, author information, and full article texts.

10 Ethical Consideration

The datasets we use are all publicly available. All the models used in this paper are publicly accessible. The inference and finetuning of models are performed on four Nvidia A6000 or Nvidia L40 GPUs. We do not annotate any data on our own.

We perform human evaluation experiments on Amazon Mechanical Turk. The annotators were compensated at a rate of $20 per hour. During the evaluation, human annotators were not exposed to any sensitive or explicit content.

We use LLMs to polish the writing of the paper.

References

AI@Meta. 2024. Llama 3 model card.

Xiang Ao, Xiting Wang, Ling Luo, Ying Qiao, Qing He, and Xing Xie. 2021. Pens: A dataset and generic framework for personalized news headline generation. 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 82–92.

Ilker Nadi Bozkurt, Ozgur Baghoglu, and Erkan Uyar. 2007. Authorship attribution. In *2007 22nd international symposium on computer and information sciences*, pages 1–5. IEEE.

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2019. Unsupervised opinion summarization as copycat-review generation. *arXiv preprint arXiv:1911.02247*.

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2020. Unsupervised opinion summarization as copycat-review generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5151–5169.

Hal Daumé III. 2009. Bayesian query-focused summarization. *arXiv preprint arXiv:0907.1814*.

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.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Alexander Richard Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084.

Xiang Gao and Kamalika Das. 2024. Customizing language model responses with contrastive in-context learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18039–18046.

Baixiang Huang, Canyu Chen, and Kai Shu. 2024. Can large language models identify authorship? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 445–460.

Kung-Hsiang Huang, Philippe Laban, Alexander R Fabbri, Prafulla Kumar Choubey, Shafiq Joty, Caiming Xiong, and Chien-Sheng Wu. 2023. Embrace divergence for richer insights: A multi-document summarization benchmark and a case study on summarizing diverse information from news articles. *arXiv preprint arXiv:2309.09369*.

Somayeh Jafaritazehjani, Gwénolé Lecorvé, Damien Lolive, and John Kelleher. 2020. Style versus content: A distinction without a (learnable) difference?

In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2169–2180, Barcelona, Spain (Online). International Committee on 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. *arXiv preprint arXiv:2310.11564*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

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. *arXiv preprint arXiv:2308.07968*.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

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.

Y Liu, X Zhang, D Wegsman, N Beauchamp, and L Wang. 2022. Politics: Pretraining with same-story article comparison for ideology prediction and stance detection. *Findings of the Association for Computational Linguistics: NAACL 2022*.

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.

Marcus Ma, Duong Minh Le, Junmo Kang, Yao Dou, John Cadigan, Dayne Freitag, Alan Ritter, and Wei Xu. 2025. Crossnews: A cross-genre authorship verification and attribution benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 24777–24785.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *EMNLP*.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. 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 188–197, Hong Kong, China. Association for Computational Linguistics.

Vishakh Padmakumar, Zichao Wang, David Arbour, and Jennifer Healey. 2025. Principled content selection to generate diverse and personalized multi-document summaries. *arXiv preprint arXiv:2505.21859*.

Yilun Qiu, Xiaoyan Zhao, Yang Zhang, Yimeng Bai, Wenjie Wang, Hong Cheng, Fuli Feng, and Tat-Seng Chua. 2025. Measuring what makes you unique: Difference-aware user modeling for enhancing llm personalization. *arXiv preprint arXiv:2503.02450*.

Justus J Randolph. 2005. Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa. *Online submission*.

Chris Richardson, Yao Zhang, Kellen Gillespie, Sudipta Kar, Arshdeep Singh, Zeynab Raeesy, Omar Zia Khan, and Abhinav Sethy. 2023. Integrating summarization and retrieval for enhanced personalization via large language models. *arXiv preprint arXiv:2310.20081*.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Alireza Salemi, Julian Killingback, and Hamed Zamani. 2025. Expert: Effective and explainable evaluation of personalized long-form text generation. *Preprint*, arXiv:2501.14956.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. 2024. 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.

Chenkai Sun, Ke Yang, Revanth Gangi Reddy, Yi Fung, Hou Pong Chan, Kevin Small, ChengXiang Zhai, and Heng Ji. 2025. Persona-db: Efficient large language model personalization for response prediction with collaborative data refinement. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 281–296.

Zhaoxuan Tan, Qingkai Zeng, Yijun Tian, Zheyuan Liu, Bing Yin, and Meng Jiang. 2024. Democratizing large language models via personalized parameter-efficient fine-tuning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6476–6491, Miami, Florida, USA. Association for Computational Linguistics.

Gemma Team. 2025. Gemma 3.

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. *arXiv preprint arXiv:2310.11593*.

Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. 2020. Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 3597–3606.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yanyue Zhang, Yulan He, and Deyu Zhou. 2025. Rehearse with user: Personalized opinion summarization via role-playing based on large language models. *arXiv preprint arXiv:2503.00449*.

Zhehao Zhang, Ryan A Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, et al. 2024. Personalization of large language models: A survey. *arXiv preprint arXiv:2411.00027*.

A Appendix

A.1 Implementation Details

`ComPSum` first generates a structured analysis for a user by comparing profile documents with other documents authored by different users on the same topic. `ComPSum` then uses the generated structured analysis to guide the generation of personalized summaries that capture the user's writing style and content focus. The prompt for the generation of structured analysis is shown in Fig. 1. The prompt for the generation of personalized summaries is shown in Fig. 2.

`AuthorMap` separately evaluates writing style and content focus. The design of the prompt for evaluating writing style is motivated by Huang et al. (2024). The prompt for evaluation of writing style is shown in Fig. 3. The prompt for evaluation of content focus is shown in Fig. 4.

We obtain the above prompts by first tuning the prompt and number of retrieved documents of `AuthorMap` to optimize its accuracy in automatic evaluation on the validation set as described in Sec. 7.2. We also adjust the prompts to ensure the generated judgments follow the given output format. We then keep the prompt for `AuthorMap` fixed and tune the prompt and number of retrieved documents of `ComPSum` to optimize its performance on the validation set measured by `AuthorMap`. We also adjust the prompts to ensure the generated structured analysis follows the given output format and the final generated summary does not have direct comparison with other users.

To match the token limit of generated summaries, all retrieved profile documents of `AuthorMap` and `ComPSum` are truncated to 100 words. This truncation not only matches the token limit of generated summaries but also prevents models from exploiting document length as a shortcut for personalization, as profile documents written by different users can vary considerably in length. For instance, in the news domain, the standard deviation of average profile lengths across users is 60 words without truncation, a large variation relative to the average length of 216 words (Table 2).

During inference, we set the temperature as 0.6 and top-p as 0.9 for Llama3.1-8b-Instruct and Llama3.3-70b-Instruct. For Qwen2.5-14B-Insturct, we set the temperature as 0.7, top-p as 0.8, top-k as 20, and repetition penalty as 1.05. Additionally, for Llama3.3-70B-Instruct, we apply 8-bit quantization to accelerate inference.

A.2 Human Evaluation of `AuthorMap`

Each human evaluation sample contains two personalized summaries generated for different users and a set of profile documents written by one of the users. We retrieve 5 profile documents using the concatenation of two personalized summaries as the query with BM25, just as `AuthorMap`. The annotators for human evaluation are recruited from Amazon Mechanical Turk. The annotators should be from English-speaking countries and have HIT Approval Rates greater than 98%. The interface of human evaluation is shown in Fig. 5.

A.3 Preprocessing detail of `PerMSum`

To obtain user labels, we use metadata of the All the News dataset and the Amazon dataset. For the

You are a helpful assistant. Respond only with a JSON object including two key elements:

{
"content_analysis": a single-paragraph analysis of unique aspect and content preferences,
"style_analysis": a single-paragraph analysis of unique writing styles,
}

You are asked to analyze the distinctive features of User X's profile texts in comparison to those written by other users on similar topics. Specifically, you should first generate an analysis of unique aspect and content preferences that set User X apart. It should be written in a consecutive paragraph with less than 150 words (denoted as "content_analysis"). You should then generate an analysis of unique writing styles that set User X apart. It should be written in a consecutive paragraph with less than 150 words (denoted as "style_analysis").

****

Below are pairs of profile texts. Each pair describes the same product—one version is written by the User X, and the other by a different user.

Pair 1(written by User X): {profile document 1}

Pair 1(written by a different user): {comparative document 1}

****

Pair 2(written by User X): {profile document 2}

Pair 2(written by a different user): {comparative document 2}

****

...

Figure 1: Prompt used by ComPSum to generate structured analysis.

You are requested to generate a personalized summary with less than 100 words for multiple query texts about the same product for the User X based on the User X's profile texts and the anlysis of User X's aspect and content preference as well as writing style. The personalized summary should cover the main information of the query texts while mimic the aspect and content preference and writing style of the User X.

****

A list of profile texts for the User X are shown below:

<profile_document>

***********<end_of_list>***********

The analysis of the User X are shown below, where "content_analysis" denotes the analysis of aspect and content preference, "style_analysis" denotes the analysis of writing style:

<structured_analysis>

***********<end_of_analysis>***********

A list of query texts to be summarized are shown below:

<input_document_set

***********<end_of_list>***********

Please write a single personalized summary with less than 100 words for the query texts. The summary should only include contents from the query texts but not from the profile texts. Do not list sources of contents in the summary. Please directly output the personalized summary without any explanation. The summary should not be first person.

Figure 2: Prompt used by ComPSum to generate personalized summaries.

You are a helpful assistant. Respond only with a JSON object including two key elements:

{

"analysis": Reasoning behind your answer,

"answer": query texts more likely to be written by the author of the profile texts (Query Text 1 or Query Text 2 or Tie)

}

You are given a set of profile texts with a certain author and two query texts (Query Text 1 and Query Text 2) on the same topic.

Your task is to determine which query text is more likely to be written by the author of the profile texts solely based on writing style.

Specifically, first identify the differences in writing style between Query Text 1 and Query Text 2. Focus on linguistic features such as phrasal verbs, modal verbs, punctuation, rare words, affixes, quantities, humor, sarcasm, typographical errors, and misspellings. Then, detemine which query text's writing style is more closely aligned with the writing style presented in the profile texts. Please disregard the differences in contents and aspects during the comparison. If you cannot determine which query text is more likely to be written by the author of the profile texts solely based on content and aspect preference, output Tie.

The profile texts written by a certain author are shown below. The profile texts are delimited with two vertical bars: ||.

<profile_documents>

The query texts are shown below.

Query Text 1: <personalized_summary_for_user1>

****

Query Text 2: <personalized_summary_for_user2>

Figure 3: Prompt used by AuthorMap to evaluate writing style.

You are a helpful assistant. Respond only with a JSON object including two key elements:

{

"analysis": Reasoning behind your answer,

"answer": query texts more likely to be written by the author of the profile texts (Query Text 1 or Query Text 2 or Tie)

}

You are given a set of profile texts with a certain author and two query texts (Query Text 1 and Query Text 2) on the same topic.

Your task is to determine which query text is more likely to be written by the author of the profile texts solely based on content and aspect preference.

Specifically, first identify the differences in content and aspect focus between Query Text 1 and Query Text 2. Then, detemine which query text's content and aspect focus is more closely aligned with the content and aspect presented in the profile texts. Please disregard the differences in writing style during the comparison. If you cannot determine which query text is more likely to be written by the author of the profile texts solely based on content and aspect preference, output Tie.

The profile texts written by a certain author are shown below. The profile texts are delimited with two vertical bars: ||.

<profile_documents>

The query texts are shown below.

Query Text 1: <personalized_summary_for_user1>

****

Query Text 2: <personalized_summary_for_user2>

Figure 4: Prompt used by AuthorMap to evaluate content focus.

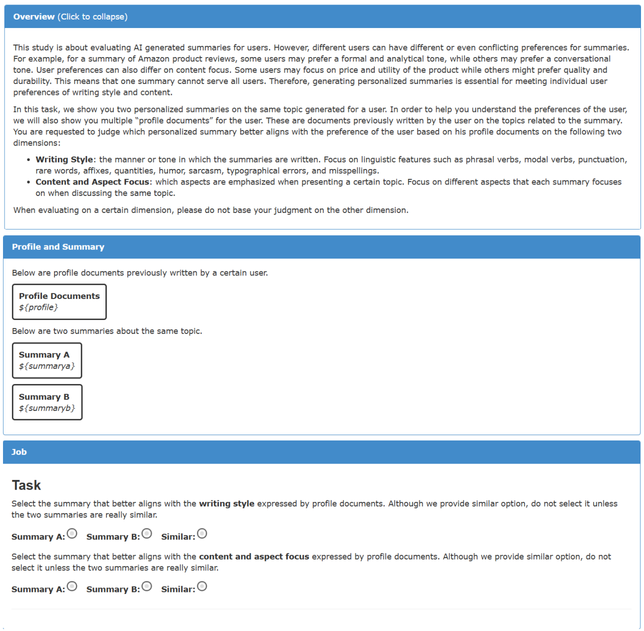

Figure 5: Interface for Human Evaluation

All the News dataset, around 60 percent of documents have non-empty author information. For these documents that have valid author information and have fewer than or equal to three authors, we label the first author as the author of the document. For the Amazon dataset, since all documents have exactly one author in their metadata, we just use the author in the metadata as the author of the document.

For the news domain, `PerMSum` clusters the news articles into document sets based on token overlap, named entities, and publishing dates, following Liu et al. (2022). Specifically, each news article is treated as a node in a graph. If two news articles are published within two days, share at least one named entity in their titles or first three sentences, have cosine similarities based on TF-IDF embedding over 0.30, there will be a line between these articles. The news articles are then clustered based on the maximum cliques of the graph. We filter out clusters where more than three articles are written by the same author to prevent the author from having too much impact on that cluster. To control the context length, we further divide clusters that contain more than 10 news articles into smaller clusters and truncate all news articles to 300 words.

A.4 Prompt for Paraphrasing

To test whether `AuthorMap` can independently measure writing style and content focus, we instruct Llama3.3-70b-Instruct to generate paraphrased documents that are originally written by certain users to follow the writing style of other users. We show the prompt for paraphrasing in Fig. 6

A.5 Experiment Details for Evalation of `ComPSum`

To evaluate factuality, we use FactScore (Min et al., 2023), which measures the proportion of atomic content units that are supported by input document sets. To evaluate relevance, we use G-Eval (Liu et al., 2023), which rates summaries based on their relevance from 1 to 5. For FactScore, we use Llama3.1-8b-Instruct to extract ACUs and judge whether ACUs are supported by input document sets. For G-Eval, we use Llama3.3-70b-Instruct to rate the relevance of summaries. To reduce the computation cost, we report the results for FactScore and G-EvaL on a subset of test set of `PerMSum`, which contains 25 percent of samples.

For the General baseline, we randomly sample two general summaries for each input document set without using any user profiles and use these sampled summaries for evaluation. For all other baselines, we retrieve 5 profile documents using BM25 similarly as `ComPSum` for a fair comparison. In the original implementation of DPL, comparable documents are retrieved based on the embeddings of user profiles. However, in `PerMSum`, not all documents have a valid user. Therefore, we instead retrieve comparable documents based on the embeddings of the documents themselves. For each retrieved profile document, we then select the four most dissimilar documents written by other users as the comparable documents.documents.

A.6 Evaluation of `ComPSum` using Gemma3

In the main content, `AuthorMap` uses Llama3.3-70b-Instruct as the LLM judge to evaluate `ComPSum`. However, `AuthorMap` can also use other LLMs as the LLM judge. We show the evaluation results when using Gemma-3-27b-it (Team, 2025) as the LLM judge in Tab. 9.

From the table, we observe that `ComPSum` still performs the best, similar as the results when using Llama3.3-70b-Instruct as the LLM judge (Tab. 5). The results show that `AuthorMap` is independent of the choice of the LLM judge.

A.7 Experiment Details for Ablation Study

For `ComPSum` w/o structure, we instruct the model to generate a profile summary for profile documents and their corresponding comparative documents without specifying that model should focus on the writing style or content focus. For a fair comparison, the token limit of the profile summary is the same as the structured analysis of `ComPSum`. For ComPSum w/multi. stage, we first instruct LLMs to generate a structured analysis of writing style and content focus for each retrieved profile document by comparing it with four comparable documents retrieved similarly as described in App. A.5. We then instruct LLMs to synthesize all structured analysis for each profile document into a final structured analysis. For a fair comparison, the token limit of the profile summary is the same as the structured analysis of `ComPSum`. The full results of ablation study are reported in Tab. 10. To reduce the computation cost, we report the results for FactScore and G-EvaL on subset of test set of `PerMSum`, which contains 25 percent of samples.

Given a set of profile texts with a certain writing style and a query document, please paraphrase the query document following the writing style of the profile texts. The paraphrased query document should have exatcly the same information as the original query document. The paraphrased query document should not include any additional information from the profile texts that are not mentioned by the original query document. The paraphrased query document should have the same length as the original query document.

Profile texts are shown below. The profile texts are delimited with two vertical bars: ||.

<profile_document>

Query Document: <query_document>

Please directly output the paraphrased query document

Figure 6: Prompt for paraphrasing.

A.8 Example Analysis Generated by `ComPSum`

In this section, we show additional examples of structured analysis generated by `ComPSum` and its ablation variant without comparative documents for writing style and content. We show the example analysis for writing style in Fig. 7. We show the example analysis for content in Fig. 8.

A.9 Example Summary Generated by `ComPSum`

In this section, we show example summaries generated by `ComPSum`and its ablation variant without comparative documents. We compare two summaries personalized for different users for the same input document set and show them in Fig. 9. We observe that summaries generated by `ComPSum` show more diverse styles and content focuses.

A.10 Ablation Study on Number of Retrieved Docuemnts

In the main content, `ComPSum` retrieves $m = 5$ profile documents for generating personalized summaries following Salemi et al. (2024). To examine the impact of the number of retrieved documents, we perform an ablation study on the number of retrieved documents and report the results in Tab. 11. From the table, we can observe that `ComPSum` performs the best when 5 documents are retrieved.

A.11 Failure Analysis of `ComPSum`

The most common failures of `ComPSum` are that generated personalized summaries can contain contents only presented in user profiles, but not in input documents. For example, for some input documents about Hulu ending its free service, a personalized summary generated by ComPSum can be 'Roku's public offering raised $252 million, valuing the company at $2 billion, while it competes against tech giants like Apple, Amazon, and Google in the streaming market. Meanwhile, YouTube announced a streaming TV service, offering live-streams and online video content for $35 a month, covering six accounts. Netflix expanded to 130 new countries, tripling its distribution. In a related development, Hulu will end its free, ad-supported service and shift its content to Yahoo View, a new platform offering free TV episodes with ads, while Hulu focuses on subscription tiers and a live TV service.' Only the last sentence in the summary is related to the event. We checked the unrelated sentences and found that they are actually from the user profile. To mitigate this issue, we instruct models to only include contents from input documents but not from the profile texts, as shown in Fig. 2.

ComPSum	
User 1: User X's writing style is characterized by a conversational tone, often incorporating direct quotes from celebrities and using a more informal language. This approach makes the content feel more approachable and relatable. In contrast, other users tend to write in a more formal and objective tone, focusing on delivering factual information without adding a personal touch. User X also tends to use a more narrative-driven approach, weaving together multiple sources and ideas to create a cohesive story, **which sets them apart from other writers who may rely on a more straightforward reporting style.**	**User 2:** User X's writing style is characterized by a clear and concise narrative voice, often incorporating direct quotes and specific details to support their points. Their texts tend to be well-structured and easy to follow, with a focus on conveying complex information in an accessible way. **Unlike other users, who may rely on sensational language or emotional appeals**, User X's tone is measured and informative, making their content feel more authoritative and trustworthy. This style suggests that User X prioritizes clarity and accuracy in their writing, setting **them apart from users who may prioritize other factors like entertainment value or clickbait headlines**.
w\o comp. doc.	
User 1: The writing style of User X appears to be conversational and engaging, with a focus on storytelling and anecdotal evidence. The use of quotes from celebrities and behind-the-scenes details adds a sense of intimacy and immediacy to the articles. The tone is generally lighthearted and celebratory, particularly in the pieces about celebrity collaborations and magazine covers. However, when discussing more serious topics like gun violence and aging, the tone shifts to one of gravity and concern. Overall, the writing style is approachable and accessible, suggesting that User X aims to inform and entertain their audience	**User 2:** User X's writing style is characterized by a conversational tone and a focus on storytelling. They often use anecdotes and quotes from celebrities to illustrate their points, making their content feel more relatable and engaging. The text is also well-structured and easy to follow, with a clear and concise writing style. User X tends to use a more informal tone, often incorporating colloquial expressions and contractions, which creates a sense of familiarity with the reader. Overall, the writing style is approachable and accessible, making it easy for readers to stay engaged and interested in the content.

Figure 7: Example analysis of writing style generated by `ComPSum` and its ablation variant with out comparative documents. The structured analysis generated by `ComPSum` additionally includes comparison with other users (in **bold**), which helps in better personalization.

ComPSum	
User 1: User X tends to focus on the positive and empowering aspects of celebrity news, often highlighting the personal growth and self-acceptance of public figures, such as Miley Cyrus embracing her past, Kim Kardashian West's loving family moments, and Cameron Diaz's openness about aging. **In contrast, other users often focus on controversy, drama, or sensationalized aspects of celebrity news.** User X also tends to emphasize the intersection of celebrity culture and social issues, such as gun violence and body positivity, in a way that is uplifting and inclusive.	**User 2:** User X's profile texts stand out for their in-depth analysis of current events, particularly in the realms of feminism and social justice. They delve into the nuances of the topics, often referencing specific quotes and details from the individuals involved. This level of depth and specificity suggests that User X values accuracy and context when discussing complex issues. Additionally, their texts often highlight the personal stories and experiences of individuals, such as Miley Cyrus and Bella Thorne, to illustrate the human impact of these issues. This approach creates a more empathetic and relatable tone**, setting User X apart from other users who may focus on more sensational or surface-level aspects of the news**.
w\o comp. doc.	
User 1: The content preferences of User X seem to revolve around entertainment, celebrity culture, and social issues. The topics of the provided texts include Miley Cyrus's past controversy, the Kardashian-Jenner family, and celebrities speaking out on social issues. The content also touches on fashion, with articles about celebrity appearances in Calvin Klein campaigns and magazine covers. Additionally, there is a focus on female empowerment and self-expression, as seen in the discussion of Miley Cyrus's unapologetic attitude and Cameron Diaz's openness about aging. This suggests that User X is interested in celebrity news, social justice, and feminist themes.	**User 2:** The content preferences of User X appear to be centered around social justice, feminism, and celebrity culture. The user frequently discusses topics such as body positivity, sexual harassment, and assault, often using high-profile figures like Miley Cyrus and Bella Thorne as examples. User X also seems to be interested in news and updates on social movements like #TimesUp and the prosecution of Harvey Weinstein. Additionally, they have a strong interest in celebrity relationships and personal stories, as evident from their coverage of Miley Cyrus's marriage and new music releases. Overall, User X's content preferences suggest a focus on current events, social issues, and celebrity culture.

Figure 8: Example analysis of content focus generated by `ComPSum` and its ablation variant with out comparative documents. The structured analysis generated by `ComPSum` additionally includes comparison with other users (in **bold**), which helps in better personalization.

	News					Review				
	style	content	fact.	rele.	overall	style	content	fact.	rele.	overall
					Llama3.1-8b-Instruct					
General	49.36	42.84	97.86	98.51	67.19	49.98	45.43	98.58	93.94	67.72
RAG	54.45	51.06	98.00	96.05	71.52	56.12	55.24	98.16	92.71	72.88
CICL	55.13	52.29	97.34	95.47	71.94	59.33	58.83	96.87	88.91	74.05
RAG+Summary	56.91	53.50	98.03	97.17	7 73.38	60.07	59.58	97.67	92.02	75.31
DPL	54.87	52.50	97.91	96.30	72.19	59.69	59.91	97.05	87.68	74.27
Rehersal	99.66	99.49	23.16	28.75	50.69	97.97	97.96	57.28	37.62	67.43
ComPSum	58.81	57.29	98.01	95.32	**74.90**	62.01	61.46	98.03	91.99	**76.57**
					Qwen2.5-14B-Instruct					
General	49.60	44.32	98.21	98.71	67.94	50.26	45.52	97.89	93.55	67.65
RAG	53.05	52.03	98.07	96.77	71.54	54.54	55.04	97.76	91.76	72.04
CICL	55.38	50.95	97.11	96.68	71.74	57.40	57.17	96.95	88.91	72.93
RAG+Summary	56.91	52.33	98.17	97.93	73.15	60.71	61.42	97.40	90.97	75.81
DPL	54.87	50.74	98.04	97.37	71.80	57.46	57.46	96.83	89.42	73.12
Rehersal	98.81	98.98	22.86	26.94	49.54	94.34	97.45	53.20	33.63	63.68
ComPSum	59.07	58.14	97.96	96.58	**75.50**	63.21	64.07	96.51	89.98	**77.01**
					Llama3.3-70b-Instruct					
General	45.34	33.31	98.75	99.03	61.99	47.67	39.48	98.36	94.17	64.62
RAG	53.82	45.08	98.59	97.98	69.58	56.30	53.26	98.76	93.16	72.48
CICL	52.47	48.14	98.55	98.03	70.28	56.26	53.19	98.70	92.79	72.36
RAG+Summary	54.41	47.63	98.70	98.08	70.77	57.64	57.27	98.68	93.43	74.28
DPL	55.52	47.80	98.81	98.17	71.23	55.81	53.15	98.50	93.10	72.22
Rehersal	99.32	99.83	21.74	24.90	48.13	95.67	97.11	62.03	33.81	66.44
ComPSum	56.78	48.81	98.64	98.03	**71.95**	67.69	59.14	98.70	93.35	**77.93**

Table 9: Evaluation results of ComPSum when using Gemma-3-27b-it as the LLM judge. A higher value indicates better performance. The best-performing method based on overall score is **bolded**. The evaluation results using Gemme-3-27b-it is consistent with evaluation results using Llama3.3-70b-Instruct in Tab. 5.

	News					Review				
	style	content	fact.	rele.	overall	style	content	fact.	rele.	overall
					Llama3.1-8b-Instruct					
ComPSum	59.75	53.94	98.01	95.32	**74.07**	59.13	57.89	98.03	91.99	74.54
w/o comp. doc.	57.03	50.69	98.16	97.29	71.09	56.08	57.21	97.07	92.28	73.24
w/o structure	58.64	47.33	98.08	95.62	71.43	68.35	67.26	95.58	86.11	**78.43**
w/ sim. comp.	59.75	47.29	97.95	96.71	73.33	58.51	57.17	97.74	91.80	74.00
w/ multi. stage	55.51	45.42	98.16	96.31	69.87	59.71	59.93	97.57	91.34	75.15
					Qwen2.5-14B-Instruct					
ComPSum	57.92	57.08	97.96	96.58	**74.69**	60.36	63.92	96.51	89.98	76.08
w/o comp. doc.	55.51	48.73	98.03	96.25	71.16	60.22	61.74	97.98	91.17	76.10
w/o structure	53.56	45.42	97.84	97.12	68.32	60.76	63.19	96.59	89.39	75.88
w/ sim. comp.	57.92	51.44	97.83	98.05	71.57	63.23	64.50	96.15	89.83	**76.85**
w/ multi. stage	52.16	44.62	98.23	96.98	67.23	59.75	59.75	96.73	90.50	74.43
					Llama3.3-70b-Instruct					
ComPSum	53.90	45.38	98.64	98.03	**69.74**	59.27	51.87	98.70	93.35	72.95
w/o comp. doc.	51.53	41.06	98.78	98.73	67.40	57.53	50.74	98.67	93.55	72.05
w/o structure	53.81	44.96	98.86	98.59	69.69	64.75	55.97	98.26	92.14	**75.69**
w/ sim. comp.	53.90	43.69	98.69	98.81	69.22	59.02	51.72	98.73	93.50	72.86
w/ multi. stage	50.21	41.53	98.72	98.75	67.14	59.03	51.44	98.62	92.94	72.63

Table 10: Full ablation results of ComPSum. A higher value indicates better performance. The best-performing method based on overall score is **bolded**.

	News					Review				
	style	content	fact.	rele.	overall	style	content	fact.	rele.	overall
					Llama3.1-8b-Instruct					
ComPSum	59.75	53.94	98.01	95.32	**74.07**	59.13	57.89	98.03	91.99	**74.54**
ComPSum(m=10)	55.17	44.83	98.18	96.71	70.12	55.17	51.07	98.02	92.06	72.36
ComPSum(m=2)	58.69	50.25	97.54	95.56	72.37	59.02	55.10	97.91	92.24	74.28

Table 11: Ablation results of ComPSum with different number of retrieved documents. A higher value indicates better performance. The best-performing method based on overall score is **bolded**.

ComPSum	
User 1: It's a double-edged sword for Julian Assange - a UN ruling says he's been unlawfully detained, but the UK and Sweden aren't budging. The UN panel's decision is a "vindication" for Assange, but a British government spokesman called it "ridiculous." Assange has been living in the Ecuadorian embassy since 2012, avoiding arrest and extradition to Sweden over rape allegations. He's hailed the ruling as a "victory of historical importance," but the UK has made it clear they'll arrest him if he leaves the embassy. It's a stalemate that's left Assange stuck in limbo.	**User 2:** A United Nations panel has ruled that Julian Assange has been arbitrarily detained at the Ecuadorian embassy in London since his arrest in 2010 over allegations of sex crimes. The UN Working Group on Arbitrary Detention called on Sweden and the UK to let Assange move freely and ensure his safety, and to provide compensation. However, British authorities have stated they will arrest and extradite Assange to Sweden if he leaves the embassy. Assange has hailed the decision as a "vindication" and a "victory of historical importance."
w\o comp. doc.	
User 1: A United Nations panel has ruled that Julian Assange has been arbitrarily detained in the Ecuadorian embassy in London since 2010. The UN Working Group on Arbitrary Detention called on Sweden and the UK to respect Assange's rights, including his freedom of movement and physical integrity. The ruling is not legally binding, but Assange sees it as a "vindication" and a "victory of historical importance." The UK and Sweden have rejected the decision, with the UK's Foreign Secretary calling it "ridiculous." Assange remains in the embassy, where he has been living for over three years.	**User 2:** A United Nations panel has ruled that WikiLeaks founder Julian Assange has been arbitrarily detained by Sweden and the UK, with the decision calling for his release and compensation. Assange, who has been living in the Ecuadorian embassy in London since 2012, hailed the ruling as a "vindication" and a "victory of historical importance." However, the UK and Swedish governments have rejected the ruling, with the British Foreign Secretary describing it as "ridiculous" and stating that Assange will be arrested if he leaves the embassy.

Figure 9: Example summaries generated by ComPSum and its ablation variant with out comparative documents.