

# Integrating Summarization and Retrieval for Enhanced Personalization via Large Language Models

Christopher Richardson\*  
Georgia Institute of Technology  
Atlanta, GA, USA  
crichardson332@gmail.com

Yao Zhang\*  
Amazon Alexa AI  
Austin, TX, USA  
yaozhanq@amazon.com

Kellen Gillespie  
Amazon Alexa AI  
Seattle, WA, USA  
kelleng@amazon.com

Sudipta Kar  
Amazon Alexa AI  
Seattle, WA, USA  
sudipkar@amazon.com

Arshdeep Singh  
Amazon Alexa AI  
Seattle, WA, USA  
adpsingh@amazon.com

Zeynab Raeesy  
Amazon Alexa AI  
Seattle, WA, USA  
raeesyzr@amazon.com

Omar Zia Khan  
Amazon Alexa AI  
Seattle, WA, USA  
ozkhan@amazon.com

Abhinav Sethy  
Amazon Alexa AI  
Seattle, WA, USA  
sethya@amazon.com

## ABSTRACT

Personalization, the ability to tailor a system to individual users, is an essential factor in user experience with natural language processing (NLP) systems. With the emergence of Large Language Models (LLMs), a key question is how to leverage these models to better personalize user experiences. To personalize a language model's output, a straightforward approach is to incorporate past user data into the language model prompt, but this approach can result in lengthy inputs exceeding limitations on input length and incurring latency and cost issues. Existing approaches tackle such challenges by selectively extracting relevant user data (i.e. selective retrieval) to construct a prompt for downstream tasks. However, retrieval-based methods are limited by potential information loss, lack of more profound user understanding, and cold-start challenges. To overcome these limitations, we propose a novel summary-augmented approach by extending retrieval-augmented personalization with task-aware user summaries generated by LLMs. The summaries can be generated and stored offline, enabling real-world systems with runtime constraints like voice assistants to leverage the power of LLMs. Experiments show our method with 75% less of retrieved user data is on-par or outperforms retrieval augmentation on most tasks in the LaMP personalization benchmark. We demonstrate that offline summarization via LLMs and runtime retrieval enables better performance for personalization on a range of tasks under practical constraints.

\*Both authors contributed equally to this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

Conference'17, July 2017, Washington, DC, USA

© 2023 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM... \$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## CCS CONCEPTS

• **Computing methodologies** → **Natural language generation**; *Information extraction*; • **Information systems** → **Personalization**.

## KEYWORDS

LLM, Personalization, Summarization, NLP, Chatbot, Voice Assistant, Conversational AI

## ACM Reference Format:

Christopher 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. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 1 INTRODUCTION

As virtual assistants and other natural language processing (NLP) systems become increasingly integrated into our daily lives, personalization has become an essential factor in user experience. Tailoring virtual assistant interactions and NLP model outputs to individual users' preferences, styles, needs, and contexts is essential in improving the performance of these systems to make them more natural and conversational.

Traditional personalization methods, such as collaborative filtering [17], deep neural networks [8], deep interest network [22] and their variations [4, 12], have enhanced user experiences in recommendation systems. These methods leverage historical user behavior data to make personalized recommendations, offering a practical and effective solution for various domains. Despite their success, these methods still struggle with the cold-start problem, where new users lack sufficient behavior history, leading to sub-optimal recommendations. The cold-start problem highlights the need for alternative approaches.

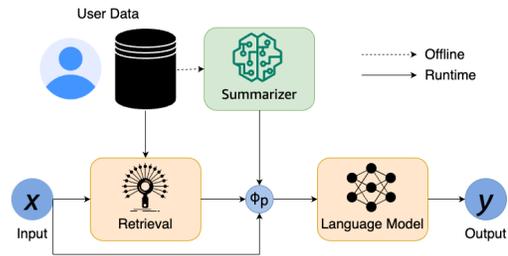
Large Language Models (LLMs) represent a promising avenue for advancing personalization techniques. LLMs have demonstrated

remarkable capabilities in understanding context and generating coherent text [2]. By incorporating knowledge about users, LLMs can potentially enhance personalization by capturing subtle user preferences, but how to capture the full spectrum of user preferences in a personalized manner remains a challenge. To personalize a language model output, a straightforward approach is to incorporate user data into the language model prompt. However, incorporating a comprehensive view of customer preferences with long-term historical user data into the prompt may exceed the input length limitations of language models and result in considerable increases in inference cost. Further, language models tend to degrade with lengthy contexts [3]. To address these concerns, a personalized retrieval augmentation framework was proposed [16]. This framework selectively extracts relevant user data to construct prompts for downstream language models. Recent work has also shown promise in combining retrieval approaches with LLMs to improve performance in recommender systems [5, 11, 20], as well as general NLP tasks [9, 10, 13, 14, 19]. However, retrieval-based methods have constraints in potential information loss, lack the ability to comprehend user data on a more profound level, and may suffer from the cold-start problem.

Our research aims to address the aforementioned limitations of both traditional personalization methods and retrieval-based methods with LLMs by proposing the hybrid approach shown in Figure 1. By integrating retrieval techniques with LLM-generated summaries of user data, we intend to create a more robust personalized system. To prevent information loss, the user summary offers contextual information at a higher level of abstraction for the downstream task. To understand user data on a more profound level, the summary generation is aware of the task and incorporates this information in the prompt for summary generation. For example, for a personalized paraphrase text generation task, the summary model is instructed by a prompt to pay attention to the user writing style in addition to the semantic content. Also, this hybrid model could overcome the cold-start problem and provide personalized outputs even in data-sparse scenarios by providing user summaries for new users based on available user data from other applications or user’s self description. The summaries in our approach can be generated offline and stored, ensuring negligible increased runtime latency and enabling systems with runtime constraints to leverage the power of LLMs into real-work online applications, such as voice assistant scenarios.

We demonstrate our method of integrating summarization and retrieval on a publicly available Language Model Personalization (LaMP) benchmark [16], including both text classification and generation tasks across a variety of domains. Experiments show our method achieves comparable or better performance compared to retrieval augmentation on most tasks. With our method, the retrieval component can use 75% less of retrieved user data without sacrificing performance on five out of six tasks, and achieves superior performance on two tasks.

In summary, our main contributions are as follows. First, we propose augmenting traditional retrieval-based personalization methods with LLMs’ summarization of user data to address the limitations of existing methods: potential information loss, the inability to understand user data at a high level, and the cold-start



**Figure 1: Personalization is achieved by combining runtime-retrieved samples with an offline-generated user summary. Given a textual input  $x$  that describes a task in natural language, the goal is to generate a personalized output  $y$  for users. The retrieval model identifies the most relevant items from user data, and the retrieved items along with the offline user summary and  $x$  form the basis for creating a prompt. This prompt is constructed using a prompt construction function  $\phi_p$ .**

challenge. Our method enables powerful LLMs to provide comprehensive information about users with no additional runtime latency. Further, we implemented our proposed approach and conducted experiments on a language model personalization benchmark dataset LaMP with 6 public tasks. With promise shown in our experiment results, we envision a personalized system that better caters to individual user preferences especially for new users by integrating summarization via LLMs and retrieval.

## 2 METHODOLOGY

### 2.1 Problem Formulation

The problem formulation follows LaMP [16]: given a textual input  $x$  that describes a task in natural language, we want to generate a personalized output  $y$  for user  $u$ . The goal is thus to learn the distribution  $p(y|x, u)$ .

**2.1.1 Baseline.** Our baseline is a retrieval-augmented method that follows a retrieve-then-model paradigm [16]. For retrieval, a manually defined query generation function  $\phi_q(x)$  is first used to help extract salient information from  $x$  as inputs to the retrieval model  $\mathcal{R}(\phi_q(x), P_u, k)$ . The retrieval model returns the top- $k$  relevant items from the user profile  $P_u$  using the retrieval query  $\phi_q(x_i)$ , and the returned items are used to construct a prompt for a downstream model using a prompt construction model  $\phi_p$ . The input to the downstream language model is as follows.

$$\bar{x}_i = \phi_p(x_i, \mathcal{R}(\phi_q(x_i), P_u, k)) \quad (1)$$

The downstream language model is fine-tuned on the dataset  $\{\bar{x}_i, y_i\}$ .

### 2.2 Integrating Summarization and Retrieval

Runtime constraints can limit the number of user data retrieved by  $\mathcal{R}$  that can be utilized. In this work, we consider scenarios where there are both latency considerations as well as input length limits for the downstream model. To improve performance without adding runtime latency, we introduce a summary of the user data,  $s_u$ , to

**Table 1: Description of LaMP tasks and data.**

LaMP-1	<b>Task:</b> Citation Identification (binary choice) <b>User Profile:</b> Scholarly article titles and abstracts <b>Metric(s):</b> Accuracy
LaMP-2	<b>Task:</b> News Categorization (classification) <b>User Profile:</b> Categorized articles published <b>Metric(s):</b> Accuracy and F1
LaMP-3	<b>Task:</b> Product Rating (classification) <b>User Profile:</b> Product reviews and scores <b>Metric(s):</b> Mean Absolute Error (MAE) and Root Mean Square Error (RMSE)
LaMP-4	<b>Task:</b> News Headline Generation (text generation) <b>Profile:</b> News articles and their headlines <b>Metric(s):</b> ROUGE-1 and ROUGE-L
LaMP-5	<b>Task:</b> Scholarly Title Generation (text generation) <b>Profile:</b> Scholarly article titles and abstracts <b>Metric(s):</b> ROUGE-1 and ROUGE-L
LaMP-7	<b>Task:</b> Tweet Paraphrasing (text generation) <b>Profile:</b> Tweets <b>Metric(s):</b> ROUGE-1 and ROUGE-L

augment the retrieved data:

$$\bar{x}_i = \phi_p(x_i, \mathcal{R}(\phi_q(x_i), P_u, k), s_u) \quad (2)$$

Our approach involves using LLMs to summarize salient information from  $u$  as it relates to optimizing  $p(y|x, u)$ . We use instruction-tuned models to generate an abstractive summary of user data:

$$s_u = \text{LLM}(P_u) \quad (3)$$

An overview of our method is shown in Figure 1. The summaries can be generated offline and stored along with the user data itself. At runtime, the retrieval algorithm retrieves the top- $k$  profile entries and concatenates them with the task input and the summary to create the full context for the downstream language model, which is fine-tuned using the standard language modeling loss against the output  $y$ .

## 3 EXPERIMENTS

### 3.1 Datasets and Evaluation Metrics

LaMP is a public benchmark dataset for training and evaluating methods for personalization with language models [16]. It consists of seven personalization NLP tasks, including three classification tasks and four text generation tasks. Data for each task include input text, reference completion text as ground truth output, and a user profile consisting of an array of items with textual data. A brief description of each task and their evaluation metrics are shown in Table 1. We have excluded task LaMP-6 from our study as it relies on private data to which we do not have access. The LaMP benchmark organizes the data in both user-based separation and time-based separation. In our study, we utilize the user-based separation approach to address cold-start issues for new users.

### 3.2 Experimental Setup

Following LaMP, we used FlanT5-base [7] as our downstream model. This model demonstrated satisfactory runtime performance in our experiments (approximately 125 milliseconds per sample when we included as many user data as possible within the 512-token limit of the input length) and, as shown in the LaMP experiments, it achieved superior performance to that achieved in zero-shot experiments with FlanT5-XXL and ChatGPT [16]. For all experiments, we used the same settings reported in LaMP: a learning rate of  $5 \times 10^{-5}$ , weight decay of  $10^{-4}$ , warmup ratio of 0.05, and a beam size of 4, and we trained for 10 epochs for text classification (tasks 1-3) and 20 epochs for text generation (tasks 4, 5, and 7).

In our experiments, we utilized the BM25 retrieval algorithm [15] due to its speed and performance. We found neural methods like Contriever [1] to be too slow for voice assistant scenarios, which induced approximately 10-30 seconds of latency per sample while not significantly outperforming BM25 on many LaMP tasks.

We experimented with two instruction-tuned models for generating summaries. The first is Vicuna [21], a 13-billion parameter model distilled from LLaMA [18]. The second model is ChatGPT using OpenAI’s API with the gpt-3.5-turbo-16k model. Vicuna has a context length of 2048 tokens, while ChatGPT’s limit is 16,384. Prompts used for generating summaries are shown in Table 2. For tasks 2 and 3, to achieve good performance given the simplicity of the tasks, we constrained the summarization model to output according to a strict template shown in the Table 2, therefore ChatGPT summaries were not included for these tasks.

We compare our methods to the retrieval-only baselines, using  $k$  values of 0, 1, and 4 for the baselines. We observed  $k = 4$  to be the limit for some tasks given the context length of FlanT5 (512 tokens). Thus, we had to reduce the number of retrieved samples to fit the summaries into the input of FlanT5 without truncating and chose  $k = 1$  (for direct comparison with the baseline), as well as  $k = 0$  to investigate the impact of summaries alone (no retrieval). We report means of three repeated runs of each experiment for comparison with statistical significance.

## 4 RESULTS AND DISCUSSION

Table 3 shows the results of our methods for both summary models alongside the baselines across various NLP tasks. Our experiments prove our summary-augmented method with  $k = 1$  is on-par or outperforms the retrieval-only baseline with  $k = 4$  on most tasks (reducing the amount of retrieved user data by 75%). As the bold results indicate the best results among compared experiments for each task, our methods (Vicuna Summ. and GPT-3.5 Summ.) outperform baselines consistently on tasks 1 and 2 at a statistical significance level with  $p$ -value  $< 0.05$ , and we achieve comparable performance on tasks 3, 5, and 7 with no statistically significant difference.

On LaMP-1 task, our method GPT-3.5 Summ. outperforms the baseline with  $k = 4$  using only offline generated summaries ( $k = 0$ ). Furthermore, Our method GPT-3.5 Summ. yields better results compared to  $k = 1$  baseline on all tasks. Worth noting that ChatGPT summaries mostly outperform those provided by Vicuna, likely due to the disparity in model size.

We observed a gap between Vicuna Summ. and GPT-3.5 Summ. attributed to the differing quality of the offline summaries.. Despite

**Table 2: Prompts used for summarization. Additional tokens were used for Vicuna summaries to match the expected prompt format for that model but the content was the same for Vicuna and ChatGPT.**

Task	Prompt
LaMP-1	Write a summary, in English, of the research interests and topics of a researcher who has published the following papers. Only generate the summary, no other text.
LaMP-2	Look at the following past articles this journalist has written and determine the most popular category they write in. Answer in the following form: most popular category: <category>
LaMP-3	Based on this user’s past reviews, what are the most common scores they give for positive and negative reviews? Answer in the following form: most common positive score: <most common positive score>, most common negative score: <most common negative score>
LaMP-4	Given this author’s previous articles, try to describe a template for their headlines. I want to be able to accurately predict the headline gives one of their articles. Be specific about their style and wording, don’t tell me anything generic.
LaMP-5	Given this author’s previous publications, try to describe a template for their titles. I want to be able to accurately predict the title of one of the papers from the abstract. Only generate the template description, nothing else.
LaMP-7	Given this person’s previous tweets, try to describe a template for their tweets. I want to take a generic sentence and rephrase it to sound like one of their tweets, with the same style/punctuation/capitalization/wording/tone/etc. as them. Only give me the template description, nothing else.

**Table 3: Results for FlanT5-base model fine-tuned on LaMP benchmark tasks. Baseline: retrieval of k user data entries; Vicuna Summ.: Baseline + summary of user data generated by vicuna; GPT-3.5 Summ.: Baseline + summary of user data generated by GPT-3.5. Underline means summary improved the corresponding baselines with the same k, and bold means the best results among compared experiments for each task. For all metrics, higher is better except in the case of MAE and RMSE used for LaMP-3.**

Task	Metric	Baseline			Vicuna Summ.		GPT-3.5 Summ.	
		k = 0	k = 1	k = 4	k = 0	k = 1	k = 0	k = 1
LaMP-1: Personalized Citation Identification	Accuracy	0.516	0.650	0.709	<u>0.704</u>	<u>0.728</u>	<u>0.738</u>	<b>0.743</b>
LaMP-2: Personalized News Categorization	Accuracy	0.731	0.782	0.807	<u>0.801</u>	<b>0.814</b>	N/A	
	F1	0.511	0.573	0.574	<u>0.550</u>	<b>0.601</b>		
LaMP-3: Personalized Product Rating	MAE	0.311	0.284	0.280	<u>0.305</u>	<b>0.277</b>	N/A	
	RMSE	0.626	0.595	<b>0.593</b>	0.632	<u>0.594</u>		
LaMP-4: Personalized News Headline Generation	ROUGE-1	0.152	0.177	<b>0.188</b>	<u>0.157</u>	0.173	<u>0.170</u>	<u>0.181</u>
	ROUGE-L	0.137	0.162	<b>0.173</b>	<u>0.142</u>	0.159	<u>0.155</u>	<u>0.166</u>
LaMP-5: Personalized Scholarly Title Generation	ROUGE-1	0.424	0.447	<b>0.448</b>	<u>0.426</u>	0.447	0.424	<b>0.448</b>
	ROUGE-L	0.382	0.408	<b>0.409</b>	<u>0.386</u>	0.408	<u>0.383</u>	<b>0.409</b>
LaMP-7: Personalized Tweet Paraphrasing	ROUGE-1	0.510	0.502	<b>0.513</b>	0.510	<u>0.512</u>	<u>0.510</u>	<u>0.512</u>
	ROUGE-L	0.455	0.448	0.459	0.455	<u>0.459</u>	<u>0.456</u>	<b>0.460</b>

studies showing Vicuna achieving up to 90% the performance of ChatGPT [6], our results suggest that Vicuna did not perform as well as GPT-3.5 on task LaMP-1 and 4. To assess the summary quality, we provide examples in the Appendix. A.

While we have shown promise in combining offline summaries with runtime retrieval for personalization, there are a few limitations to this work. For one, the data and tasks provided in the LaMP benchmark are simplistic and narrow in scope. More work is needed to assess the potential of our method on more realistic user data. Also, the benefits of summarization can be improved by fine-tuning a larger language model and end-to-end training for the tasks.

## 5 CONCLUSION

This paper introduces a novel method for augmenting retrieval with offline summarization for improving personalization in various NLP tasks. We implemented our method and achieved comparable or better performance on most NLP tasks in the LaMP personalization benchmark while reducing the amount of retrieved user data by 75%. In some cases, we even achieved superior performance after removing retrieval entirely, showing an advantage for sparse data scenarios, such as the cold-start problem. Our method efficiently leverages offline summary generation and is suitable for runtime constrained applications such as voice assistants.

## ACKNOWLEDGMENTS

Thanks to Camden Parsons, Farideh Tavazoei, Xingzhi (Jacky) Guo for reviewing and providing feedbacks on our initial paper draft.

## REFERENCES

- [1] Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. 2022. Task-aware retrieval with instructions. *arXiv preprint arXiv:2211.09260* (2022).
- [2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901. [https://proceedings.neurips.cc/paper\\_files/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf)
- [3] Jiuhai Chen, Lichang Chen, Chen Zhu, and Tianyi Zhou. 2023. How Many Demonstrations Do You Need for In-context Learning? *arXiv:2303.08119* [cs.AI]
- [4] Qiwei Chen, Changhua Pei, Shanshan Lv, Chao Li, Junfeng Ge, and Wenwu Ou. 2021. End-to-End User Behavior Retrieval in Click-Through Rate Prediction Model. <http://arxiv.org/abs/2108.04468> *arXiv:2108.04468* [cs].
- [5] Zheng Chen. 2023. PALR: Personalization Aware LLMs for Recommendation. *arXiv preprint arXiv:2305.07622* (2023).
- [6] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%\* ChatGPT Quality. <https://lmsys.org/blog/2023-03-30-vicuna/>
- [7] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416* (2022).
- [8] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*. New York, NY, USA.
- [9] Shiran Dudy. 2022. Personalization and Relevance in NLG. In *Companion Proceedings of the Web Conference 2022*. 1178–1178.
- [10] Lucie Flek. 2020. Returning the N to NLP: Towards contextually personalized classification models. In *Proceedings of the 58th annual meeting of the association for computational linguistics*. 7828–7838.
- [11] Jinming Li, Wentao Zhang, Tian Wang, Guanglei Xiong, Alan Lu, and Gerard Medioni. 2023. GPT4Rec: A generative framework for personalized recommendation and user interests interpretation. *arXiv preprint arXiv:2304.03879* (2023).
- [12] Pi Qi, Xiaoqiang Zhu, Guorui Zhou, Yujing Zhang, Zhe Wang, Lejian Ren, Ying Fan, and Kun Gai. 2020. Search-based User Interest Modeling with Lifelong Sequential Behavior Data for Click-Through Rate Prediction. <https://doi.org/10.48550/arXiv.2006.05639> *arXiv:2006.05639* [cs, stat].
- [13] Hongjin Qian, Xiaohe Li, Hanxun Zhong, Yu Guo, Yueyuan Ma, Yutao Zhu, Zhanliang Liu, Zhicheng Dou, and Ji-Rong Wen. 2021. Pchatbot: a large-scale dataset for personalized chatbot. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 2470–2477.
- [14] Omid Rafeian and Hema Yoganarasimhan. 2022. AI and Personalization. *Available at SSRN 4123356* (2022).
- [15] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gattford, et al. 1995. Okapi at TREC-3. *Nist Special Publication Sp 109* (1995), 109.
- [16] Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. 2023. LaMP: When Large Language Models Meet Personalization. *arXiv preprint arXiv:2304.11406* (2023).
- [17] J. Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative Filtering Recommender Systems. In *The Adaptive Web: Methods and Strategies of Web Personalization*, Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl (Eds.). Springer, Berlin, Heidelberg, 291–324. [https://doi.org/10.1007/978-3-540-72079-9\\_9](https://doi.org/10.1007/978-3-540-72079-9_9)
- [18] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. [n. d.]. LLaMA: open and efficient foundation language models, 2023. URL <https://arxiv.org/abs/2302.13971> ([n. d.]).
- [19] Yuwei Wu, Xuezhe Ma, and Diyi Yang. 2021. Personalized response generation via generative split memory network. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1956–1970.
- [20] Jiajing Xu, Andrew Zhai, and Charles Rosenberg. 2022. Rethinking personalized ranking at Pinterest: An end-to-end approach. In *Proceedings of the 16th ACM Conference on Recommender Systems*. 502–505.
- [21] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *arXiv preprint arXiv:2306.05685* (2023).
- [22] Guorui Zhou, Chengru Song, Xiaoqiang Zhu, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep Interest Network for Click-Through Rate Prediction. <https://doi.org/10.48550/arXiv.1706.06978> [cs, stat].

## A SUMMARY EXAMPLES

We are presenting examples in which the LaMP baseline model exhibited errors, while our methods demonstrated superior performance in Example 1, and in which the LaMP baseline model and Vicuna Summ. exhibited errors while GPT-3.5 Summ. demonstrated superior performance in Example 2.

### A.1 Example 1

#### Task: LaMP-1

**Input:** For an author who has written the paper with the title "QuickSense: Fast and energy-efficient channel sensing for dynamic spectrum access networks", which reference is related? Just answer with [1] or [2] without explanation. [1]: "A first look at cellular network performance during crowded events" [2]: "On the complexity of scheduling in wireless networks"

#### Output: [1]

**Retrieved user data:** On the Stability of Rational, Heterogeneous Interdomain Route Selection

A decision-theoretic approach to resource allocation in wireless multimedia networks

A general algorithm for interference alignment and cancellation in wireless networks

Enterprise social network analysis and modeling: A tale of two graphs

#### Baseline Prediction: [2]

**User Summary by Vicuna:** (The original output is in Chinese and here is the translation:) Network Traffic Analysis

Wireless Communication

Software-Defined Networking

Traffic Routing

Low-Latency Communication

Multimedia Networks

Network Security

Error Handling in Networks

Sensor Networks

Multi-Path Routing

IP Routing

Traffic Analysis Tools

#### Vicuna Summ. Prediction: [1]

**User Summary by GPT-3.5:** The research interests of this researcher span several areas in computer networking and wireless communications. They have published papers on topics such as unified cellular and ad-hoc network architecture, optimization of wireless mesh networks, networking analytics, incentive-compatible routing protocols, software-defined networking, and network security. They have expertise in optimizing network throughput, designing scalable network architectures, developing game-theoretic and cryptographic techniques for wireless networks, and leveraging

machine learning and data analytics for network monitoring and diagnosis. Their recent works also focus on addressing the latency and energy efficiency challenges in wireless multimedia networks and leveraging software-defined networking for next-generation cellular networks. In summary, this researcher's work revolves around developing novel algorithms, protocols, and systems that improve the performance, resilience, and security of networked systems.

**GPT-3.5 Summ. Prediction:** [1]

## A.2 Example 2

### Task: LaMP-1

**Input:** For an author who has written the paper with the title "Visual-audio integration for user authentication system of partner robots", which reference is related? Just answer with [1] or [2] without explanation. [1]: "Pronunciation Modeling for Improved Spelling Correction" [2]: "Path following algorithm for highly redundant manipulators"

**Output:** [1]

**Retrieved user data:** Advancing Matrix Completion by Modeling Extra Structures beyond Low-Rankness

Improved Asymmetric Locality Sensitive Hashing (ALSH) for Maximum Inner Product Search (MIPS)

Real-Time Implementation of Improved State-Space MPC for Air Supply in a Coke Furnace

On Practical Algorithms for Entropy Estimation and the Improved Sample Complexity of Compressed Counting

**Baseline Prediction:** [2]

**User Summary by Vicuna:** 1. Nystrom Method for Approximating the GMM Kernel.

2. Very sparse random projections.

3. Very sparse stable random projections for dimension reduction in  $l_{\alpha}$  ( $0 < \alpha < 2$ ) norm.

4. A Comparison Study of Nonlinear Kernels.

5. One sketch for all: Theory and Application of Conditional Random Sampling.

6. Collaborative Multi-objective Ranking.

7. Accurate Estimators for Improving Minwise Hashing and b-Bit Minwise Hashing.

8. A new space for comparing graphs.

9. Stability and Risk Bounds of Iterative Hard Thresholding.

10. Hashing Algorithms for Large-Scale Learning.

11. Variational Flow Graphical Model.

**Vicuna Summ. Prediction:** [2]

**User Summary by GPT-3.5:** The researcher's primary research interests revolve around the development of practical algorithms for entropy estimation, compressed counting, and matrix completion. They explore various techniques for improving the efficiency and accuracy of these methods. Additionally, the researcher focuses on advancing hashing algorithms, particularly in the context of maximum inner product search (MIPS) and locality-sensitive hashing (LSH). They also investigate sketch-based sampling techniques, especially for sparse data, and propose novel approaches for estimating entropy of data streams. Furthermore, the researcher has contributions in optimization methods, classification, and dimensionality reduction using stable random projections. Their work

demonstrates a strong emphasis on efficient and scalable solutions for a wide range of machine learning and data analysis tasks.

**GPT-3.5 Summ. Prediction:** [1]