Novel Techniques in Optimizing Query Generation for Retrieval Augmentation based Personalized News Categorization

Tushar Parmanand Budhwani
University of Massachusetts Amherst
Amherst, Massachusetts, USA
tbudhwani@umass.edu

ABSTRACT

In the world of computer systems that understand and use Natural Language Processing (NLP), making the experience more personal for users has become really important. Recently, the big and advanced computer models, called Large Language Models (LLMs), are creating new ways to make virtual assistants and other language systems more tailored to individual users[1]. In this project, I am using the power of LLMs to make responses that are more personalized. This method is designed to be better than current ways of personalizing these systems, especially because it can reduce the loss of important information and better handle situations where we don't have a lot of data about a new user[2].

1. INTRODUCTION

1.1. Introduction and context of the study

Personalization in Large Language Models (LLMs) significantly enhances the efficiency and impact of AI-driven interactions and effectiveness of AI interactions[5]. Personalization improves the efficiency of information delivery. An LLM, equipped with insights into an individual's historical interactions and contextual background, can deliver information that is both relevant and precise[6]. This capability significantly reduces the occurrence of redundant or non-relevant exchanges in user interactions.

1.2. Research questions and objectives of the project

This project aims to enhance how Large Language Models (LLMs) categorize news, focusing on aligning with each user's preferences and interests[3]. The goal is to develop an LLM system that accurately reflects individual news consumption patterns and distinguishes between different news genres. A significant part of the project involves training LLMs with diverse news and user interaction data to improve news sorting for each user. The project also addresses potential biases in AI-driven news feeds by refining the query generation function φ(q), ensuring the news categorization process is effective and unbiased[10].

1.3. Objectives and Milestones of the Task

The effectiveness of personalizing news categorization largely depends on how well the query generation function can pick out key information from user data. By using prompts that match closely with the user's interests and past interactions, the language model can tailor its responses more accurately to each user. The query generation function plays a crucial role in linking user profiles with personalized model inputs. By refining this function, we can significantly improve the flow of the personalization process. My goal is to advance the precision of this function, which will lead to better user data retrieval and, ultimately, a more refined user experience in news categorization.

a. Optimize Query Generation Function φ(q): Develop and refine the φ(q) function to more precisely retrieve relevant user profile items for personalized news categorization. Aim to enhance the effectiveness of queries, focusing on precision in the context of news categorization tasks.
b. Bias Reduction and Fairness Enhancement: Investigate how optimizing φ(q) can minimize bias and promote fairness in news categorization, ensuring that the outputs are unbiased and equitable.
c. Performance Improvement in Personalization: Improve the overall efficiency and user experience in personalized news categorization by ensuring that prompts contain profile data closely related to the user's interests and past interactions.

2. BACKGROUND AND RELATED WORK

The concept of personalizing natural language processing systems is emerging as a pivotal aspect in enhancing user experience and aligning outputs with individual preferences[7][8]. Previous studies in this domain, mainly within the information retrieval and human-computer interaction spheres, have typically focused on search engines and recommender systems (Fowler et al., 2015; Xue et al., 2009; Naumov et al., 2019). However, the integration of personalization within large language models (LLMs), especially for text classification and generation tasks, has been relatively underexplored.

The LaMP Benchmark: The LaMP Benchmark: Addressing this gap, the LaMP benchmark represents a significant stride in this field[1]. Unlike traditional NLP benchmarks which adopt a 'one-size-fits-all' approach, LaMP introduces personalized tasks, creating a more comprehensive evaluation framework[1]. The benchmark encompasses three classification tasks (Personalized Citation Identification, Personalized News Categorization, and Personalized Product Rating) and four text generation tasks (Personalized News Headline, Scholarly Title, Email Subject, and Tweet Paraphrasing). This diversity facilitates a holistic assessment of language models in personalization scenarios.

Retrieval-Augmented Personalization: A notable innovation in LaMP is its retrieval augmentation approach. This method retrieves personalized items from user profiles to create tailored prompts for LLMs, addressing the constraints posed by the large models' input length limitations. The benchmark results demonstrate that LLMs enhanced with profile information outperform those without such personalization.

Zero-Shot and Fine-Tuned Models: The benchmark examines both zero-shot and fine-tuned models, revealing that fine-tuned models often surpass the zero-shot capabilities of larger models. This insight is crucial for developing more efficient personalized language models. The introduction of LaMP opens several research avenues.

3. METHODOLOGY

The experiments are based on the 'LaMP 2: Personalized News Categorization' dataset. Given a news article 'x' in the form of textual data, we want to categorize it based on the given set of categories. In this experiment, I have personalized the model by optimizing the query generation function. I applied the large language model (LLM) across 13 distinct configurations, detailed in Table 1[9].

Enhanced Lexical Augmentation	K - Documents Retrieved	LLM Fine-Tuned	Description
No	0,1,2,3	No	Baseline to Expanded Retrieval
Yes	1,2,3	No	Enhanced Retrieval Range
No	1,2,3	Yes	Fine-Tuned Retrieval Range
Yes	1,2,3	Yes	Optimized Retrieval Range

Table 1: Streamlined Experimental Settings for Query Optimization

4. EXPERIMENTS

4.1. Dataset

LaMP, Language Model Personalization, is an extensive benchmark for datasets designed for the training and evaluation of methodologies aimed at personalizing language models. This benchmark is comprehensive, comprising seven distinct datasets for classification and text generation tasks. Among these, the 'LaMP 2: Personalized News Categorization' dataset was being used in this experiment. It consists of an input prompt for LLM, which is then supplemented by a comprehensive profile section. This profile section consists of multiple articles, each characterized by its text, title, and a designated category.

4.2. Experimental Setup

In this study, the BM25 retrieval algorithm was employed owing to its notable performance in information retrieval tasks. The chosen language model for the experiment is the Flan-T5-base, which operates with a token limit of 512 tokens. Personalization is performed by retrieving top-k articles from the profile section using a retriever function. The retriever is fed with the output of the Query Generation Function ($\phi(q)$) to retrieve top-k documents. After retrieving top-k documents from the retriever, we create the input prompt to the LLM using Prompt Generation Function ($\phi(p)$), which is finally fed to the LLM, Flan-T5-base for my case.

4.3. Query Generation Function ($\phi(q)$)

In the context of enhancing query generation for the personalization of Large Language Models (LLMs), a crucial step involves the preprocessing of textual data. This preprocessing pipeline is designed to refine and adapt the input articles (queries) to optimize the performance of the BM25 retrieval algorithm. The following subsections detail each step of the preprocessing workflow:

Lowercasing - Each character in the article was transformed to its lowercase variant, thereby mitigating any case-sensitive discrepancies during the retrieval process.

Contraction Expansion - Common contractions like "don't" were expanded to "do not", ensuring that the text is in its most explicit and interpretable form. This expansion aided in improving the semantic understanding of the text.

Stop Word Removal - Words such as "the", "is", and "in", which appear frequently and add little informational value, were removed from the text. This led to a more concise and relevant set of terms for the retrieval process.

Non-Semantic Character Filtration: - Characters that do not hold semantic significance (e.g., special symbols, extra spaces) were eliminated.

Lemmatization - Words were converted to their lemmatized form, acknowledging their part-of-speech. For instance, “running” was transformed to “run”.

Enhanced Lexical Augmentation - Synonym Enrichment Technique: - In this advanced step, the preprocessed text undergoes a synonym enrichment process. Each token in the article is paired with a relevant synonym, effectively broadening the scope and variation of query terms. This technique aims to encapsulate a more comprehensive range of relevant documents from the user profile, capturing nuanced and varied lexical representations. It's a strategic expansion to enrich the query's context and reach, ensuring a more robust and encompassing retrieval performance.

4.4. Retrieval Function

I have used the BM25 retriever for all 13 settings to fetch the top 'k' documents from the 'profile' section for each news article, experimenting with various values of k - specifically 1, 2, and 3. Performing Enhanced Lexical Augmentation to the articles resulted in a noticeable improvement in their BM25 scores for some profile articles. I conducted a test on two randomly selected articles, comparing the retrieval scores from the profile before and after including Lexical Augmentation in the input article. (refer Table 2).

4.5. Prompt Generation Function ($\phi(p)$)

The prompt generation method used in this experiment is the same as followed in the LaMP benchmark. The structure of the prompt for the LaMP-2: Personalized News Categorization Task is as follows -

Per Profile Entry Prompt (PPEP) -

the category for the article: "Pi[text]" is "Pi[category]"

Aggregated Input Prompt(AIP) -

concat([PPEP(P1), ..., PPEP(Pn)], ", and "). [INPUT]

where concat is a function that concatenates the strings in its first argument by placing the string in the second argument between them.

PPEP is a function that creates the prompt for each entry in the retrieved profile entries. [INPUT] is the task’s input.

4.6. Language Model

In this research, I used the Flan-T5-base language model in two distinct configurations: one version was fine-tuned, and the other was used as-is, without any fine-tuning. The fine-tuning process involved adjusting the model using a learning rate of 3e-4 and training it across three epochs. This fine-tuning significantly enhanced the model's performance.

Sample Article 1 -

Scores without Enhanced Lexical Augmentation -

```
[array([0.        , 0.        , 4.78241772, 0.        , 2.39120886,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 3.35023668,
        2.65139221, 0.        , 0.        , 0.        , 0.        ,
        1.6318087 , 0.        , 0.        , 1.5891885 , 0.        ,
        0.        , 0.        , 4.57092271, 5.97501645, 7.799573  ,
        2.39145184, 0.        , 0.        ])]
```

Scores with Enhanced Lexical Augmentation -

```
[array([0.        , 0.        , 4.78241772, 0.        , 5.78841147,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 3.35023668,
        2.65139221, 0.        , 0.        , 0.        , 0.        ,
        1.6318087 , 0.        , 0.        , 1.5891885 , 0.        ,
        0.        , 0.        , 4.57092271, 5.97501645, 7.799573  ,
        4.78290368, 0.        , 0.        ])]
```

Sample Article 2 -

Scores without Enhanced Lexical Augmentation -

```
[array([2.39518005, 4.81358388, 2.7630574 , 5.008226  , 2.7630574 ,
        2.05052647, 0.        , 0.        , 2.504113  , 0.        ,
        2.43042248, 0.        , 0.        , 0.        , 2.32767498,
        2.62342662, 0.        , 0.        , 0.        , 0.        ,
        0.        , 2.78409674, 5.59638553, 1.72031541, 1.41091647,
        1.87445871, 1.52678144, 0.        , 1.44336514, 0.        ,
        2.34757186, 1.21907243, 0.        ])]
```

Scores with Enhanced Lexical Augmentation -

```
[array([4.7903601 , 4.81358388, 2.7630574 , 7.512339  , 6.11726411,
        2.05052647, 0.        , 0.        , 2.504113  , 0.        ,
        2.43042248, 0.        , 0.        , 0.        , 2.32767498,
        2.62342662, 0.        , 0.        , 0.        , 0.        ,
        0.        , 2.78409674, 7.23241828, 1.72031541, 1.41091647,
        1.87445871, 1.52678144, 0.        , 1.44336514, 0.        ,
        2.34757186, 1.21907243, 0.        ])]
```

Table 2: Performance improvement in BM25 retrieval scores when Enhanced Lexical Augmentation is used.

The FlanT5-base model was selected for its efficient runtime performance, averaging around 125 milliseconds per sample[2]. This efficiency was crucial, especially considering the approach of maximizing user data inclusion within the 512-token limit for input length. The choice of FlanT5-base was also influenced by its demonstrated superiority in the LaMP benchmark experiments. In these tests, FlanT5-base showed enhanced performance compared to the results obtained in zero-shot experiments using larger models like FlanT5-XXL and ChatGPT. This superior performance of the FlanT5-base model, especially in its fine-tuned form, was a key factor in its selection for the experiment.

4.7. Evaluation

The performance of the models in 13 different settings was evaluated by the script provided in the LaMP benchmark repository. This section presents the findings from tests conducted on the validation dataset. Table x reports the results of fine tuning the language model on the user-based separation setting on the validation set. Table X:

K' - Documents Retrieved	Enhanced Lexical Augmentation	Accuracy	F1 Score
1	No	0.422	0.35
	Yes	0.426	0.361
2	No	0.381	0.317
	Yes	0.375	0.307
3	No	0.352	0.31
	Yes	0.354	0.299

Table 3: Performance Metrics without LLM Fine-Tuning

K' - Documents Retrieved	Enhanced Lexical Augmentation	Accuracy	F1 Score
1	No	0.867	0.768
	Yes	0.869	0.77
2	No	0.871	0.765
	Yes	0.873	0.772
3	**No**	**0.875**	**0.773**
	Yes	0.873	0.769

Table 4: Performance Metrics with LLM Fine-Tuning

5. ANALYSIS AND CONCLUSION

This project stands out for its novel application of Large Language Models (LLMs) in the domain of personalized news categorization. By leveraging LLMs, particularly in refining the query generation function, the project innovates in the way news content is categorized and presented to individual users, aligning with their unique preferences.

5.1. Limitations

The study faces limitations, including potential bias towards certain user profiles, which might lead to less effective personalization for diverse user preferences. The effectiveness of the personalized model depends on the availability and diversity of user data; limited or non-varied data can weaken personalization. Additionally, the resource-intensive nature of fine-tuned LLMs and enhanced lexical augmentation may challenge scalability and practicality, particularly in resource-constrained environments. Moreover, the fine-tuning process risks overfitting, which could reduce the model's adaptability and accuracy for user profiles or news categories not covered in the training data.

5.2. Conclusion

This project enhanced the query generation function φ(q) in Large Language Models (LLMs) for personalized news categorization. The improved φ(q) led to more effective queries, aligning news content closely with user interests and creating a more tailored news experience. Additionally, this refinement reduced bias in AI-driven news sorting, promoting fairness and ethical AI practices. The study highlighted the potential of LLMs in creating precise, relevant, and unbiased digital experiences. However, challenges like limited user profile diversity, dependency on extensive user data, scalability issues, and privacy concerns were identified. These challenges underscore the need for ongoing development of responsible and efficient AI systems. Future efforts will focus on enhancing AI personalization while maintaining fairness, privacy, and user-centricity.

REFERENCES

[1] A. Salemi, S. Mysore, M. Bendersky, and H. Zamani, “LaMP: When Large Language Models Meet Personalization,” arXiv e-prints, 2023. doi:10.48550/arXiv.2304.11406.

[2] Richardson, C., Zhang, Y., Gillespie, K., Kar, S., Singh, A., Raeesy, Z., Khan, O. Z., & Sethy, A. 2023. Integrating Summarization and Retrieval for Enhanced Personalization via Large Language Models. In arXiv preprint arXiv:2310.20081v1.

[3] 1X. Ao et al., "PENS: A dataset and generic framework for personalized news headline generation," in *Proc. 59th Annu. Meet. Assoc. Comput. Linguistics 11th Int. Jt. Conf. Nat. Lang. Process. (Vol. 1: Long Pap.)*, pp. 82–92, 2021.

[4] Go, A., Bhayani, R., & Huang, L. 2009. Twitter sentiment classification using distant supervision.

[5] Asai, A., Schick, T., Lewis, P., Chen, X., Izacard, G., Riedel, S., Hajishirzi, H., & Yih, W. 2022. Task-aware retrieval with instructions. arXiv preprint arXiv:2211.09260 (2022).

[6] Chen, Z. 2023. PALR: Personalization Aware LLMs for Recommendation. arXiv preprint arXiv:2305.07622 (2023).

[7] Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., & Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality. https://lmsys.org/blog/2023-03-30-vicuna/.

[8] Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, E., Wang, X., Dehghani, M., Brahma, S., et al. 2022. Scaling instruction-finetuned language models. arXiv preprint arXiv:2210.11416 (2022).

[9] Li, J., Zhang, W., Wang, T., Xiong, G., Lu, A., & Medioni, G. 2023. GPT4Rec: A generative framework for personalized recommendation and user interests interpretation. arXiv preprint arXiv:2304.03879 (2023).

[10] Rafieian, O., & Yoganarasimhan, H. 2022. AI and Personalization. Available at SSRN 4123356 (2022).