# Can Large Language Models Support Editors Pick Related News Articles?

Bilal Mahmood[1], Mehdi Elahi[1], Samia Touileb[1], and Lubos Steskal[2]

[1] MediaFutures, University of Bergen, Bergen, Norway
Bilal.Mahmood@uib.no
Mehdi.Elahi@uib.no
Samia.Touileb@uib.no
[2] TV 2, Oslo, Norway
Lubos.Steskal@tv2.no

## 1 Abstract

Editors and journalists play an important role on news platforms. Besides creating trustworthy news stories, they also provide valuable expertise on which stories are placed on the front page and hand-pick related articles for platform users to read further. This paper focuses on the specific task of related article selection commonly carried out daily by editors and journalists on news platforms. This is typically a manual process that first utilizes an internal search tool to find a pool of potential candidate articles. Then, from those candidate articles, editors and journalists hand-pick the top related articles for a given news article as a form of expert-selected suggestions for the readers. Although this task can be an important part of the editorial process in news platforms, it may become time-consuming and demanding, often requiring significant human effort.

In addressing this challenge, we propose an automatic mechanism to support editors and journalists in this task by incorporating one of the latest Large Language Models (LLMs), i.e., GPT4o-mini, to shortlist a set of related articles and recommend them to be checked by journalists and editors. Our evaluation of the proposed approach, based on a real-world dataset from one of the largest commercial Norwegian news platforms (i.e., TV 2), demonstrates the effectiveness of the approach in supporting editors and journalists in their task of selecting relevant news articles.

**Keywords:** LLMs · Recommender systems · Editorial tool

## 2 Introduction and Background

In recent years, the swift advancement of Large Language Models (LLMs) has transformed many fields, especially Natural Language Processing (NLP). These sophisticated language models have demonstrated numerous opportunities in handling various NLP tasks and real-world applications, ranging from natural language understanding to generation tasks [10]. However, in other related tasks,

such as Recommender Systems (RSs) and Information Retrieval (IR), the usage of LLMs continues to grow and may require additional attention and exploration [4, 10]. Few recent studies have demonstrated the promising capabilities of ChatGPT in generating recommendations and providing explanations [2, 4].

One of the many potential applications of LLMs is within news platforms, where various tasks performed by editors and journalists could be supported by current advancements. Editors and journalists have the responsibility to ensure that the published content of the news articles is both accurate and relevant. This enables platforms to follow editorial guidelines and adhere to standards in the news domain, which is essential for maintaining credibility and the trust of their readership [3, 9]. Key tasks include reporting accurate news stories and fact-checking. Additionally, one of the subtasks is identifying and selecting related articles for a given news article that is relevant and likely of interest to readers for further exploration. This process demands significant attention and has traditionally been performed manually perhaps due to the lack of supportive tools. In this process, editors and journalists typically use search tools to initially filter a set of candidate articles related to a published article. Afterward, they select the most related ones from this initial collection, relying on their editorial expertise. However, this process can be supported by an automated tool that analyzes articles to find the most related ones and recommends them to editors and journalists for consideration. This can be beneficial in various aspects, mainly saving time.

To investigate this idea, we have formulated the following research question:

– *How can a recommendation mechanism based on Large Language Models (LLMs) be effectively utilized to support editors and journalists in their task of selecting related news articles?*

We have received a real-world dataset from one of Norway's largest editor-managed commercial media houses, TV 2. This dataset comprises the full text of news articles, as well as the related articles selected by editors and journalists. We have integrated one of the latest LLMs, i.e., GPT4o-mini, to generate related article recommendations intended for presentation and use by editors and journalists in their daily editorial workflow and routines. To evaluate our proposed recommendation mechanism, we utilized this dataset to compare the generated recommendations against the editors' selections. We assessed the performance of our approach and measured the quality of recommendations using various evaluation metrics, including Recall@$K$, Precision@$K$, and MAP@$K$. In addition to our proposed method, we considered a simpler baseline, i.e., K-Nearest Neighbors (KNN), based on *Cosine* similarity [7, 5] between news articles, following the approach described in [6]. The results of the evaluation demonstrate the capability of our automatic mechanism to recommend related news articles for editors, thereby supporting them in their editorial workflow.

In summary, the main contributions of this paper are the following:

– We propose a novel recommendation mechanism based on Large Language Models (LLMs) to support editors and journalists in their daily editorial

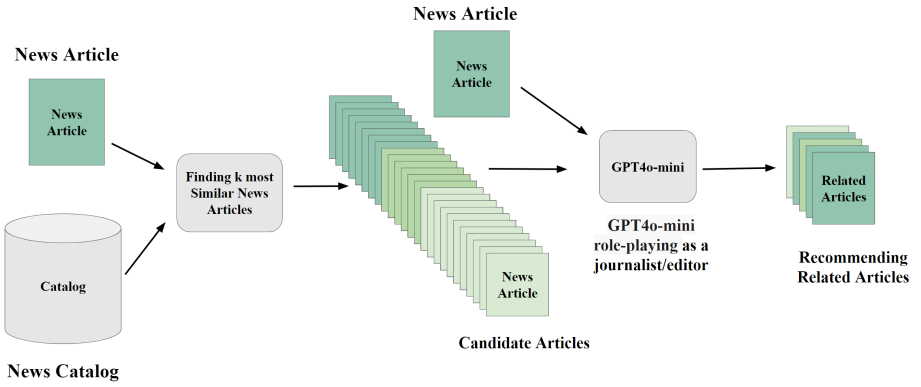



Fig. 1: Overall schematic view of our approach to support journalists and editors in their task of selecting the related articles for a given news article

processes. Our approach takes a news article as input, identifies a set of candidate articles most related to the input article, and recommends them to editors or journalists for consideration;

- We have evaluated our recommendation mechanism using a real-world dataset comprising a unique set of editor-provided feedback on related news articles from one of the largest editor-managed media houses in Norway, i.e., TV 2;
- We have evaluated the quality of our recommendation mechanism across different candidate sizes of news articles presented to editors. Additionally, we compared our approach with a rather traditional news recommendation method, namely K-Nearest Neighbors (KNN), using different metrics such as Precision@$K$, Recall@$K$, and MAP@$K$;
- Our findings indicate that differences in candidate size can impact the quality of our proposed recommendation approach, however, it still outperforms the traditional recommendation method (KNN) in terms of all considered metrics.

The rest of our paper is organized as follows: Section 3 describes the methodology we have devised in this paper. Section 4 discusses the experimental results. Finally, Section 5 provides a conclusion and discusses potential direction for future work.

## 3 Methodology

### 3.1 Dataset

We initially received a dataset of 49757 news articles, each accompanied by at least one related article published by TV 2, one of Norway's largest editor-managed media houses. After pre-processing and filtering to include the most

recent articles, i.e., those published between January 2022 and January 2023, the dataset was reduced to 236 news articles, each with an average of 1.34 related articles. This processed dataset was then utilized to evaluate our recommendation mechanism in the offline settings. This focused scope enabled a more nuanced assessment of the model's effectiveness in a real-world editorial environment using recent news stories.

### 3.2 Recommendation Mechanism

We employed one of the latest multilingual LLMs, i.e., GPT4o-mini [3], in our recommendation mechanism. We adopted a zero-shot setting for generating recommendations based on *role-playing* prompts, as proposed by [2]. Our methodology first involved generating a set of candidate articles, followed by prompting GPT4o-mini with the main article and each candidate article to obtain a relatedness score and explanation. The relatedness score is then utilized to generate the final recommendation list containing the most related articles.

Specifically, we first generated candidate articles by identifying the $K$ most similar articles published within the past year, thereby reducing the pool of potential articles for comparison. To compute similarity, we employed the *Cosine* similarity metric [7, 5]. This approach is inspired by our previous work, where we demonstrated the effectiveness of the K-Nearest Neighbors (KNN) method using Cosine similarity applied to OpenAI embeddings (text-embedding-3-small)[4], which are derived from all textual information in the news articles, to effectively identify related articles from a large set of catalog [6].

We then prompted GPT4o-mini to provide a relatedness score between the main news article and each potential candidate article. Comparing a main article with all articles in the catalog would be expectedly time-consuming and cost-intensive. Therefore, we reduced the number of potential articles by first generating a list of the most similar candidates. We believe this approach makes the API usage of GPT4o-mini more cost-effective and improves the overall efficiency of the proposed recommendation approach.

Finally, the relatedness scores computed by GPT4o-mini are used to rank and select the top five related articles from the candidate set. These recommendations are then evaluated against the ground truth, i.e., the related articles chosen by editors (and journalists). We considered different recommendation metrics for evaluation, i.e., Recall@$K$, Precision@$K$, and MAP@$K$.

Figure 1 describes the overall schematic of our proposed approach. As can be seen, our proposed automatic approach is built on the following steps:

- Step 1: Identify a set of candidate articles for the main news article using the KNN method with Cosine similarity;
- Step 2: Prompt the GPT4o-mini to compute the relatedness score for each candidate article considering the main news article;

[3] https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/
[4] https://openai.com/index/new-embedding-models-and-api-updates/

– Step 3: Use the scores from Step 2 to rank and select the top $K$ most related articles for the main news article.

Following the *role-playing* approach, proposed by [2], we have formulated a prompt that is presented in Figure 2. We explicitly asked the LLM (GPT4o-mini) to return the results in JSON format with a relatedness score given to the potentially related article as well as an explanation for that score.

**System Prompt**

You are a journalist and want to select the related article. I will give you two articles, one main article and the other potential related article and you have to tell me whether that potential article is related or not by providing a score on a scale from 1 to 10. Return the result in a JSON format with related score, and explanations describing the reasoning behind the score as keys. The explanations should be less than 50 words. Start the explanation with the main article and then the potential article.

**Prompt**

Main article: {main_article} Potential article: {potential_article}. Answer:

Fig. 2: System prompt given to the LLM (GPT4o-mini) asking it to give relatedness score to the potential article as well as the explanation as if it were a journalist. In formulating this prompt, we followed a *role-playing* approach, proposed by [2].

### 3.3 Evaluation

We considered different evaluation metrics to measure the quality of recommendation, i.e., Recall@$K$, Precision@$K$, and MAP@$K$ [1, 8] where we have set $K$ = 5. Recall@$K$ is the key metric used to assess the quality of the related article recommendation. For a given main news article $i$, $R_i$@$K$ is defined as:

$$R_i@K = \frac{|L_i \cap \hat{L}_i|}{|L_i|}$$

In this formula, $L_i$ is a set of related articles picked by the editor or journalist for a given main news article $i$, $\hat{L}_i$ represents the recommendation list containing the top $K$ articles in the candidate set with the highest relatedness scores as generated by the GPT4o-mini for the main news article $i$. The overall *Recall@K* ($R@K$) is then measured by averaging the $R_i@K$ values across all the 236 news articles.

Precision@$K$ is another common metric that measures the accuracy in recommending relevant items. To measure Precision@$K$, the top $K$ items are selected for a recommendation for each main news article $i$. Then Precision@$K$ ($P@K$) is calculated as follows:

$$P_i@K = \frac{|L_i \cap \hat{L}_i|}{|\hat{L}_i|}$$

In this formula, $L_i$ denotes the set of related articles selected by the editor and/or journalist for a given main article $i$, and $\hat{L}_i$ represents the recommendation list containing the top $K$ articles in the candidate set with the highest relatedness scores as generated by the GPT4o-mini for the main news article $i$. The overall *Precision@K* ($P@K$) is then obtained by averaging the $P_i@K$ values across all the 236 news articles.

Mean Average Precision@$K$ (MAP@$K$) is the other metric we have considered that measures the quality of output ranking in a recommendation list. MAP@$K$ is calculated by taking into account the arithmetic mean of the Average Precision_i@$K$ ($AP_i@K$) across all the test news articles. Average Precision_i@$K$ ($AP_i@K$) for the top $K$ recommendations ($AP_i@K$) is measured as follows:

$$AP_i@K = \frac{1}{\min(N, K)} \sum_{j=1}^{K} P@j \cdot rel(j)$$

Here, $rel(j)$ is an indicator function that will equal 1 if the $j^{\text{th}}$ recommended item in the recommended list is related and 0 otherwise, as determined by the editor/journalist. $P@(j)$ will equal Precision@$j$. $N$ denotes the total number of related articles for a given main news article $i$ and $K$ is the size of the recommendation list. It is worth noting that since candidate size is an important factor, we have also explored different sizes of the candidate set, varying it from 5 to 50.

Table 1: Results of using an LLM (i.e., GPT-4o-mini) to select the top 5 most related articles to be recommended to the editors and journalists, considering different candidate sizes. KNN stands for K-Nearest-Neighbors based on Cosine similarity, GPT for GPT-4o-mini, and CandSize for Candidate Size.

| **Approach** | **CandSize** | **Recall** | | **Precision** | | **MAP** | |
|---|---|---|---|---|---|---|---|
| | | **@5** | **@CandSize** | **@5** | **@CandSize** | **@5** | **@CandSize** |
| **KNN (baseline)** | _ | 0.436 | 0.436 | 0.107 | **0.107** | 0.366 | 0.366 |
| **KNN+GPT** | 5 | 0.436 | 0.436 | 0.107 | **0.107** | 0.398 | 0.366 |
| **KNN+GPT** | 10 | 0.488 | 0.520 | 0.121 | 0.065 | 0.422 | **0.370** |
| **KNN+GPT** | 20 | 0.554 | 0.606 | **0.136** | 0.039 | **0.456** | 0.369 |
| **KNN+GPT** | 50 | **0.559** | **0.656** | **0.136** | 0.017 | 0.448 | 0.367 |

## 4 Results

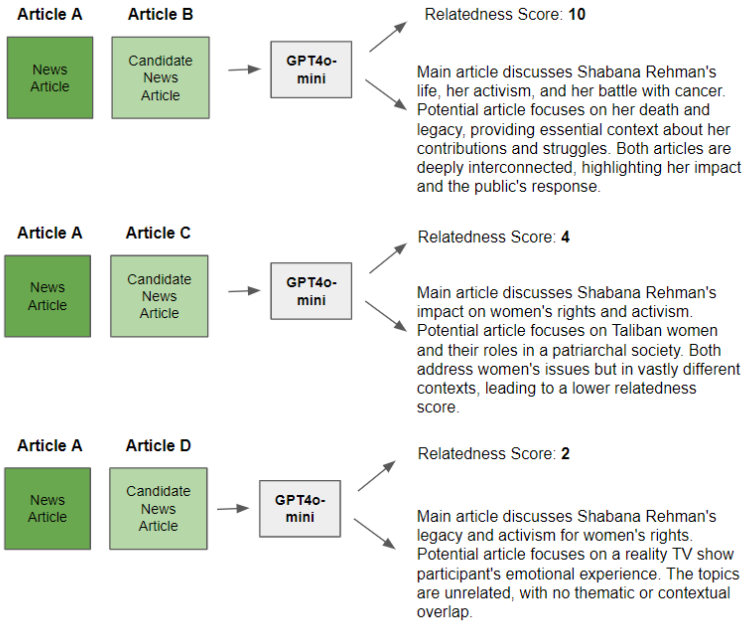
The results of the evaluation are presented in Table 1. We report the recommendation quality when five news articles are selected by the LLM (GPT4o-mini) and suggested to editors and journalists. Additionally, we report the recommendation quality across various candidate set sizes (referred to as CandSize in Table 1). This offers the editors and journalists a flexibility to explore a broader selection of candidate articles and select related articles from an expanded pool.

As shown in the table, the overall results for both cases demonstrate that integrating the LLM (i.e., GPT4o-mini, referred to as GPT in Table 1), into the editorial recommendation process substantially improves the quality of recommendations compared to the traditional recommendation method, based on KNN only (baseline), across all considered metrics. In terms of recall@5, we found that the best results were achieved by applying LLM (referred to as KNN+GPT in Table 1) with a candidate size of 50. The Recall@5 value for this approach is 0.559, meaning that 55.9% of relevant articles were successfully identified and included in the recommendation. Compared to our baseline (KNN only), this is an increase of 28.2% compared to Recall@5 of 0.436 for the traditional approach (KNN only). For Recall@CandSize, the highest value obtained is 0.656, as expected, for the candidate size of 50. Since we only select from the candidate pool, Recall@CandSize represents the upper bound for our Recall@5 metric. For instance, a Recall@50 of 0.656 indicates that only 65.6% of the related articles are available for the LLM to choose from. In other words, this suggests that our LLM-based recommendation approach is capable of identifying nearly 85.2% of the related articles that editors and journalists would ultimately select.

Considering Precision@5, the best results were observed for KNN+GPT with candidate sizes of 20 and 50, both achieving a value of 0.136. This represents a substantial improvement compared to the baseline approach (KNN only), which had a Precision@5 of 0.107. It is important to note that the majority of news articles in the dataset had only one related article, which means the maximum possible Precision@5 score for them was bounded at 0.2. This further highlights the effectiveness of the proposed approach in identifying and recommending a large portion of the related articles to the editors and journalists. For Precision@CandSize, on the other hand, the best-performing approach was the KNN+GPT with the smallest candidate size, i.e., 5. This result is expected, as Precision@$K$ typically tends to decrease with larger recommendation sets. Additionally, the KNN+GPT approach with a candidate size similar to the baseline resulted in both approaches achieving the same Precision@CandSize. However, the difference between these two approaches lies in their ranking output, which can still be influenced by the LLM (GPT4o-mini), potentially impacting the quality of the ranking in terms of MAP@$K$.

In terms of MAP@5, our result has shown that KNN+GPT with a candidate size of 20 had the highest score of 0.456 compared to the baseline approach (KNN only) with the MAP@5 of 0.366. This again showcases that the proposed approach based on the LLM was able to improve the ranking of the related article recommendations.

Fig. 3: An example scenario presenting an article (referred to as A) provided to the LLM, along with the relatedness scores and explanations computed for three candidate articles (referred to as B, C, and D). The relatedness score is then used to rank the candidate news articles and generate the top-5 recommendations.




In summary, the presented results are promising, highlighting the effectiveness of the proposed recommendation mechanisms in supporting editors and journalists with selecting related articles for a given article. We hypothesize that our novel mechanism achieves this level of performance in zero-shot settings due to the significant amount of data used during the model's training phase [10]. This extensive training likely enables the model to effectively infer complex relationships between the main news article and the related articles.

In addition to the previous results, we would like to emphasize the importance of providing explanations for editors and journalists when providing the recommendations. Indeed, one of the key advantages of our approach, based on LLMs (specifically GPT4o-mini), is its ability to generate meaningful explanations for the recommended related news articles, alongside predicting their relatedness scores.

For example, in Figure 3, a scenario is presented where a given article (referred to as A in the Figure) is provided to the LLM. The article discusses the life of a comedian and activist advocating for women's rights, particularly for women of Muslim background in Norway, and her eventual death from cancer. The LLM is then prompted to compute relatedness scores for three candidate

articles (referred to as B, C, and D) based on their relevance to article A. As seen in the example, a high relatedness score of 10 was assigned to a candidate article that detailed her death from cancer, making it a strongly related article and a potential choice for readers to continue following the story. In contrast, candidate article C, which describes the fear faced by participants in a reality TV show, was given a low relatedness score of 2, as it is not connected to the life of the activist in article A. Additionally, another candidate article (B) describes the challenges women face under the Taliban government. The LLM assigned it a relatedness score of 4, as it discusses women's rights but in a different context, resulting in a score higher than article C. Again, while this is just an example, it may illustrate a potential scenario in real-world applications.

## 5 Conclusion and Future Work

One of the editorial tasks carried out daily by editors and journalists on news platforms is reviewing recently published articles and finding the most related articles for users to explore further. This task is performed manually and, despite its importance, can be time-consuming and require substantial expert effort.

In this paper, we address this challenge by proposing a recommendation mechanism that integrates one of the latest Large Language Models (LLMs), i.e., GPT4o-mini, into the process. Given a particular news article, the LLM is prompted to compute a relatedness score and provide an explanation justifying the score for editors and journalists. Accordingly, the top related articles are recommended to them for consideration. We received a real-world dataset from one of the largest media houses in Norway, i.e., TV 2. This dataset contains a unique form of feedback data indicating which articles have been selected as related by editors. We incorporated this data and evaluated our approach, comparing it with a traditional K-Nearest Neighbors (KNN) recommendation method. The results were promising and demonstrated the superiority of our proposed approach across various evaluation metrics.

In future work, we plan to experiment with different prompts, test several LLMs of various sizes, and ultimately build a practical tool for journalists and editors at TV 2 to incorporate into their workflow and enhance their productivity.

## 6 Acknowledgement

The Research Council of Norway partly supported this work with funding to MediaFutures: Research Centre for Responsible Media Technology and Innovation, through the Centre for Research-based Innovation scheme, project number 309339.

## References

1. Paolo Cremonesi, Franca Garzotto, Sara Negro, Alessandro Vittorio Papadopoulos, and Roberto Turrin. Looking for "good" recommendations: A comparative evaluation of recommender systems. In *Human-Computer Interaction–INTERACT 2011:*

*13th IFIP TC 13 International Conference, Lisbon, Portugal, September 5-9, 2011, Proceedings, Part III 13*, pages 152–168. Springer, 2011.

2. Dario Di Palma, Giovanni Maria Biancofiore, Vito Walter Anelli, Fedelucio Narducci, Tommaso Di Noia, and Eugenio Di Sciascio. Evaluating chatgpt as a recommender system: A rigorous approach. *arXiv preprint arXiv:2309.03613*, 2023.
3. Mehdi Elahi, Dietmar Jannach, Lars Skjærven, Erik Knudsen, Helle Sjøvaag, Kristian Tolonen, Øyvind Holmstad, Igor Pipkin, Eivind Throndsen, Agnes Stenbom, et al. Towards responsible media recommendation. *AI and Ethics*, pages 1–12, 2022.
4. Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. Is chatgpt a good recommender? a preliminary study. *arXiv preprint arXiv:2304.10149*, 2023.
5. Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. *Recommender systems handbook*, pages 73–105, 2011.
6. Bilal Mahmood, Mehdi Elahi, Samia Touileb, Lubos Steskal, and Christoph Trattner. Incorporating editorial feedback in the evaluation of news recommender systems. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization*, pages 148–153, 2024.
7. Michael J Pazzani and Daniel Billsus. Content-based recommendation systems. In *The adaptive web: methods and strategies of web personalization*, pages 325–341. Springer, 2007.
8. Markus Schedl, Hamed Zamani, Ching-Wei Chen, Yashar Deldjoo, and Mehdi Elahi. Current challenges and visions in music recommender systems research. *International Journal of Multimedia Information Retrieval*, 7:95–116, 2018.
9. Christoph Trattner, Dietmar Jannach, Enrico Motta, Irene Costera Meijer, Nicholas Diakopoulos, Mehdi Elahi, Andreas L Opdahl, Bjørnar Tessem, Njål Borch, Morten Fjeld, et al. Responsible media technology and ai: challenges and research directions. *AI and Ethics*, 2(4):585–594, 2022.
10. Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *ACM Transactions on Knowledge Discovery from Data*, 18(6):1–32, 2024.