LANCE: Exploration and Reflection for LLM-based Textual Attacks on News Recommender Systems

Yuyue Zhao
University of Science and Technology of China
Hefei, China
University of Amsterdam
Amsterdam, Netherlands
yuyuezha00@gmail.com

Jin Huang[†‡]
University of Cambridge
Cambridge, United Kingdom
jh2642@cam.ac.uk

Shuchang Liu
Rutgers University
New Jersey, USA
shuchang.syt.liu@rutgers.edu

Jiancan Wu
University of Science and Technology of China
Hefei, China
wujcan@gmail.com

Xiang Wang[†]
University of Science and Technology of China
Hefei, China
xiangwang@u.nus.edu

Maarten de Rijke
University of Amsterdam
Amsterdam, Netherlands
M.deRijke@uva.nl

Abstract

News recommender systems rely on rich textual information from news articles to generate user-specific recommendations. This reliance may expose these systems to potential vulnerabilities through textual attacks. To explore this vulnerability, we propose LANCE, a **LA**rge language model-based **N**ews **C**ontent r**E**writing framework, designed to influence news rankings and highlight the unintended promotion of manipulated news. LANCE consists of two key components: an *explorer* and a *reflector*. The *explorer* first generates rewritten news using diverse prompts, incorporating different writing styles, sentiments, and personas. We then collect these rewrites, evaluate their ranking impact within news recommender systems, and apply a filtering mechanism to retain effective rewrites. Next, the *reflector* fine-tunes an open-source LLM using the successful rewrites, enhancing its ability to generate more effective textual attacks. Experimental results demonstrate the effectiveness of LANCE in manipulating rankings within news recommender systems. Unlike attacks in other recomendation domains, negative and neutral rewrites consistently outperform positive ones, revealing a unique vulnerability specific to news recommendation. Once trained, LANCE successfully attacks unseen news recommender systems (*i.e.,* those for which LANCE received no information during training), highlighting its generalization ability and exposing shared vulnerabilities across different systems. Our work underscores the urgent need for research on textual attacks and paves the way for future studies on defense strategies.

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

Large language model, Textual attack, News recommender system

ACM Reference Format:
Yuyue Zhao, Jin Huang, Shuchang Liu, Jiancan Wu, Xiang Wang, and Maarten de Rijke. 2025. LANCE: Exploration and Reflection for LLM-based Textual Attacks on News Recommender Systems. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems (RecSys '25), September 22–26, 2025, Prague, Czech Republic.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3705328.3748081

1 Introduction

News recommender systems (RSs) rely on rich textual content of news articles and play a unique role in supporting users' participation in a democratic society by recommending news articles. This makes them different from other RS scenarios, *e.g.,* those for products, movies, and books [26, 29]. Neural news RSs often use language models (LMs) to enhance their understanding of the content of news articles and improve recommendation accuracy [44]. These news RSs have been shown to be vulnerable to malicious attacks [11, 21]. The goal of attacks is to produce recommendations as the attacker desires, *e.g.,* an attacker-chosen target news article is recommended to many users. This could lead to severe threats to the trustworthiness of news RSs and significant social consequences, *e.g.,* manipulating users' opinions and spreading misleading information.

Textual attacks. A widely studied type of attacks in general recommendation scenarios is namely data poison attacks [11], a.k.a. shilling attacks, which commonly inject fake users [27] or fake interactions [4] into the RS to increase the exposure of a target item set. Such attacks are relatively easy to defend against, *e.g.,* by fake user detection [1] and adversarial training [32]. In contrast, considering the unique nature of news RSs, which relies on textual information, attack strategies w.r.t. news content perturbation have gradually gained attention [21, 42]. In this paper, we consider a specific, widely studied scenario, where an adversarial content provider wants to boost the ranking of a specific article for all users

[†] Jin Huang and Xiang Wang are corresponding authors.
[‡] Work done while the author was with the University of Amsterdam.

RecSys '25, Prague, Czech Republic

ACM ISBN 979-8-4007-1364-4/25/09
https://doi.org/10.1145/3705328.3748081

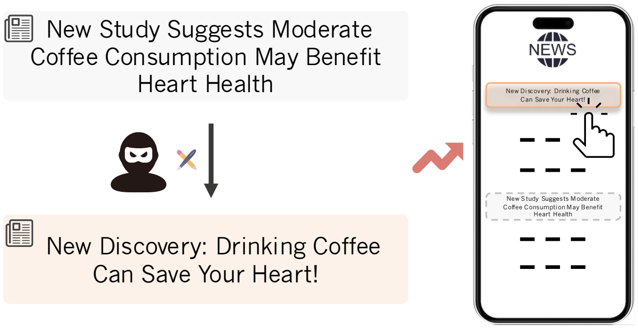

New Study Suggests Moderate Coffee Consumption May Benefit Heart Health
New Discovery: Drinking Coffee Can Save Your Heart!
NEWS
New Discovery: Drinking Coffee Can Save Your Heart!
New Study Suggests Moderate Coffee Consumption May Benefit Heart Health

Figure 1: Illustrating how textual attack works.

by textual content perturbation techniques (see Figure 1). Accordingly, following Oh et al. [21], we define the *textual attack* in this scenario as an act of the news provider rewriting the news article to increase its exposure to users while keeping the content similar to the original.

Current limitations. Given the powerful capabilities of large language models (LLMs) in understanding and generating text, it is both straightforward and interesting to explore their use in textual attacks. Zhang et al. [39] mentioned a method to prompt LLMs to rewrite news to make it more "attractive." Due to the inherent difference in objectives between LLMs and textual attacks, as well as the lack of guidance (*e.g.,* indicators of whether the rewritten news will boost its ranking in a specific RS), this approach is not effective in textual attacks (see Section 4.2). Wang et al. [30] alter the textual information of target items by simulating the characteristics of popular items. However, this approach is not effective when applied to news RSs, as popular news topics change very frequently. To address these limitations and present an effective attack approach designed for news RSs, Oh et al. [21] propose ATR-2FT, which fine-tunes a small-sized LM [*e.g.,* OPT-350M, 40], following a joint learning objective that simultaneously optimizes item ranking and content quality. Although ATR-2FT is implemented using a small-size LM, it can ideally be extended to use LLMs. Still, we identify three limitations of ATR-2FT, even when extended to LLMs:

(1) ATR-2FT requires prior knowledge of the RS to be attacked.[1]
(2) Due to training efficiency constraints, ATR randomly select a subset of users and items to optimize the item ranking promotion objective, which reduces its effectiveness.
(3) ATR optimizes rewritten content quality by updating the text embeddings from the fine-tuned OPT rather than the language space, which reduces naturalness and makes the attack easier to detect.

Proposed method. To address these limitations and explore the LLMs' ability in textual attacks on news RSs, we propose LANCE, a two-stage **LA**rge language model-based **N**ews **C**ontent r**E**writing framework. In the first stage, we propose an *explorer* component to effectively generate the diverse rewritten content, which can be used to guide the optimization of our LLM-based attack method. We use a powerful closed-source LLM (*e.g.,* GPT-4o) capable of accessing online information to generate rewritten versions of a given news article and identify which rewrites successfully boost its ranking on the news RSs and which do not. To explore potential textual factors that might influence ranking in the news scenario [2, 28, 37], we employ diverse prompts — covering five writing styles, three sentiments, and six personas — to produce varied rewrites, and implement a filtering mechanism, which provides binary feedback on whether the rewrite successfully improves the rank to the top K or not, to ensure quality.

In the second stage, we fine-tune a *reflector* module to learn effective textual attacks based on the explored rewrites. For each original article, we form a triplet consisting of the original text, a successful rewrite, and a failed rewrite. During fine-tuning, the original article serves as the instruction. We adopt a DPO [25] training procedure that prioritizes successful rewrites over failed ones, ensuring the model learns how to generate more effective attacks.

LANCE operates entirely at the textual level. We explicitly instruct the LLMs to rewrite the news while preserving its original meaning, aiming to avoid semantic mismatches and maintain consistency with the original article. By combining systematic exploration, targeted reflection, our approach addresses current limitations and effectively achieves ranking manipulation.

Main findings. Extensive experiments show that LANCE achieves state-of-the-art attack performance on three news RSs. By using a small fraction (4.59%) of input news articles and their corresponding rankings, the fine-tuned *reflector* can rewrite news content and effectively boost its rank during inference, causing the RS to promote it above the original version. Our analysis of diverse prompts in the *explorer* indicates that in contrast to attacks on other types of RSs [39], which often insert positive words to raise item ranking, negative and neutral rewrites tend to outperform positive ones in the news domain, revealing a unique attack preference in news RSs. When trained on rewritten data from a single news RS, LANCE can successfully generate rewrites that enhance the target news rankings on unseen news RSs (*i.e.,* systems for which LANCE received no information during training), demonstrating its generalization capabilities.

Contributions. To summarize our contributions:

(1) We introduce LANCE and, using it, show that textual attacks can pose a significant vulnerability for news RSs, requiring low attack costs and limited system information knowledge.
(2) We highlight the unique effectiveness of negative rewrites in news RSs, showing how they differ from attacks on other types of RSs in a news context.
(3) We demonstrate the generalization capability of LANCE by showing that a model trained on one news RS can generate successful attacks on unseen news RSs, showing shared vulnerabilities across news RSs.
(4) We propose an intuitive defense strategy by measuring token probabilities in news text. Although it cannot fully detect rewritten text, it highlights the need and potential for developing more robust defenses in the future.

2 Related Work

2.1 Attack on Recommender Systems

Posioning attacks on RSs have been shown to be effective in manipulating RSs predictions [31]. Most existing work on poisoning attacks involves injecting fake user interactions into training or test

[1] Even in the black-box setting, ATR-2FT assumes prior knowledge of the type of RS method (*e.g.,* sequential or collaborative filtering-based).

data to control the recommended items. E.g., RAPU-R [38] identifies incomplete and perturbed data, and then crafts fake user-item interactions to influence the recommendations. With the rise of content-based RSs, textual attacks have emerged. These attacks focus on manipulating the textual content associated with items, without requiring fake user interactions. E.g., ARG [5] introduces a reinforcement learning framework to generate fake reviews that target review-based RSs.

From a seller's perspective, promoting items by inserting fake reviews still requires the creation of fake user accounts to post the reviews. To address this limitation, ATR [21] is a two-phase fine-tuning method to rewrite item descriptions, enabling sellers to unfairly boost their product rankings without needing fake user accounts. Similarly, TextRecAttack [39] targets LLM-based RSs and uses adversarial textual attacks in NLP tasks [8, 12] by perturbing and searching the item text to increase item exposure. It iteratively modifies each text until a stopping criterion is met, requiring repeated system feedback for every item.

These methods often rely on knowledge of the victim RS, such as its parameters or embeddings, or require input-output pairs to train a surrogate victim RS. Inspired by the correlation between popular items and their ranking positions, TextSimu [30] exploits LLMs to simulate the textual characteristics of popular items. Because news RSs are constantly evolving, the attributes that made older news articles popular may no longer be relevant for rewriting new content or enhancing its ranking. Hence, the applicability of existing textual attack models to news RSs is limited.

2.2 News Recommender Systems

News RSs provide personalized recommendations by encoding news articles using LMs [15]. Early systems like LSTUR [3] use GloVe embeddings [24] to represent news content and employed GRU networks to learn user representations from their browsing history. NRMS [34] and NAML [33] also use GloVe, with NRMS using multi-head attention and NAML adopting multi-view learning for unified news representations. With the success of pre-trained LMs, models like BERT [6] and RoBERTa [19] have been employed in news RSs. E.g., MINER [14] uses BERT for news encoding and introduces poly-attention for user representation. PLM-NR [35] explores multiple pre-trained models to improve news representation. Recently, LLMs have been adopted for encoding news content. ONCE [18] uses both closed and open-source LLMs for news encoding, and Zhao et al. [44] show that LLMs excel in cold-start user scenarios in news RSs. Unlike other RSs domains, such as e-commerce or movies, most news articles that show up during inference do not appear during training [44]. This makes it difficult for news RSs to leverage item ID information during inference, forcing them to rely on content to capture relationships between news and users. This reliance exposes news RSs to vulnerabilities from textual attacks based on news content.

3 Methodology

This section presents our methodology for conducting textual attacks on news RSs. We first define the problem, outlining the core components of news RSs and the attack scenario where a content provider rewrites news articles to improve their rankings. We then introduce the LANCE framework, which has two stages (as shown in Figure 2): an *Exploration* module, which generates diverse rewritten variants of news content by perturbing textual factors such as writing styles, sentiment polarity, and author personas, followed by a filtering process to select effective rewrites; a *Reflection* module, which fine-tunes a LLM on the filtered data to learn rewriting strategies. Finally, we describe how the fine-tuned *reflector* generates the rewritten content at inference time.

3.1 Problem Definition

News recommender systems. In news RSs, let $\mathcal{V}$ denote the set of news items and $\mathcal{U}$ denote the set of users. Each news item $v \in \mathcal{V}$ has its textual content t_v, which is encoded by the system's news encoder into a news representation $\boldsymbol{q}_v$. Each user $u \in \mathcal{U}$ has a click history $H_u = \{v_1, v_2, \ldots, v_n\}$. The user encoder processes H_u to produce the user representation $\boldsymbol{p}_u$. The goal of news RSs is to learn a scoring/rank function as follows:

$$\mathcal{R}(t_v, u; f^{RS}) := f^{RS}(t_v, H_u) \tag{1}$$

where $f^{RS}(\cdot)$ is the pre-trained and frozen news RS model, and the $\mathcal{R}(\cdot)$ denotes the rank function.

Attack scenario. A textual attack on news RSs is conducted by a news content provider seeking to promote a target news $t_g \in \mathcal{V}_g$. The attacker rewrites the original content t_g as $t_g{}'$, with the aim of achieving a higher rank (*i.e.,* a lower value in the rank function):

$$\mathcal{R}(t_g{}', u; f^{RS}) < \mathcal{R}(t_g, u; f^{RS}). \tag{2}$$

The task is to find a rewriting transformation $t_g \longrightarrow t_g{}'$ that improves the news's ranking in $f^{RS}(\cdot)$.

3.2 Explorer: Rewriting and Filtering

To explore how rewriting news articles can promote their rankings, we draw inspiration from LLM-based data augmentation [7, 43] and propose a rewriting process. In particular, we consider textual factors that might influence ranking in the news domain [2, 28, 37]. Our *explorer* prompts an LLM to generate rewritten versions of news content across three dimensions: writing styles (*e.g.,* formal to colloquial), sentiment polarity (*e.g.,* positive to negative), and author personas (*e.g.,* objective to opinionated). Detailed descriptions of these dimensions are provided in our repository.[2] Formally, for a news content $t_e \in \mathcal{V}_e$, the *explorer* gets a diverse set of rewritten variants:

$$\mathcal{S}_e = \mathcal{T}_{style}(t_e) \cup \mathcal{T}_{sentiment}(t_e) \cup \mathcal{T}_{persona}(t_e), \tag{3}$$

where $\mathcal{T}_{style}(\cdot)$, $\mathcal{T}_{sentiment}(\cdot)$ and $\mathcal{T}_{persona}(\cdot)$ denote rewriting operations that perturb the original text t_e by altering its writing style, sentiment, and persona, respectively. $\mathcal{S}_e$ collects all rewritten versions of $t_e \in \mathcal{V}_e$. Here, $\mathcal{V}_e$ is sampled from the training-stage news, while the target news $t_g \in \mathcal{V}_g$ is sampled from the inference-stage news to evaluate the attack effectiveness against the deployed news RS. Below is an example of the rewriting prompt used by the *Explorer* to apply different writing styles:

> **Task**: *Rewrite the provided news title and abstract into 5 different versions, each reflecting a specific writing*

[2] https://github.com/Go0day/LANCE.

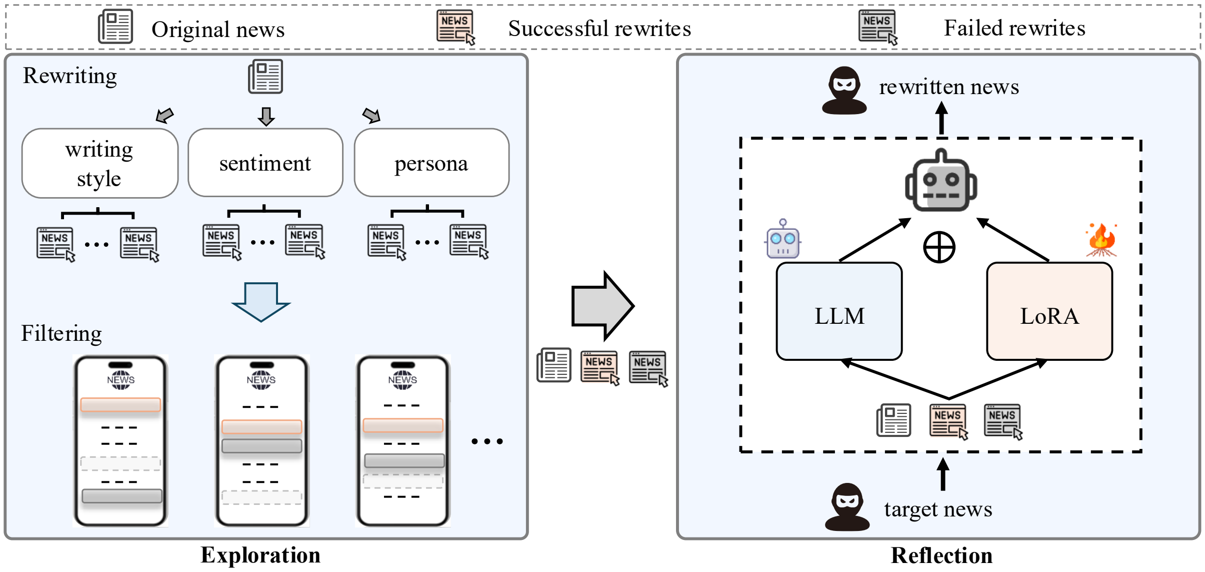


Figure 2: An overview of the proposed LANCE framework for textual attacks on news RSs.

> *style. The output should maintain the original core information while adhering to the designated tone and length constraints.*
> **Writing Styles**: *Narrative, Persuasive, Journalistic, Humorous and Conversational.* <$\mathcal{D}$>
> **Original News**: <t_e>

where <$\mathcal{D}$> denotes the detailed description of different writing styles [37], and <t_e> represents the original news article to be rewritten.

While the rewritten data $\mathcal{S}_e$ can be used to fine-tune the *reflector*, we apply additional binary filtering mechanism to ensure the quality of rewrites. Consider two news articles t_e^A and t_e^B, each with rewritten versions s^A and s^B. *Case 1*: t_e^A improves from rank 100 to 60 with s^A. *Case 2*: t_e^B improves from rank 50 to 10 with s^B. Though both rewrites achieve a 40-rank improvement, s^B is more impactful: users typically browse only the top-ranked news (*e.g.*, top 50), so s^B gains visibility while s^A remains largely unnoticed. Therefore, we filter the rewritten news $s_e \in \mathcal{S}_e$ and construct the successful rewrites S_e^+ as follows:

$$\begin{aligned} S_e^+ &= S_e^+ \cup \{s_e\} \\ &\text{if} \begin{cases} \mathcal{R}(s_e, u; f^{RS}) < K, \\ \mathcal{R}(t_e, u; f^{RS}) > K. \end{cases} \end{aligned} \tag{4}$$

where K represents the top-K ranking threshold and serves as a hyperparameter. For unsuccessful rewrites S_e^-, we identify explicit cases where the rewritten news fails to improve ranking:

$$\begin{aligned} S_e^- &= S_e^- \cup \{s_e\} \\ &\text{if } \mathcal{R}(s_e, u; f^{RS}) < \mathcal{R}(t_e, u; f^{RS}). \end{aligned} \tag{5}$$

Finally, the selected rewritten news samples (t_e, S_e^+, S_e^-) are collected for training the *reflector*.

3.3 Reflector: Fine-Tuning for Textual Attack

Based on the explored rewritten news, we now illustrate how the *reflector* learns to rewrite news content to improve its ranking. Recent studies have shown that human-labeled, pairwise data can serve as reward signals to align LMs with human preferences, such as RLHF [22] and DPO [25]. RLHF uses a preference model to model preference distributions, whereas DPO directly learns the optimal policy from pairwise preference data and is often more practical for preference alignment [22, 25]. Therefore, we employ DPO to train the *reflector* on the rewritten data, enabling it to perform textual attacks on news RSs more effectively. Below is the fine-tuning template used for preference alignment:

> **Task**: *You are an expert in news content optimization for recommender systems. Your goal is to rewrite the given news title and abstract to maximize their chances of ranking higher in a typical news recommender system. The output should maintain the original core information while adhering to the goals and length constraints.* <t_e>
> *Please rewrite the title and abstract according to two optimization goals: (1) Focused on maximizing user clicks. (2) Focused on improving algorithmic ranking based on relevance and engagement.*
> **Chosen rewrite**: $s_e^+ \in S_e^+$
> **Rejected rewrite**: $s_e^- \in S_e^-$

where <t_e> is the original news content, s_e^+ and s_e^- denote rewrites that respectively succeed or fail to promote the t_e's ranking. We implement the *reflector* using Llama 3.1-8B, parameterized by θ, and fine-tune it with a DPO loss to maximize the probability of the chosen rewrite and minimize the probability of the rejected one:

$$\begin{aligned} L_{\text{DPO}}(\theta) = \; & -\mathbb{E}_{(t_e, s_e^+, s_e^-)} \left[\log \sigma \left(\beta \log \frac{p_\theta(s_e^+|t_e)}{p_{\text{ref}}(s_e^+|t_e)} \right. \right. \\ & \left. \left. -\beta \log \frac{p_\theta(s_e^-|t_e)}{p_{\text{ref}}(s_e^-|t_e)} \right) \right], \end{aligned} \tag{6}$$

Table 1: Statistics of the MIND dataset.

#users	#news	$\lvert\mathcal{V}_e\rvert$	$\lvert\mathcal{V}_g\rvert$	$\frac{\lvert\mathcal{V}_e\rvert}{\#\text{news}}$
94,057	65,238	3,000	300	4.59%

where $\sigma(\cdot)$ is the sigmoid function, β is a temperature parameter, $p_\theta(s_e|t_e)$ represents the probability of the *reflector* generating s_e given t_e, and $p_{\text{ref}}(s_e|t_e)$ denotes the probability of generating s_e given t_e under the reference model, which is the pre-trained LLM before preference fine-tuning.

Final attack generation. Finally, we describe the attack generation process. Given a target news t_g, the fine-tuned *reflector* transforms and rewrites it into a new version: $t_g \xrightarrow{p_\theta(\cdot)} t_g{}'$. The rewritten $t_g{}'$ is then submitted to the victim news RSs to promote its ranking, thereby achieving the attack goal.

4 Experiments

In this section, we conduct extensive experiments to answer the following research questions:[3]

(RQ1) How does the proposed LANCE approach perform compared to existing models in attacking news RSs?

(RQ2) How do variations in diverse news content (*i.e.,* writing style, sentiment, and persona) affect the performance of rewrite attacks on news RSs?

(RQ3) How does fine-tuning a single LLM perform in attacking different news RSs?

(RQ4) How does the LANCE attack affect wider dimensions (*i.e., recommendation performance*, *semantic preservation*, and *detectability*) of news RSs ?

4.1 Experimental Setup

4.1.1 Dataset. **Benchmark datasets.** We conduct experiments on the MIND dataset [36], a publicly available news recommendation dataset from Microsoft News. The dataset's statistics are detailed in Table 1. We employ impressions from November 9–14, 2019 for training the RSs and impressions from November 15, 2019 for testing [16, 44].

Target news. Previous textual attack methods on recommender systems [21, 39] typically select a random set of target items and perform word perturbation search or rewrite learning directly on them. In the news domain, where information changes rapidly, such approaches lack robustness and may fail to effectively rewrite and promote news articles. To address this limitation, we derive attack training news t_e from news that appears in the MIND training impressions and derive target news t_g from news in the MIND test impressions. To ensure that the textual attack generalizes across news with varying popularity levels [20, 44], we randomly select 1,000 training news items and 100 test news items from three popularity levels based on frequency: (0–20%), (40–60%), and (80–100%). These selections form $\mathcal{V}_e$ and $\mathcal{V}_g$, with final sizes of $|\mathcal{V}_e| = 3,000$ and $|\mathcal{V}_g| = 300$, respectively.

4.1.2 News Recommender Systems. We select three mainstream news RS models as victim models: NAML [33], NRMS [34], and LSTUR [3]. Following [35, 44], we re-implement these models using the BERT-base version as the news encoder, with its parameters fine-tuned on recommendation signals. Their architectures are summarized below:

- **NAML** [33]: NAML models news representations using both titles and abstracts. It applies a multi-view learning mechanism to integrate titles, bodies, categories, and subcategories. User representations are learned through an attention mechanism based on browsing history.
- **NRMS** [34]: NRMS models news representations using only titles. It employs a multi-head self-attention mechanism to capture semantic features. User representations are learned through self-attention on browsing history.
- **LSTUR** [3]: LSTUR models news representations using titles and topic categories. It employs a GRU network to learn user representations from browsing history. The model captures both long-term and short-term user interests.

4.1.3 Baseline Rewriting Methods. Textual attacks rewrite news content to improve its ranking. We compare LANCE with eight LLM-based baselines, including the state-of-the-art ATR [21]; we exclude shilling attacks from the baselines because they require generating fake users and ratings, which falls outside the scope of our work. Similarly, we exclude adversarial textual attack methods from NLP tasks, such as TextRecAttack [39], because they rely on repeated feedback from a RS for each item. This approach is impractical for our target news set, $\mathcal{V}_g$, where no information is available at this stage. The baselines are as follows:

- **GPT-4o**, **GPT-3.5** and **Llama-3.1** (without context). We follow the implementation described in existing work [39]. In this baseline group, we prompt these LLMs to rewrite news items without incorporating any news RSs data as contextual input.
- **GPT-4o**, **GPT-3.5** and **Llama-3.1** (with context). We follow the implementation from prior work [30]. In this baseline group, we use popular news articles from the same category as the target news as context. The LLMs are then prompted to rewrite the news, leveraging contextual information to improve ranking performance.
- **Llama-FT$_{\text{rec}}$**. We fine-tune Llama-3.1-8B on news recommendation data. This fine-tuning injects domain-specific knowledge into the LM. The fine-tuned model then performs the rewriting task without additional contextual input to enhance the target news ranking.
- **ATR-2FT** [21]. ATR-2FT is a recent textual attack method that uses a fine-tuned LM (*i.e.,* OPT-350M [40]). To adapt ATR-2FT to our experimental setting, we isolate the news text encoder from the news RS and perform embedding alignment between the LM embeddings and the news text encoder. The attack process is optimized using a promotion loss and a text generation loss. To ensure a fair comparison, we train ATR-2FT on our sampled news dataset $\mathcal{V}_e$ and test it on the target news dataset $\mathcal{V}_g$.

4.1.4 Implementation Details. We use GPT-4o to explore rewrites, as it can access external knowledge and perform well in the news domain. We use Llama-3.1-8B in LANCE for *Reflector*. This open-source model has strong reasoning capabilities for textual attacks. All attack models and news RSs are implemented in PyTorch. For DPO fine-tuning, we use Lora (Low-Rank Adaptation) [10] to efficiently fine-tune the model. We set $\beta = 0.1$, and the learning rate is selected from {1e-5, 5e-5, 1e-4, 5e-4}. The number of fine-tuning

[3] Our codes are available at https://github.com/Go0day/LANCE.

Table 2: Bold values indicate the performance of LANCE in each column, while underlined values represent the best performance among the baselines. Results are averaged over five runs. The raw "%Impv" column shows the relative improvement of LANCE over the best baseline; For the Rank metric, "%Impv" denotes the average rank improvement compared to the original target news ranking.

Model		NAML				NRMS				LSTUR			
	Context?	BSR↑	Rank↓	Expo↑	Appear↑	BSR↑	Rank↓	Expo↑	Appear↑	BSR↑	Rank↓	Expo↑	Appear↑
Original	-	-	21,001	0.29	0.0058	-	20,970	0.11	0.0022	-	20,811	0.04	0.0007
GPT-4o	Y	0.44	22,097	0.46	0.0093	0.43	22,557	0.08	0.0016	0.46	21,808	0.03	0.0006
	N	0.43	22,193	0.43	0.0086	0.41	22,901	0.09	0.0019	0.43	22,241	0.03	0.0007
GPT-3.5	Y	0.42	22,317	0.48	0.0095	0.43	22,429	0.09	0.0018	0.47	21,317	0.06	0.0012
	N	0.42	22,304	0.48	0.0097	0.41	22,692	0.09	0.0018	0.43	21,983	0.03	0.0007
Llama-3.1	Y	0.45	21,800	0.43	0.0087	0.45	22,107	0.09	0.0018	0.46	21,562	0.04	0.0008
	N	0.45	21,828	0.50	0.0101	0.44	22,153	0.10	0.0020	0.45	21,674	0.05	0.0010
Llama-FT$_{rec}$	-	0.45	21,742	0.51	0.0103	0.46	21,972	0.07	0.0014	0.45	21,806	0.05	0.0009
ATR-2FT	-	0.53	20,604	0.52	0.0104	0.47	21,182	0.17	0.0034	0.49	21,013	0.06	0.0015
w/ Llama	-	0.56	19,846	0.63	0.0126	0.52	20,737	0.22	0.0046	0.60	18,732	0.08	0.0018
LANCE	-	**0.69**	**18,125**	**0.79**	**0.0159**	**0.56**	**18,677**	**0.41**	**0.0084**	**0.57**	**19,375**	**0.08**	**0.0016**
% Impv	-	30.2%	2,479	51.9%	52.9%	19.2%	2,293	141.2%	147.0%	16.3%	1,436	33.3%	6.7%

epochs is fixed at 3. For news RSs, we use the same hyperparameters as in [44]. All training is performed on three NVIDIA RTX A6000 GPUs, each with 49,140M of memory.

4.1.5 Evaluation Metrics. **Attack performance.** We evaluate attacks using four metrics: (i) Boost Success Rate (BSR): The percentage of target news articles successfully boosted above the average rank across all users. (ii) Average Predicted Rank (Rank): The mean rank of target news articles across all users, with and without adversarial rewrites. (iii) Exposure@K (Expo): The proportion of users who see the target news articles in their top-K recommendations. (iv) Appear@K (Appear): The frequency of target news articles appearing in the top-K recommendations across all users. We set $K = 50$.

Naturalness performance. We evaluate the naturalness of rewritten news articles using the following metrics: (i) Language model perplexity (PPL), which measures how well a language model predicts the rewritten text. (ii) BLEU [23], which evaluates the overlap of n-grams to assess the quality of the rewritten content compared to the original. (iii) ROUGE-L [17], which measures recall-oriented similarity based on the longest common subsequence between the rewritten and original news. (iv) BERTScore (BertS) [41], which uses contextual embeddings to capture semantic similarity between the rewritten and original news.

4.2 Performance Comparison (RQ1)

We compare LANCE with several baseline rewriting methods, as outlined in Section 4.1.3, to evaluate their effectiveness in manipulating news RSs while preserving the naturalness of the rewritten text. The baselines consist of eight methods: six are not fine-tuned (GPT-4o, GPT-3.5 and Llama-3.1, each with and without context), and two are fine-tuned (Llama-FT$_{rec}$ and ATR-2FT). Other than the baselines, we also include a variant of our LANCE, "w/ Llama", which replaces GPT-4o with the open-source Llama-3.1 model in the *Explorer* stage. This variant keeps the *Reflector* unchanged, thereby reducing reliance on proprietary APIs and cutting exploration cost.

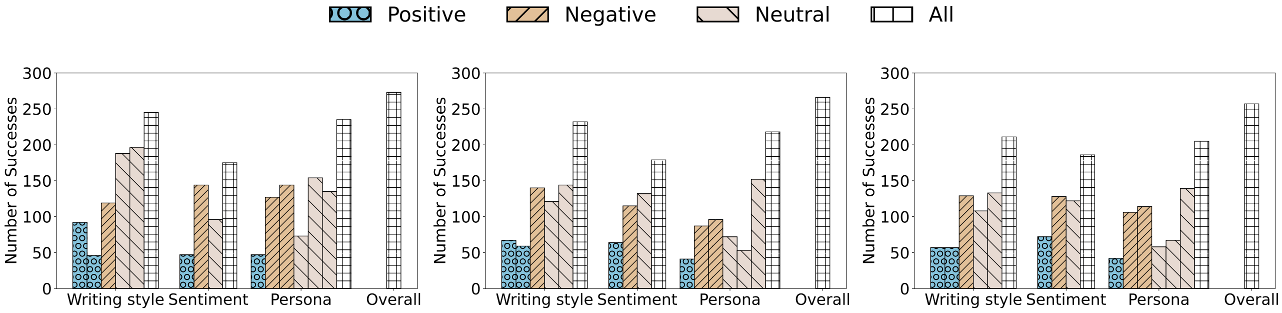
4.2.1 Attack Performance. The results of the attack performance are presented in Table 2. Below, we make the following three observations:

- LANCE outperforms all baselines across the three news RSs. This improvement is attributed to the diverse rewrites generated by the *explorer* and the quality control mechanism for successful rewrites. Fine-tuning on these collected rewrites enables the *reflector* to effectively align with the attack task.
- The open-source variant "w/ LLaMA" also improves the rank of the target news. It outperforms all non-fine-tuned baselines on every attack metric and even surpasses LANCE on LSTUR. This demonstrates that our framework remains effective when the *explorer* uses an open-source LLM, confirming its robustness and highlighting the critical role of the quality of exploration data.
- ATR-2FT gets the second-best attack performance in improving news rankings, demonstrating the effectiveness of its two-phase fine-tuning framework. However, it fails on NRMS and LSTUR. This is likely because both news RSs encode only title text, which limits the impact of content-based rewrites.
- All LLM-based baselines perform poorly on BSR and Rank but show improvements in Expo and Appear. This suggests that prompting LLMs without clear rewriting directions causes performance variance. Some rewrites boost rankings, but others result in significant rank drops. Additionally, using popularity-based context does not help, likely because dynamic popularity shifts make simulated popularity-based rewrites ineffective. Fine-tuning Llama on recommendation text also fails to help, as injecting recommendation information without guiding the attack objective is ineffective for the attack task.

4.2.2 Naturalness. The results regarding naturalness are shown in Table 3. We make the following observations:

Table 3: Comparison of naturalness performance. LSTUR is excluded due to a similar title rewriting method as NRMS. The yellow background in the PPL column indicates that ATR-2FT has worse perplexity than the original text.

Model		NAML				NRMS			
	Context?	PPL↓	BLEU↑	RougeL↑	BertS↑	PPL↓	BLEU↑	RougeL↑	BertS↑
Original	-	144.3	-	-	-	257.1	-	-	-
GPT-4o	Y	112.5	0.090	0.319	0.653	253.6	0.064	0.364	0.644
	N	124.3	0.097	0.333	0.660	223.7	0.068	0.388	0.652
GPT-3.5	Y	127.8	0.139	0.384	0.690	293.5	0.135	0.450	0.698
	N	145.9	0.127	0.372	0.678	290.5	0.138	0.445	0.687
Llama-3.1	Y	83.6	0.137	0.355	0.666	209.0	0.089	0.395	0.652
	N	74.6	0.140	0.350	0.660	200.4	0.083	0.391	0.647
Llama-FT$_{rec}$	-	79.2	0.141	0.357	0.666	199.2	0.087	0.399	0.653
ATR-2FT	-	179.7	0.090	0.313	0.636	323.9	0.070	0.411	0.657
w/ Llama	-	120.2	0.105	0.323	0.638	219.7	0.078	0.369	0.634
LANCE	-	79.6	0.120	0.328	0.649	195.8	0.075	0.401	0.649



(a) NAML (b) NRMS (c) LSTUR

Figure 3: Impact of diverse rewrites on attacking news RSs. 'All' represents the combined successful rewrites within each group, while 'Overall' denotes the total number of successful rewrites across all versions.

- On naturalness metrics, LANCE and its variant "w/ Llama" performs competitively across all baselines. It achieves a lower perplexity (PPL) than the original text, indicating improved fluency. Although its BLEU, RougeL, and BertS scores are not the highest, they remain reasonable, suggesting that rewrites retain sufficient similarity to the original text.
- In contrast, ATR-2FT increases perplexity compared to the original text, resulting in less natural output. This degradation likely results from its joint fine-tuning approach with a promotion loss mechanism, which updates the LM OPT parameters to prioritize attack success over text quality.
- The LLM-based baselines generally produce fluent and similar text, reflected in low PPL and high BLEU, RougeL, and BertS scores . However, their lack of attack-specific optimization limits their utility for improving rankings.

4.3 Diverse Rewrite Effectiveness (RQ2)

To investigate how different rewrite styles affect attack performance, we collect diverse rewrites of the target news articles (cf. Section 3.2) and measure the number of successful rank improvements for each rewrite version. To ensure a natural classification, we use GPT-4o to label each rewrite into three categories based on style: Positive, Negative, and Neutral.[4] Figure 3 shows the distribution of successful rewrites by category. We have the following observations:

- Diverse rewrites complement each other in attacks. Although the prompts differ in style, all versions of rewrites achieve successful attacks, contributing to a high overall success rate. This suggests that diverse rewrites can complement each other when attacking different news articles. Such diversity has the potential to enhance the *reflector*'s ability to generate effective attacks.
- Negative and neutral rewrites are more effective in attacking news RSs. Negative news often triggers curiosity, leading to more clicks and higher rankings. Neutral news, which emphasizes factual content, is typically more informative and straightforward. Such content is easier for recommendation systems to classify and surface based on relevance and clarity.
- Positive rewrites show lower success rates. Positive news often lacks urgency and fails to spark curiosity, resulting in lower engagement metrics such as clicks and shares. Additionally, since recommendation systems prioritize content with high engagement signals (*e.g.,* click-through rates, comments, and shares),

[4] Details can be found in our code repository https://github.com/Go0day/LANCE.

Table 4: Generalization comparison of attack performance across different news recommender systems.

Model	NAML				NRMS				LSTUR			
	BSR↑	Rank↓	Expo↑	Appear↑	BSR↑	Rank↓	Expo↑	Appear↑	BSR↑	Rank↓	Expo↑	Appear↑
Original	-	21,001	0.29	0.0058	-	20,970	0.11	0.0022	-	20,811	0.04	0.0007
w/ $\mathcal{V}_e^{\text{NAML}}$	-	-	-	-	0.53	20,193	0.16	0.0032	0.58	19,416	0.10	0.0020
w/ $\mathcal{V}_e^{\text{NRMS}}$	0.53	20,518	0.44	0.0088	-	-	-	-	0.55	19,700	0.07	0.0015
w/ $\mathcal{V}_e^{\text{LSTUR}}$	0.58	20,018	0.74	0.0149	0.52	20,562	0.20	0.0040	-	-	-	-
w/ $\mathcal{V}_e^{\text{ALL}}$	0.56	19,941	0.94	0.0187	0.54	20,174	0.27	0.0054	0.58	19,289	0.09	0.0018
LANCE	0.69	18,125	0.79	0.0159	0.56	18,677	0.41	0.0084	0.57	19,375	0.08	0.0016

they may rank positive news lower due to its relatively weaker interaction rates. This finding contrasts with attacks observed in other RSs domains, such as e-commerce [39], where positive descriptions are more effective. It highlights a unique vulnerability in the news recommendation scenario.

4.4 Generalization Capability in Cross-System Attacks (RQ3)

To evaluate the generalization capability of LANCE, we consider four versions of LANCE with different training and evaluation settings: (i) w/ $\mathcal{V}_e^{\text{NAML}}$ – fine-tuned on rewrites from NAML and evaluated on NRMS and LSTUR; (ii) w/ $\mathcal{V}_e^{\text{NRMS}}$ – fine-tuned on NRMS and evaluated on NAML and LSTUR; (iii) w/ $\mathcal{V}_e^{\text{LSTUR}}$ – fine-tuned on LSTUR and evaluated on NAML and NRMS; and (iv) w/ $\mathcal{V}_e^{\text{ALL}}$ – fine-tuned on all rewrites from the three news RSs and evaluated on all models.

We observe the following:

- *Cross-system generalization*: Table 4 shows that all versions of LANCE improve the target items' rankings and exposure rates, demonstrating the effectiveness of our framework. Notably, w/ $\mathcal{V}_e^{\text{NAML}}$, w/ $\mathcal{V}_e^{\text{NRMS}}$ and w/ $\mathcal{V}_e^{\text{LSTUR}}$ achieve performance gains on unseen models they were never trained on. This indicates that the textual attack model can generalize to unseen news RSs.
- *Effect of training on combined data*: The w/ $\mathcal{V}_e^{\text{ALL}}$ model performs worse than individually trained versions on NAML and NRMS but achieves better results on LSTUR. This suggests that while fine-tuning on specific models yields strong performance, training on combined data has the potential to enhance generalization, particularly on models like LSTUR.
- *Impact of data quality*: We observe that w/ $\mathcal{V}_e^{\text{ALL}}$ performs worse than w/ $\mathcal{V}_e^{\text{LSTUR}}$ when attacking NAML. This may be because NRMS employs multi-head self-attention to identify key words, making it more sensitive to semantic rather than stylistic rewrites. As a result, mixing data from NRMS causes the model trained on $\mathcal{V}_e^{\text{NRMS}}$ to become confused, leading to subpar performance on NAML. This highlights the role of data quality in textual attacks.

4.5 Effect on Wider Dimensions (RQ4)

Table 5: News recommendation performance before and after the LANCE attack. The "Change" row denotes the relative performance change.

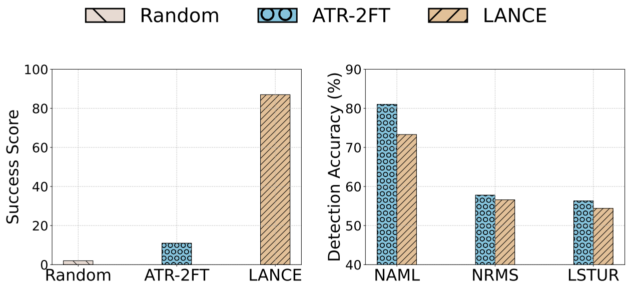
	Mode	AUC	MRR	nDCG5	nDCG10
NAML	Before	68.49	33.30	36.99	42.98
	After	68.41	33.28	36.97	42.93
	Change	-0.117%	-0.060%	-0.054%	-0.116%
NRMS	Before	66.39	31.24	34.10	40.85
	After	66.37	31.24	34.09	40.84
	Change	-0.030%	0.000%	-0.029%	-0.024%
LSTUR	Before	60.41	26.38	28.81	35.35
	After	60.34	26.36	28.79	35.32
	Change	-0.116%	-0.076%	-0.069%	-0.085%

4.5.1 Recommendation Performance. We assessed the impact of our LANCE attack on recommendation performance by replacing targeted news items ($\mathcal{V}_g$) with rewritten versions and evaluating key metrics: Area Under the Curve (AUC), Mean Reciprocal Rank (MRR), and normalized Discounted Cumulative Gain (nDCG) at 5 and 10. The results, shown in Table 5, indicate slight performance drops, with a maximum AUC decrease of 0.117% for NAML. These minimal changes demonstrate that LANCE disrupts recommendation rankings—our primary goal—while maintaining stealthiness. By altering the content of target news $\mathcal{V}_g$, our approach ensures minimal degradation, avoids detection by platform owners, and promotes the rank of target news (see Section 4.2), achieving a successful attack.

4.5.2 Semantic Preservation. In addition to evaluating naturalness metrics like BLEU, ROUGE-L, and BERTScore (as mentioned in Section 4.2), we conducted a simulated user study using LLM (GPT-4o) to directly assess how well LANCE preserves textual semantics compared to ATR-2FT. The methodology is outlined below:

- Candidate Preparation: We sampled 100 news articles from the test set as original targets. For each article, we created three candidate lists: (i) rewritten by LANCE, (ii) rewritten by ATR-2FT, both under identical attack conditions, and (iii) a random article from the same category.
- User Simulation: GPT-4o evaluated the three candidates for each article, selecting the one that best preserved the original semantics. This process was repeated five times per article, with the candidate receiving the most votes across the five runs earning one point. The Success Score reflects the total points accumulated across all 100 articles. List positions were randomized for each presentation to ensure fairness.

The results, shown in Figure 4a, indicate that LANCE achieves the highest Success Score, outperforming both ATR-2FT and the



(a) Success Scores **(b) Detection Accuracies**

Figure 4: (a) Success scores in simulated user study for semantic preservation, and (b) detection accuracies of rewritten content across victim models.

random baseline. This demonstrates LANCE's superior ability to retain semantic integrity during rewriting.

4.5.3 Detectability of Rewritten Content. As noted in Section 4.2, LANCE-generated attack text exhibits high naturalness, complicating human detection. However, its perplexity scores are notably lower than those of original news text, suggesting a detectable difference between LANCE-rewrited and original news content. To investigate potential defenses, we conducted a preliminary detection experiment, detailed as follows:

- Data Preparation: We combined original news text with rewritten versions from LANCE, labeling each as original or rewritten. The dataset was split into 70% training and 30% testing sets. We did the same to ATR-2FT, providing a reference about the detection accuracy.
- Text Processing: Using GPT-2 (which we used to calculate perplexity scores in Section 4.2), we computed probabilities for each token in the text sequences. These sequences were truncated or padded to a uniform length for consistent feature representation.
- Detection Model Training: A three-layer multilayer perceptron (MLP) was trained on the labeled training set to classify texts as original or rewritten. Separate models were trained for LANCE and ATR-2FT outputs.

Detection accuracies across victim models are presented in Figure 4b. The MLP achieves more than 70% detection accuracy for LANCE on NAML, indicating that language model probabilities can effectively identify rewritten text. LANCE proves stealthier than ATR-2FT, with consistently lower detection rates. However, accuracy drops on NRMS and LSTUR for LANCE and ATR-2FT, likely because these models encode news titles for recommendations, limiting the token probability information for detection. These findings highlight LANCE's stealth advantage, and reveal a promising direction for refining detection strategies.

5 Limitations and Broader Impact

Our work has several limitations. First, we use only the MIND dataset due to the limited availability of news recommendation datasets. Additionally, we limit our textual attacks to English news rewrites to focus on attack effectiveness, excluding other languages. Investigating performance on non-English news recommendation datasets [9, 13] is an important direction for future work. Second, we conduct attacks using Llama-3.1-8B and fine-tune it with the DPO method. Exploring other LLM versions, model sizes, and designing a specialized fine-tuning mechanism to better align the LLM with the attack task has the potential to yield further improvements. This would further highlight the vulnerability of news RSs to textual attacks. Third, we evaluate attacks on three news RSs models: NAML, NRMS, and LSTUR, which are commonly used benchmarks in the news recommendation domain [35, 44]. Due to the frequent updates in news domain, ID-based collaborative filtering methods are less suitable for news recommendation [44]. We leave the exploration of attacks on other recommendation techniques for future work.

Beyond limitations, our study has broader impacts. It reveals the vulnerabilities present in news RSs, which are shared across systems, thereby allowing textual attacks like LANCE to be effectively generalized. In today's world, and potentially more in the future, everyone can be not only a consumer of information but also a content provider. This means that users can easily conduct textual attacks, leading to severe consequences for user trust, platform integrity, and even societal cohesion. This underscores the urgent need for research into defense strategies to counter such attacks. Furthermore, given the importance of explored data, we release the GPT-4o-generated exploration results as a shared resource in our open-source code repository, enabling others to validate and extend our findings without incurring similar computational expenses. We believe our work can contribute to the development of more secure and socially responsible news RSs.

6 Conclusion

In this paper, we propose LANCE, an LLM-based news rewriting framework for textual attacks on news RSs. With a diverse *explorer* and fine-tuned *reflector*, LANCE generates rewrites that effectively boost rankings while preserving text naturalness and semantic meaning. Our results reveal a unique vulnerability in news RSs compared to other RSs domains (*e.g.,* e-commerce), as negative and neutral rewrites consistently outperform positive ones, highlighting a distinct ranking preference. Additionally, we demonstrate a shared vulnerability across different news RSs, as LANCE successfully attacks unseen systems. Notably, although our attack text achieves high naturalness – making it difficult for humans to detect – its perplexity scores are significantly lower than those of the original news text. This indicates a detectable difference between LLM-generated content and human-written text, which could be utilized to identify such attacks. A promising future direction is to investigating performance on non-English news recommendation systems, *e.g.,* using the recently released EB-NeRD dataset [13]. Another important direction is to develop defense mechanisms that exploit these differences between LLM and human text generation, helping news RSs mitigate textual attacks more effectively.

Acknowledgments

This research was (partially) funded by Ahold Delhaize, through AIRLab Amsterdam, the Dutch Research Council (NWO), under project numbers 024.004.022, NWA.1389.20.183, and KICH3.LTP.-20.006, and the European Union's Horizon Europe program under grant agreement No 101070212. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

[1] Mehmet Aktukmak, Yasin Yilmaz, and Ismail Uysal. 2019. Quick and accurate attack detection in recommender systems through user attributes. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 348–352.

[2] Mehwish Alam, Andreea Iana, Alexander Grote, Katharina Ludwig, Philipp Müller, and Heiko Paulheim. 2022. Towards analyzing the bias of news recommender systems using sentiment and stance detection. In *Companion Proceedings of the Web Conference 2022*. 448–457.

[3] Mingxiao An, Fangzhao Wu, Chuhan Wu, Kun Zhang, Zheng Liu, and Xing Xie. 2019. Neural news recommendation with long-and short-term user representations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 336–345.

[4] Huiyuan Chen and Jing Li. 2019. Data poisoning attacks on cross-domain recommendation. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2177–2180.

[5] Hung-Yun Chiang, Yi-Syuan Chen, Yun-Zhu Song, Hong-Han Shuai, and Jason S Chang. 2023. Shilling black-box review-based recommender systems through fake review generation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 286–297.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Bosheng Ding, Chengwei Qin, Ruochen Zhao, Tianze Luo, Xinze Li, Guizhen Chen, Wenhan Xia, Junjie Hu, Luu Anh Tuan, and Shafiq Joty. 2024. Data augmentation using LLMs: Data perspectives, learning paradigms and challenges. In *Findings of the Association for Computational Linguistics ACL 2024*. 1679–1705.

[8] Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 50–56.

[9] Jon Atle Gulla, Lemei Zhang, Peng Liu, Özlem Özgöbek, and Xiaomeng Su. 2017. The adressa dataset for news recommendation. In *Proceedings of the international conference on web intelligence*. 1042–1048.

[10] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685* (2021).

[11] Hai Huang, Jiaming Mu, Neil Zhenqiang Gong, Qi Li, Bin Liu, and Mingwei Xu. 2021. Data poisoning attacks to deep learning based recommender systems. *arXiv preprint arXiv:2101.02644* (2021).

[12] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 8018–8025.

[13] Johannes Kruse, Kasper Lindskow, Saikishore Kalloori, Marco Polignano, Claudio Pomo, Abhishek Srivastava, Anshuk Uppal, Michael Riis Andersen, and J. Frellsen. 2024. EB-NeRD: A large-scale dataset for news recommendation. *arXiv preprint arXiv:2410.03432* (Oct. 2024).

[14] Jian Li, Jieming Zhu, Qiwei Bi, Guohao Cai, Lifeng Shang, Zhenhua Dong, Xin Jiang, and Qun Liu. 2022. MINER: Multi-interest matching network for news recommendation. In *Findings of the Association for Computational Linguistics: ACL 2022*. 343–352.

[15] Miaomiao Li and Licheng Wang. 2019. A survey on personalized news recommendation technology. *IEEE Access* 7 (2019), 145861–145879.

[16] Xinyi Li, Yongfeng Zhang, and Edward C Malthouse. 2024. Prompt-Based Generative News Recommendation (PGNR): Accuracy and Controllability. In *European Conference on Information Retrieval*. Springer, 66–79.

[17] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.

[18] Qijiong Liu, Nuo Chen, Tetsuya Sakai, and Xiao-Ming Wu. 2024. ONCE: Boosting content-based recommendation with both open-and closed-source large language models. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*. 452–461.

[19] Yinhan Liu. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[20] Yu-An Liu, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024. Multi-granular adversarial attacks against black-box neural ranking models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1391–1400.

[21] Sejoon Oh, Gaurav Verma, and Srijan Kumar. 2024. Adversarial Text Rewriting for Text-aware Recommender Systems. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 1804–1814.

[22] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35 (2022), 27730–27744.

[23] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 311–318.

[24] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543.

[25] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2024).

[26] Shaina Raza and Chen Ding. 2022. News recommender system: a review of recent progress, challenges, and opportunities. *Artificial Intelligence Review* (2022), 1–52.

[27] Junshuai Song, Zhao Li, Zehong Hu, Yucheng Wu, Zhenpeng Li, Jian Li, and Jun Gao. 2020. PoisonRec: An adaptive data poisoning framework for attacking black-box recommender systems. In *2020 IEEE 36th international conference on data engineering (ICDE)*. IEEE, 157–168.

[28] Lawrence Van den Bogaert, David Geerts, and Jaron Harambam. 2024. Putting a human face on the algorithm: co-designing recommender personae to democratize news recommender systems. *Digital Journalism* 12, 8 (2024), 1097–1117.

[29] Sanne Vrijenhoek, Gabriel Bénédict, Mateo Gutierrez Granada, Daan Odijk, and Maarten de Rijke. 2022. Radio–Rank-aware divergence metrics to measure normative diversity in news recommendations. In *Proceedings of the 16th ACM conference on recommender systems*. 208–219.

[30] Zongwei Wang, Min Gao, Junliang Yu, Xinyi Gao, Quoc Viet Hung Nguyen, Shazia Sadiq, and Hongzhi Yin. 2024. LLM-Powered Text Simulation Attack Against ID-Free Recommender Systems. *arXiv preprint arXiv:2409.11690* (2024).

[31] Zongwei Wang, Junliang Yu, Min Gao, Wei Yuan, Guanhua Ye, Shazia Sadiq, and Hongzhi Yin. 2024. Poisoning Attacks and Defenses in Recommender Systems: A Survey. *arXiv preprint arXiv:2406.01022* (2024).

[32] Chenwang Wu, Defu Lian, Yong Ge, Zhihao Zhu, and Enhong Chen. 2021. Triple adversarial learning for influence based poisoning attack in recommender systems. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1830–1840.

[33] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with attentive multi-view learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. 3863–3869.

[34] Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with multi-head self-attention. 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)*. 6389–6394.

[35] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021. Empowering news recommendation with pre-trained language models. In *Proceedings of the 44th international ACM SIGIR Conference on Research and Development in Information Retrieval*. 1652–1656.

[36] 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*. 3597–3606.

[37] Yuting Yang, Juan Cao, Mingyan Lu, Jintao Li, and Chia-Wen Lin. 2019. How to write high-quality news on social network? predicting news quality by mining writing style. *arXiv preprint arXiv:1902.00750* (2019).

[38] Hengtong Zhang, Changxin Tian, Yaliang Li, Lu Su, Nan Yang, Wayne Xin Zhao, and Jing Gao. 2021. Data poisoning attack against recommender system using incomplete and perturbed data. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2154–2164.

[39] Jinghao Zhang, Yuting Liu, Qiang Liu, Shu Wu, Guibing Guo, and Liang Wang. 2024. Stealthy attack on large language model based recommendation. *arXiv preprint arXiv:2402.14836* (2024).

[40] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068* (2022).

[41] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* (2019).

[42] Xudong Zhang, Zan Wang, Jingke Zhao, and Lanjun Wang. 2022. Targeted Data Poisoning Attack on News Recommendation System by Content Perturbation. *arXiv preprint arXiv:2203.03560* (2022).

[43] Huanhuan Zhao, Haihua Chen, Thomas A Ruggles, Yunhe Feng, Debjani Singh, and Hong-Jun Yoon. 2024. Improving text classification with large language model-based data augmentation. *Electronics* 13, 13 (2024), 2535.

[44] Yuyue Zhao, Jin Huang, David Vos, and Maarten de Rijke. 2025. Revisiting Language Models in Neural News Recommender Systems. *arXiv preprint arXiv:2501.11391* (2025).