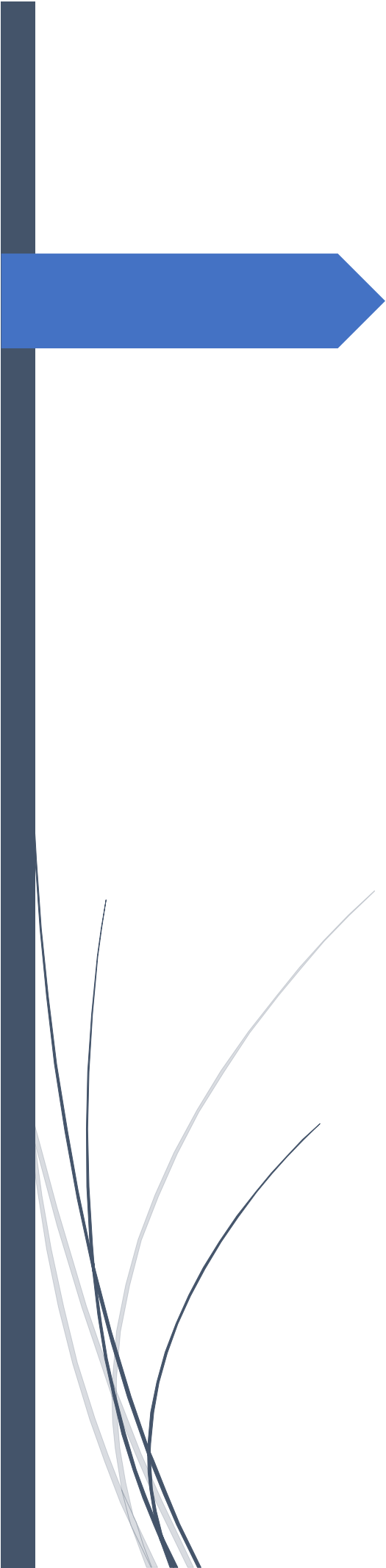




Interactive Social Media Federated Recommendation System via Chrome Extension

Odai Atef Atiya Abdelbaky
200025

Table of Contents

<i>Abstract</i>	2
<i>Introduction</i>	2
<i>Problem Statement</i>	3
<i>Motivation</i>	3
<i>Objectives</i>	4
<i>Literature Review</i>	4
<i>Datasets</i>	6
<i>Roadmap</i>	8
<i>References</i>	9

Abstract

Social media sites create abundant, ever-growing flows of information, making it increasingly hard for users to assess information of interest to them efficiently when it aligns with their interests, goals, and information needs. Current recommendation systems within these platforms mostly rely on passive behavioral signals likes, clicks, scrolling behavior, and dwell time to learn about user preferences. While effective for maximizing user engagement to the platform, these systems lack transparency, build a limit on user autonomy, and offer little to no information on the actual reasoning behind recommendations. Conventional recommendation algorithms also remain closely tied to the social media platform, which stops users from maintaining a consistent preference profile across social media platforms. All these difficulties culminate to lead to information overload, diminished trust, privacy violations, and feelings of distancing between recommended information and users' interests.

This project showcases an alternative, user-directed recommendation framework that is delivered as a Chrome extension, enabling users to explicitly shape their content experience. Rather than depending on platform-native algorithms, the proposed framework allows users to manually annotate posts that they see from their social media feeds as either **"Recommended"** or **"Unwanted."** The user's annotations will form their own data set reflective of their own explicit preferences as opposed to inferred behavioral patterns. A simple ML model that has been locally repurposed via Fine Tuning with techniques such as low-rank adaptation (LoRA) has the ability to use these annotations to learn user-specific relevance patterns and anticipate likely user interactions with upcoming posts based on what users have shared previously. Additionally, since all fine-tuning has taken place on the user device, it helps protect the user's privacy by eliminating dependency on centralized collections of user data and dictating that sensitive information about user behavior should not be collected or shared. The cross-platform website uses a Chrome extension, which provides the ability to customize your experience on both platforms without needing to rely on Private API or Backend Service access. Instead, it relies only on publicly visible text the browser extracts. This model provides compatibility and transparency and is not ethically objectionable, unlike some paradigms based on surveillance-inspired tracking. The recommendation model is inherently dynamic, developing over time as users contribute more annotations, and will become more accurate and personalized while allowing users to retain agency over the recommendations.

The recommendation framework is able to surface relevant posts, deprioritize unwanted content, and provide a transparent alternative to conventional opaque recommendation models. In sum, this project puts forward a novel, privacy-centric, user-centered approach to content recommendation based on explicit feedback, transparency, and user agency. It combines interactive annotation, lightweight learning, and cross-platform functionality to arise at a viable and ethical approach to improve content consumption online when algorithmic filtering and information overload dominate the field.

Introduction

Since social media has become the main source of communication, entertainment, education, and news for most, users can no longer keep track of the sheer volume of posts that are

produced every day. So to keep up with that, social media sites use recommendation algorithm-based recommendation engines to organize the content in user feeds and prioritize that feed based on the user's previous interactions. The recommendation algorithm-based recommendation engines are a significant portion of what users view and ultimately what they interact with.

Problem Statement

Although mainstream recommendation systems have increased user engagement, four significant shortcomings of these approaches exist. The first limitation is that the primary method for inferring user preferences is to use implicit behavioral signals (e.g., clicks, watch time, scrolling); however, these types of signals tend to be noisy and often easily misinterpreted. Secondly, because most mainstream recommendation algorithms are opaque (i.e., users do not receive any explanations about the content recommended), users are less likely to trust the algorithm and correct undesired patterns. Thirdly, most mainstream recommendation algorithms tend to optimize engagement for the platform (i.e., encouraging more time spent using the platform) instead of for goals of users, resulting in the emergence of irrelevant, repetitive, and/or sensationalized content in user feeds. Fourthly, most personalization models require extensive centralized tracking, raising significant privacy concerns, and restrict the learned preference profiles to the single platform where they were created, limiting personalizing experiences across platforms.

This research project attempts to solve this issue via user-directed recommendations via a Chrome extension. Users will explicitly label posts as “Recommended” or “Unwanted,” creating clear preference datasets. Using these explicit datasets, a lightweight model that runs locally on the user’s device will learn from user-generated feedback in order to make personalized predictions regarding future ranking and filtering decisions. By transferring control of personalized recommendations from platform-inferred data to user-annotated data and keeping the training local, this proposed system will provide a more transparent, private, and less overwhelming experience for users while providing more consistent personalized experiences across platforms.

Motivation

The social media platforms offer a considerable volume of user-generated and unregulated content, which often prevents users from engaging with content relevant to their specific interests. The recommendation systems in circulation today are often generally hidden, often centralized, and programmed for engagement rather than user satisfaction. These systems face issues with transparency and trustworthiness due to the usage of hidden algorithms and non-consensual behavior (behavioral) data collection. The aim of this work is to develop an interactive recommendation system that is transparent with user-centered personalization. If users can directly label content showing it is recommended or unwanted, the system will develop individual preferences using explicit feedback rather than implicit tracking. This improves accuracy of personalization and empowers users to control the metadata used to tune the model, making it a privacy-first approach to recommendation systems.

Another important motivating factor includes privacy concerns. Traditional algorithms rely on behaviorally extensive tracking and large-scale data captures that exist in centralized systems outside users' control. This data, which can be used for profiling and unintentional surveillance, carries with it privacy risks. While privacy increasingly becomes an item of concern for the public, concern for and inquiries banked on digital privacy pose a possible direction needed for online systems to provide valuable solutions to address data privacy by reducing data capture for personalization.

This project is also informed by the increasing shift in research toward user-centric and ethical AI systems. Modern users increasingly expect digital technologies to inform autonomy rather than behavioral manipulation. The proposed system allows explicit user annotation that is, users label content "Recommended Report" or "Unwanted Report" where clear principled reasoning can reveal transparency, intentionality, and fairness. Additionally, lightweight on-device fine-tuning or LoRA allows this personalization at the user level with little sacrifice to the privacy and efficiency of personalization technologies.

Overall, the need to rethink online recommendation systems to support user agency while encouraging ethical data practices, personalized data relevance content, and privacy-focused cross-platform filter alternatives to traditional algorithmic and online recommendations, equals the precepts of this project initiative. This initiative aims to establish a pedagogical practice that leads to endorsing user agency in directly enhancing their content experiences online, while responding to users' growing requests for more transparent, trustworthy, and human-centered digital technologies.

Objectives

Develop a Chrome extension capable of overlaying annotation buttons ("Recommended" and "Unwanted") on social media posts within supported platforms.

Collect labeled feedback from users to create a personalized training dataset reflecting individual content preferences.

Train and optimize a recommendation model that learns from these annotations to predict which future posts should be highlighted or hidden.

Implement local or lightweight model training to preserve user privacy and minimize data transfer.

Evaluate the system's effectiveness in improving content relevance and user satisfaction through both quantitative (accuracy, precision, recall) and qualitative (user experience) metrics.

Literature Review

A survey of Campana et al.(2023) [\[1\]](#) reported on the different types of recommender systems (RS) available for Online Social Networks (OSN) and Mobile Social Networks (MSN) that make use of social relationships and user mobility to improve the design of modern RSs. First, the paper introduced four main approaches to RSs: Collaborative Filtering (CF), Content-Based Filtering, Graph-Based Approaches, and Context-Aware Recommender Systems (CARS). As these approaches depend solely on existing conventional interaction data, each suffers the common shortcomings of this reliance: scalability issues, cold start problems, and sparser

feedback due to fewer interactions. The authors argue that social networking environments, such as OSNs, have the potential to significantly boost the quality of recommendations as they offer rich social signals, including friendship links/follow Markup tags, trust channels, tagging behavior, geographical contexts, and the availability of user-generated content. Additionally, they categorised the published literature on recommendation systems into the following system types: Friend Recommendations Based on Link Prediction, Social Trust Enhanced Item Recommendations, Tag Based Recommendations, and Geolocation Based Recommendations. MSN environments create a set of unique challenges for RSs relating to the decentralised nature of communication between devices, the intermittent nature of the connections between devices, limited device resource availability, and privacy issues; thus, RSs are required to work primarily with local device data resources and to rely on opportunistic communication to connect devices, rather than following a centralised model. The authors therefore reviewed multiple distributed and decentralised approaches to RSs including OppCF, MobHinter, differRS, locPref, and p-PLIERS, illustrating how RSs may continue to be successful even with the absence of a centralised approach.

Piccardi et al.'s (2024) [\[2\]](#) current study describes a newly developed methodology for conducting field studies of social media feed ranking using web browsers as a platform to conduct experiments and/or interventions. The impetus for this study is based upon the growing body of research regarding how feed ranking systems affect user well-being and the way in which users access information through their feeds, as well as the increasing number of restrictions placed on the access that platforms have to feed ranking and data selection systems. The authors have developed a methodology called a browser-based client-side 'reranking pipeline' that allows users to intercept the content of feeds as it is delivered to them, select which content they would like to see and apply the desired interventions to the selected content, thereby re-rendering the modified feed for their respective participants. The authors have also identified that there are three different types of intervention methods—namely, (1) downranking 'bad' content, (2) upranking 'good' and/or properly balanced content and (3) rewriting or editing content. Furthermore, the authors have discussed the technical challenges associated with implementation (e.g., inventory constraints, latency, consistency in handling user interface changes) as well as the ethical issues (consent, privacy and minimization of harm). The authors describe various ways to measure outcomes using a combination of survey data, passive behavioral metrics, and real-time ecological momentary assessments, as well as practical information about enrolling participants in the study, assigning participants to conditions, designing the interface, optimizing performance, and storing participant data securely. The study's conclusion is that extension-based reranking can facilitate ecologically-valid experimental research conducted in social media platforms, even if the social media company does not support such activity; therefore, the experimental designs used will still produce rigorous results for measuring the effect of interventions.

Hu et al.'s (2021) [\[3\]](#) study introduces LoRA (Low-Rank Adaptation) as a low-cost way of adapting a large number of language models (for example: GPT-3) for downstream tasks, without having to fine-tune/old model parameters for those tasks. The authors state that traditional fine-tuning for extremely large models (e.g., GPT-3) is costly in storage and computation, and that traditional fine-tuning methods (full tuning, adapter, prefix tuning) can add inference overhead (e.g., latency) and/or reduce available sequence length. To address these issues, LoRA uses the insight that many model fine-tuning updates are low-dimensional; therefore, if only low-dimensional updates are required to perform the task, then LoRA can

be used without the need to fully update a model for all tasks, unlike adapter and prefix tuning. LoRA's method involves freezing the pre-trained weights and representing ΔW , the update, as a product of two small matrices (BA) whose rank is extremely low, generally 1 or 2. The only matrices trained will therefore be BA, which dramatically reduces the number of parameters that need to be trained as well as the amount of memory consumed. The authors of this article find that during their experimentation with RoBERTa, DeBERTa, GPT-2, and GPT-3, LoRA is able to reach performance levels comparable to or higher than that of full fine-tuning, using as much as 10,000x fewer trainable parameters. Furthermore, LoRA allows for the merging of the LoRA updates into the base weights without introducing any notable inference latency. The analysis also proposes that LoRA learns low-rank and meaningful adaptations that are primarily concentrated in the attention projection matrices (e.g., W_q and W_v) and that they are stable across multiple trials. The conclusions drawn indicate that adaptation can be effectively captured in a low-rank space. Overall, the paper establishes LoRA as a valid way of scalably and practically adapting large pre-trained models to new tasks in cases where compute and memory limitations are a primary concern. According to Kong et al.'s (2025) [\[4\]](#) article, utilizing a single instance of a LoRA Module across an entire community of LLM-Based Sequential Recommenders creates a high risk for negative transfer due to model and user differences throughout the entire lifetime of the recommender system. The authors go on to describe how the common LLM-SR process creates the optimal training pathways for the creation of end-to-end training data via PEF-trained LoRA Modules for prompt-texts by converting user histories into prompts and utilizing prompt-context and expected translation prompts as potential output hypotheses within a retraining LLM process; however, they also show that different clusters of behavior have a tendency to present issues in the manner that their respective gradients misalign and lead to conflicts that arise during the creation of an optimal single LoRA Module'. They approach the challenge of large-scale sequence recommendation by developing a single multi-task training system with LoRA. Model updates use an MoE structure with a single gating network, driven by what a user has selected from prior recommendations and captured through a recurrent model like SARPS. Users identify "sequences," or sets of recommendations based on the user's selected sequence, and each sequence identifies its expert "weight." A recommendation generated by the MoE can then be combined with the user's prior selections to update formulations of LoRA for the specific user and sequence. iLoRA provides users with greater accuracy of their recommendations than does LoRA using fewer than 1% additional trainable parameters. A scalable method of providing instance-wise expert activation will also be beneficial to the future of personalization and elimination of negative transfer within sequential recommendation.

Datasets

This section describes the datasets utilized for the creation and assessment of a content only recommendation classifier (recommended and not recommended) throughout this project. Both public datasets available via Kaggle will be employed to establish, validate and evaluate the Language Understanding component, and a custom dataset created from real user submissions will be used as the primary data source for personalization. The datasets are listed below, with a description of the source they originate from and what fields are present

in each dataset and how they work within the overall architecture of the recommendation system.

1- Public Datasets (Kaggle Sources):

The datasets provide a wealth of example text that was necessary for initial experimentation, testing of preprocessing and (if required) pre-training and/or warm-starting the text classifier on the datasets before it is further personalised with LoRA.

Platform	Dataset	Description	Ref
Twitter	Sentiment140	1.6M tweets labeled by sentiment (positive, negative, neutral). Can be used to pretrain your embedding and text classification components.	[5]
Twitter	Twitter US Airline Sentiment	14K tweets labeled for sentiment. Smaller and good for experiments.	[6]
LinkedIn-style text	Job Posts Dataset	Job posts with titles and descriptions — simulates LinkedIn-style professional text.	[7]
1.3M LinkedIn Jobs & Skills	Linkedin Jobs	This dataset contains 1.3 million job listings scraped from LinkedIn in the year 2024.	[8]

2- Custom Data Collection from 5 Users (Primary Dataset):

The custom dataset created from real users is the most critical dataset in predicting individual preference between 'Recommended' and 'Not Recommended.' As such, public datasets do not represent the individual user preference; therefore, the custom dataset is the source of LoRA Personalisation.

What the custom dataset looks like:

- user_id
- post_id (local ID)
- platform (Twitter / LinkedIn)
- text (the full post content: paragraph, article snippet, long text, etc.)
- label (recommended or unwanted, or binary 1/0)
- timestamp

Example in JSON format

```
{  
  "user_id": "U03",  
  "post_id": "T7845",
```

```
"platform": "Twitter",  
"text": "",  
"label": "recommended",  
"timestamp": "2025-11-01T14:25:00"  
}
```

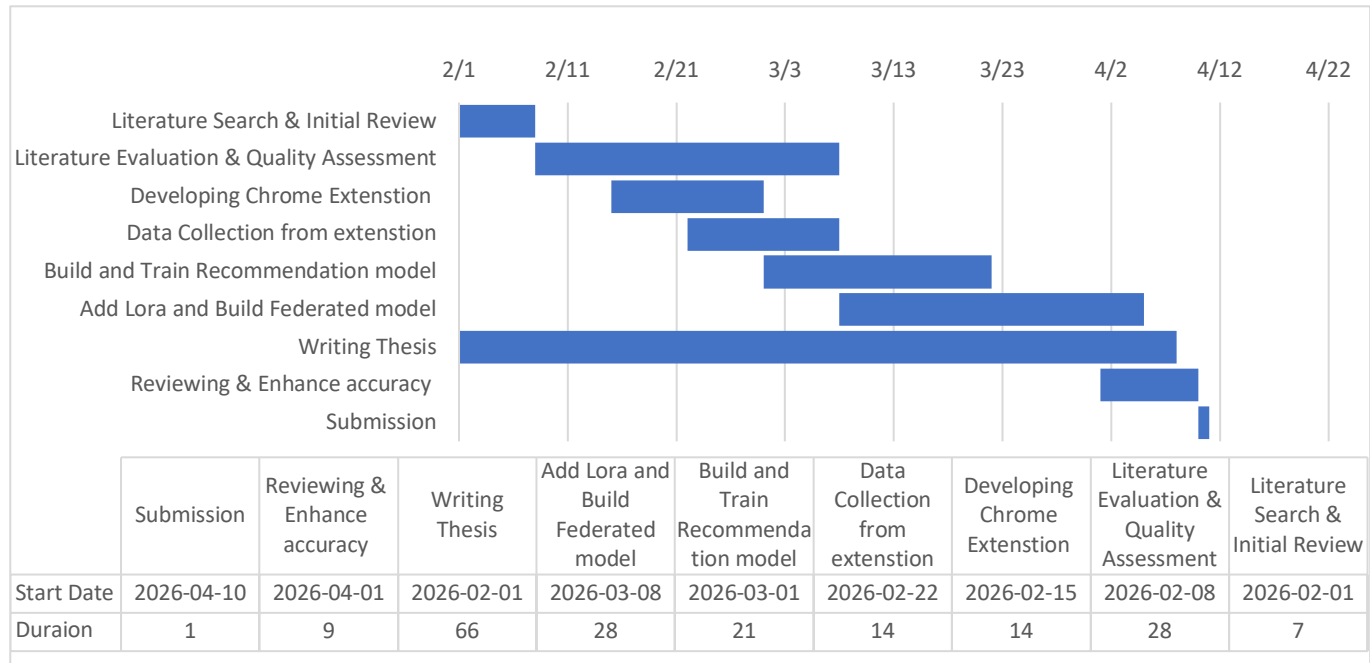
How we collect custom data from different users:

- 1- Select 5 participants (friends, classmates, or volunteers).
- 2- Each user installs your Chrome extension prototype.
- 3- The extension fetches a stream of posts (from public Twitter API, or LinkedIn posts) and displays them one by one.
- 4- Each post has two buttons: “Recommended” , “Unwanted”
- 5- The extension records the text of the post, platform/source (Twitter, LinkedIn), the user’s label (recommended/unwanted), timestamp and user ID (anonymized)
- 6- When a user clicks on a button, the extension will take an example of the user's label and either send it directly to your back end or store it locally until there is enough data to upload in bulk.
- 7- The backend will store all labels from users across different formats (CSV,JSON, and database).

Required Amount of Data Per User

- 100 labeled items from each user
- Roughly 500 labeled items total
- This is the minimum to test:
 - Training of Baseline Classifier
 - Fine-Tuning Using LoRA for Each User
 - User's Evaluation Of Improved Personalization Compared to Global Model

Roadmap



References

- [1] M. G. Campana and F. Delmastro, "Recommender systems for online and mobile social networks: A survey," *arXiv*, 2023. [Online]. Available: <https://arxiv.org/abs/2307.01207>.
- [2] T. Piccardi *et al.*, "Reranking social media feeds: A practical guide for field experiments," *arXiv*, 2024. [Online]. Available: <https://arxiv.org/html/2406>.
- [3] E. J. Hu *et al.*, "LoRA: Low-rank adaptation of large language models," *arXiv*, 2021. [Online]. Available: <https://arxiv.org/abs/2106.09685>.
- [4] X. Kong *et al.*, "Customizing language models with instance-wise LoRA for sequential recommendation," *arXiv*, 2025. [Online]. Available: <https://arxiv.org/abs/2408.10159>.
- [5] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," Stanford Univ., CS224N Project Rep., 2009.
- [6] CrowdFlower, "Twitter US Airline Sentiment," *Kaggle*, 2015. [Online]. Available: <https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment>.
- [7] K. Regmi, "Job Title and Job Description Dataset," *Kaggle*. [Online]. Available: <https://www.kaggle.com/datasets/kshitizregmi/jobs-and-job-description>.
- [8] asaniczka, "1.3M LinkedIn Jobs & Skills (2024)," *Kaggle*, 2024. [Online]. Available: <https://www.kaggle.com/datasets/asaniczka/1-3m-linkedin-jobs-and-skills-2024>.