國立臺灣大學管理學院資訊管理學系

碩士論文

Department of Information Management

College of Management

National Taiwan University

Master’s Thesis

運用內容相似度與大型語言模型提升轉推預測之表現

Enhancing Retweet Prediction Performance via Content-Based Similarity and Large Language Models

賴煒奇

Wei-Chi Lai

指導教授：陳建錦 博士、何承遠 博士

Advisors: Chien Chin Chen, Ph.D.; Cheng Yuan Ho, Ph.D.

中華民國 114 年 6 月

June 2025

doi:10.6342/NTU202501568

國立臺灣大學碩士學位論文
口試委員會審定書

運用內容相似度與大型語言模型提升轉推預測之表現

Enhancing Retweet Prediction Performance via Content-Based Similarity and Large Language Models

本論文係 賴煒奇 君（學號 R12725052）在國立臺灣大學資訊管理學系、所完成之碩士學位論文，於民國 114 年 6 月 23 日承下列考試委員審查通過及口試及格，特此證明

口試委員：何承遠 陳建錦

詹益禎

盧信銘

所　　長：盧信銘

doi:10.6342/NTU202501568

誌謝

首先，謹向何承遠老師表達誠摯的感謝。兩年來，何老師從文獻回顧、實驗設計、論文撰寫到口試簡報的準備，皆給予許多寶貴的建議，使我能順利完成論文。同時，感謝陳建錦老師於資訊檢索與文字探勘課程中所傳授的知識與經驗，令我受益匪淺。此外，感謝何承遠老師、陳建錦老師、盧信銘老師以及詹益禎老師在口試過程中所提供的寶貴建議，對本研究的修正與精進助益良多。感謝實驗室的聖傑學長協助電腦與網路的架設，使實驗得以順利進行。最後，衷心感謝我的父母及孜耘在過程中的支持與鼓勵，讓我得以順利完成碩士班的學業。

賴煒奇　謹識

于國立臺灣大學資訊管理學研究所

中華民國一一四年六月

doi:10.6342/NTU202501568

摘要

近年來，人們習慣透過社群平台接收資訊或表達自身觀點，其中在 Twitter (X.com) 平台上的「轉推」行為，是指分享一則已發佈的訊息，不僅可用以傳達使用者立場，亦有助於強化個人觀點。轉推預測任務作為社群媒體中探討資訊傳播的重要研究方向，旨在提升預測準確率，並藉由分析轉推行為，深入了解使用者偏好與其背後的決策因素。為了提升預測表現，許多研究提出深度學習模型應用於轉推預測任務。與傳統機器學習方法相比，深度學習不僅免除人工特徵工程的繁瑣程序，亦能顯著提升預測效果。隨著近年來大型語言模型（LLM）的快速發展，其於文本理解、摘要生成與推理等方面展現出強大能力，並廣泛應用於各種自然語言處理領域。然而，目前針對大型語言模型在轉推預測任務中的應用仍屬少見，有待進一步探討與發展。

本文聚焦於以內容為基礎的使用者轉推行為預測，利用使用者與目標推文作者的歷史發文紀錄，透過分析目標推文與這些發文紀錄之間的相似度，以預測使用者是否會轉發該目標推文。本研究提出一個創新的預測框架，結合可進行輸入權重分析的深度學習模型，識別對預測結果影響最顯著的輸入資料，並據此調整大型語言模型的提示指令，以提升其在轉推預測任務中的表現。此方法對未來將大型語言模型應用於類似任務具有重要的參考價值與貢獻。此外，本文所提出的基於內容相似度的深度學習模型具備簡化的模型架構，能以直觀的方式進行特徵歸因分析，同時展現出優異的預測效能與執行效率，為相關領域提供一個兼具解釋性與效能的實用解決方案。

關鍵字：轉推預測、深度學習、大型語言模型、特徵歸因、提示詞工程

doi:10.6342/NTU202501568

Abstract

In recent years, social media platforms have become central to how people receive information and express opinions. On Twitter (X.com), retweeting—sharing an existing post—serves both to express user stance and reinforce personal views. Retweet prediction is a key research area in understanding information diffusion, aiming to improve accuracy and reveal user preferences and decision-making factors. Deep learning models have been widely adopted for this task, offering superior performance over traditional machine learning by eliminating the need for manual feature engineering. With the rapid development of large language models (LLMs), their capabilities in text understanding and reasoning have been applied across various NLP tasks. However, their use in retweet prediction remains underexplored.

This study focuses on content-based retweet prediction, using the posting histories of both the user and the tweet author to analyze similarity with the target tweet. We propose a novel framework that combines a deep learning model capable of input weight analysis with prompt refinement for LLMs, improving their predictive performance. This approach offers valuable insights for applying LLMs to similar tasks. Additionally, our proposed Similarity-Based deep learning model features a simplified architecture that enables intuitive feature attribution, strong prediction performance, and efficient execution—making it a practical and interpretable solution for related research.

Keywords: Retweet prediction, deep learning, large language model, feature attribution, prompt engineering

doi:10.6342/NTU202501568

Table of Contents

doi:10.6342/NTU202501568

doi:10.6342/NTU202501568

List of Figures

doi:10.6342/NTU202501568

List of Tables

doi:10.6342/NTU202501568

doi:10.6342/NTU202501568

Chapter 1: Introduction

1.1 Background

In recent years, social media platforms have become integral to how individuals access information and express opinions. The rapid spread of information across these platforms has made them a crucial media for marketing, entertainment, and politics. Consequently, predicting information cascades, or the diffusion of information, has gained significant attention. One fundamental mechanism facilitating this spread is retweeting, which allow users to share an existing post on social platforms, often to express thoughts and participate in topic discussions [1]. Given its central role in content diffusion, retweet behavior prediction has emerged as a critical task, with wide-ranging applications including opinion mining [2], stock prediction [3], real-time event detection [4], and content recommendation [5]. Consequently, accurate prediction of retweets not only addresses theoretical challenges in modeling information diffusion but also delivers practical value across various applications, while offering deeper insights into user behavior and preferences on social platforms.

Retweet prediction tasks can be broadly categorized by granularity, focusing on either a global or local perspective. From a global perspective, the general characteristics of tweets are analyzed to evaluate their retweetability, forecast the overall popularity a post might achieve and modeling collective behavior, and address questions like "Why do some tweets get more retweets?". In contrast, the local perspective explores retweeting activity from an individual user’s standpoint, focusing on their decisions based on their profile and interests. It addresses questions such as "Which tweet will be retweeted by the user?" and "Who will retweet the target tweet?" [6].

doi:10.6342/NTU202501568

1.2 Research Motivations and Objectives

Various approaches, including machine learning and deep learning techniques, have been explored to address the task of retweet prediction. While deep learning models have achieved notable performance gains, their interpretability remains limited due to the "black-box" nature of neural networks [7]. Moreover, identifying the latent attitudes or behavioral patterns that drive a user's retweet decision remains a challenging task [6]. Previous studies [1], [8], [9] have primarily employed hyperparameter analysis or ablation study to investigate feature attribution—namely, the influence of individual input features to the model's prediction. Although few works [10] have addressed the issue of training time on deep learning models for retweet prediction, the inherent complexity of these models hinders training efficiency. Furthermore, although recent studies [11], [12], [13], [14] have demonstrated the effectiveness of large language models (LLMs) in recommendation tasks—particularly in user profiling and content generation—few prior studies, to the best of our knowledge, have explored the application of LLMs to the specific task of retweet prediction.

This study aims to improve the interpretability and training efficiency of deep learning models in retweet prediction by introducing an alternative approach to feature attribution. Furthermore, it evaluates the effectiveness of LLMs in this task, with a particular focus on enhancing their performance through prompt refinement. To this end, we propose a framework comprising two model categories: Similarity-Based and LLM-Based models. The Similarity-Based models simplify prior deep learning methods by using cosine similarity between the target tweet and user representations, enabling clearer feature interpretation, competitive performance, and improved time efficiency. The LLM-Based models utilize the language understanding and reasoning abilities of LLMs for prediction and self-explanation. Inspired by the proxy model approach [15], we

doi:10.6342/NTU202501568

incorporate insights from Similarity-Based models to refine LLM prompt instructions, aligning the model's attention with key predictive features and thereby enhancing overall performance.

1.3 Research Scope

In this research, we focus on the question: "Will a given tweet be retweeted by a specific user?" We adopt a local perspective on retweet prediction instead of a global perspective, which emphasizes overall tweet popularity—particularly viral content—rather than individual behavior. However, most tweets are not viral, and prior studies have shown that tweet popularity is largely influenced by factors such as the author's follower count [16]. In contrast, the local perspective places greater emphasis on the semantic relevance between tweets and users, making it more suitable for capturing personal interests and supporting applications like personalized recommendation. In line with prior studies [5], [8], [17], we concentrate on modeling user representations using content-based features, thereby excluding retweet decisions from the influence of social network structures. Here, content-based features refer to information extracted from the textual content of tweets, while social features relate to user relationships and interaction patterns.

To evaluate and generalize the proposed models, we conduct experiments on two datasets: a social media dataset (Twitter) and a scientific citation dataset (DBLP [18]). Both datasets contain rich textual content and reflect user interaction behavior—retweeting in Twitter and citation in DBLP. The DBLP dataset is considerably larger but exhibits greater sparsity, as users typically have fewer posting or citation activities. While the DBLP dataset is entirely in English, the Twitter dataset is predominantly English with some multilingual entries. Nonetheless, both the embedding models used in the Similarity-Based methods and the LLMs employed in the LLM-Based methods are capable of processing multilingual content.

1.4 Research Contributions

The key contributions of our work are summarized as follows:

1. This work introduces Similarity-Based models featuring a simplified deep learning architecture that preserves competitive predictive performance while enhancing interpretability through more transparent feature attribution and improving execution time by at least 23% compared to baseline models.
2. We explore LLM-Based models as an initial effort to apply LLMs to the retweet prediction task. In addition, we analyze LLM-generated self-explanations to identify which input features are deemed most influential during inference.
3. We propose a novel framework that connects deep learning models with LLMs by combining weight analysis from Similarity-Based models with the self-explanatory capabilities of LLMs. By designing different prompt variants that guide the LLM to focus on specific input feature relationships, we observe results that align closely with the weight analysis. This approach leads to a 10% improvement in F1 score on the Twitter dataset.

doi:10.6342/NTU202501568

Chapter 2: Related Works

Current research on retweet prediction employs various approaches. Many studies leverage features extracted from tweet content and user profiles, utilizing machine learning models such as logistic regression, support vector machines (SVM), and random forests. Others apply statistical methods, including the Hierarchical Dirichlet Process (HDP) [19] and Conditional Random Fields (CRF) [20]. Some researchers frame retweet prediction as a ranking problem, adopting techniques such as learn-to-rank approaches [21], matrix factorization [22], and collaborative ranking [23]. With the recent advances in deep learning models, there has been a growing focus on utilizing architectures including convolution models, recurrent models, attention-based models for their efficiency and ability to learn optimal features automatically [6]. Driven by advancements in LLMs in the past few years, research in the field of recommender systems has increasingly focused on methods that leverage the summarization and inference abilities of LLMs to capture user preferences, rank candidate tweets, or predict user behaviors [24].

2.1 Retweet prediction

2.1.1 Machine Learning Approaches

Previous machine learning research has explored various predictive models and features to enhance the performance of retweet prediction tasks. Firdaus et al. [5] develop retweet prediction models using XGBoost, Random Forest and matrix factorization, incorporating both explicit (hashtags, URLs, mentions) and implicit (topics, emotion, personality) features from users' tweets and retweets. They found that user behavior features significantly improved prediction accuracy. Using only past retweets for user profiling reduced processing time while maintaining comparable accuracy, highlighting the value of psychological and topical user traits in modeling information diffusion on Twitter.

doi:10.6342/NTU202501568

Tang et al. [25] introduce IRBLRUS, a prediction model that frames individual retweet behavior as a multi-task learning problem. Each user is treated as a separate task, modeled by a personalized logistic regression function composed of a shared global parameter and a user-specific component. The model integrates a user similarity–based regularization term that encourages users with similar characteristics to have aligned prediction parameters. Similarity is computed from structural, profile, and content-based features—such as mutual follow relationships, demographic attributes, and topic distributions via LDA—enabling the model to leverage social proximity for more coherent and personalized retweet predictions.

Luo et al. [21] address the challenge of identifying which followers are likely to retweet a given tweet by modeling it as a ranking problem. Using a learning-to-rank framework, they incorporated features such as retweet history, follower status, active times, and shared interests. Their experiments showed that the most effective predictors were past retweet behavior and interest similarity between users and tweets. The combined feature model significantly outperformed random and history-based baselines, indicating the importance of personal interaction history and content alignment in predicting retweeters.

2.1.2 Deep Learning Approaches

Recent studies have explored various deep learning models for retweet prediction by leveraging tweet content, user interests, and social interactions. Zhang et al. [8] propose an attention-based deep neural network (SUA-ACNN) for retweet prediction, integrating tweet content, user and author embeddings, user interests, and tweet-user similarity. Tweet content is encoded using a CNN, while user interests are extracted by clustering a user's historical tweets using K-means and selecting central tweets as interest representations. These are weighted using an attention mechanism. User and author

doi:10.6342/NTU202501568

embeddings are learned as trainable continuous vectors during model training, capturing latent social features. All components are combined in a Softmax layer for prediction.

Ma et al. [1] introduce AUT-MSAM, a masked self-attentive model for retweet prediction that integrates user interests and hot topics from users' social exposures. It encodes user and author histories using a hierarchical attention memory network, capturing both word- and tweet-level semantics. Concurrently, a masked self-attention mechanism processes the recent tweets of a user's followees to detect trending topics, applying a Transformer-based architecture with masking to emphasize relevant content. These components are combined and passed through a multi-layer perceptron (MLP) to predict retweet behavior.

Huang et al. [17] addressed the problem of predicting which topics a user will join by proposing MACNN, a memory-based convolutional neural network with attention mechanism for predicting which social media topics a user will join in the future. The model utilizes two key inputs: the user's posting history and topic participation history. Both are represented as tweet collections embedded in external neural memory. An attention mechanism identifies the most relevant content, and convolutional layers extract semantic features. These features are combined and passed through a multi-layer perceptron to predict future topic participation.

2.2 Utilizing LLMs for Recommendation Tasks

Since the emergence of LLMs such as ChatGPT, an increasing number of studies have investigated the application of LLMs to various recommendation tasks, employing zero-shot, few-shot, and fine-tuning strategies to evaluate their effectiveness. Kang et al. [26] investigate the capacity of LLMs to understand user preferences through the lens of user rating prediction, a foundational task in recommendation systems. The study systematically evaluates various LLMs across zero-shot, few-shot, and fine-tuning paradigms. The methodology involves prompting LLMs with user interaction histories and item metadata to predict user ratings. The findings reveal that while zero-shot and few-shot LLMs underperform compared to traditional recommender systems that leverage user interaction data, fine-tuned LLMs demonstrate competitive performance and superior data efficiency.

Liu et al. [27] explore the use of ChatGPT as a general-purpose recommender system without fine-tuning, applying it to five key tasks: rating prediction, sequential recommendation, direct recommendation, explanation generation, and review summarization. While ChatGPT lags behind traditional models in accuracy-based tasks, the study demonstrates its strong capability in producing explanations and summaries, as validated through human evaluations. These findings underscore ChatGPT's potential for enhancing interpretability in recommendation systems, despite its current limitations in predictive accuracy.

2.3 Advances in Interpreting and Evaluating Explanations in LLMs

Recent research has explored various strategies to enhance the interpretability and evaluation of LLMs, focusing on both the generation and assessment of natural language explanations. Krishna et al. [15] propose AMPLIFY, a framework that enhances LLM performance by using post hoc explanation methods rather than human-annotated rationales. It generates natural language rationales by extracting key input features—specifically, top-k tokens with the highest attribution scores—using gradient-based explanation techniques applied to a smaller proxy model such as BERT or GPT-2. These rationales are incorporated into in-context learning prompts to guide LLM predictions. Experiments demonstrate that AMPLIFY improves performance across reasoning and language tasks, highlighting the potential of post hoc explanations to enhance both

doi:10.6342/NTU202501568

interpretability and effectiveness.

Huang et al. [28] study ChatGPT's ability to generate self-explanations for sentiment analysis, comparing them to traditional interpretability methods like occlusion and LIME. Using two prompting strategies—explain-then-predict and predict-and-explain—they evaluate explanation faithfulness and consistency. While self-explanations perform similarly to traditional methods on faithfulness metrics, they show low agreement in terms of which words are considered important, suggesting that LLMs may require new interpretability frameworks.

Huang et al. [29] investigate the potential of ChatGPT to evaluate the quality of natural language explanations (NLEs) by comparing its assessments with human annotations across different granularity levels. Their method involves rating explanations from three benchmark datasets using clarity and informativeness scores, and analyzing alignment via classification and pairwise comparison tasks. The key contribution lies in demonstrating that ChatGPT approximates human judgment more accurately in coarse-grained settings and comparative evaluations, highlighting its viability as a scalable, cost-effective alternative to human evaluators in subjective quality assessment tasks.

doi:10.6342/NTU202501568

Chapter 3: Methodology

3.1 Problem Formulation

The retweet prediction task addresses the problem of predicting whether a user will retweet a specific target tweet. In this study, we adopt a content-based approach, inspired by prior work [8], [17], deriving user representations solely from the textual content of tweets. This method excludes temporal information and social network structure. The input data for each user-tweet pair includes the following components:

1. Target Tweet Content: This serves as the primary input for prediction. In the Twitter dataset, it includes the complete tweet text along with hashtags, mentions, and links. In the DBLP dataset, it corresponds to a paper abstract. This content provides the semantic context needed to assess its relevance to the user's interests.

2. User's Posting History: It represents the user's preferences and interests as inferred from prior activity. In the Twitter dataset, this includes past tweets—comprising text, hashtags, mentions, and links—excluding multimedia content. In the DBLP dataset, it consists of paper abstracts, reflecting the user's academic focus and research interests.

3. User's Retweet History: Like the posting history, this captures user preferences, but with a stronger emphasis on content the user actively engages with. In the Twitter dataset, it includes the textual content of retweeted tweets, while in the DBLP dataset, it is represented by citations of the user's papers, reflecting academic connections. As noted by Firdaus et al. [5], retweet history tends to play a more significant role in user profiling than posting history, as it more directly captures the content users actively choose engage with.

4. Target Tweet Author's Posting History: Analyzing the author's prior activity

enhances the interpretation of the target tweet. In the Twitter dataset, where tweets are often brief and informal, the author’s posting history offers valuable context regarding their interests and communication style, aiding in understanding the current tweet. Likewise, in the DBLP dataset, the author’s previous publications—captured through abstracts—provide insight into their research trajectory, enriching the evaluation of a new paper’s relevance.

3.2 Overall Framework

Building upon the AMPLIFY framework proposed by Krishna et al. [15], we propose a novel framework consisting of two distinct model categories: Similarity-Based models and LLM-Based models. The Similarity-Based models employ a simplified deep learning architecture that delivers competitive predictive performance while enabling interpretability through feature attribution. In contrast, the LLM-Based models represent one of the earliest applications of LLMs to the task of retweet prediction. As illustrated in Figure 3.1, both models generate predictions based on identical input features described in Section 3.1. However, the LLM-Based models offer the additional capability of providing self-explanations to justify their predictions. Moreover, interpretability insights obtained through weight analysis of the Similarity-Based models can be leveraged to refine prompts used in LLM-Based models, thereby enhancing their predictive performance.

Our framework and AMPLIFY share a common objective: leveraging insights from a proxy model to refine prompts for LLMs to enhance their performance. As shown in Table 3.1, we adopt the Similarity-Based model as a proxy due to its proven effectiveness in retweet prediction and its relatively low computational cost. In contrast, the AMPLIFY framework employs models such as GPT-2 and BERT, which require backward propagation to compute gradients, thereby incurring significantly higher computational

doi:10.6342/NTU202501568

overhead. A key distinction lies in the attribution approach: whereas AMPLIFY conducts word-level attribution for each instance to dynamically extract keywords, our framework emphasizes the four input components defined in Section 3.1 to maintain alignment with the weight analysis of the Similarity-Based model. This design results in a consistent and stable prompt structure across instances.

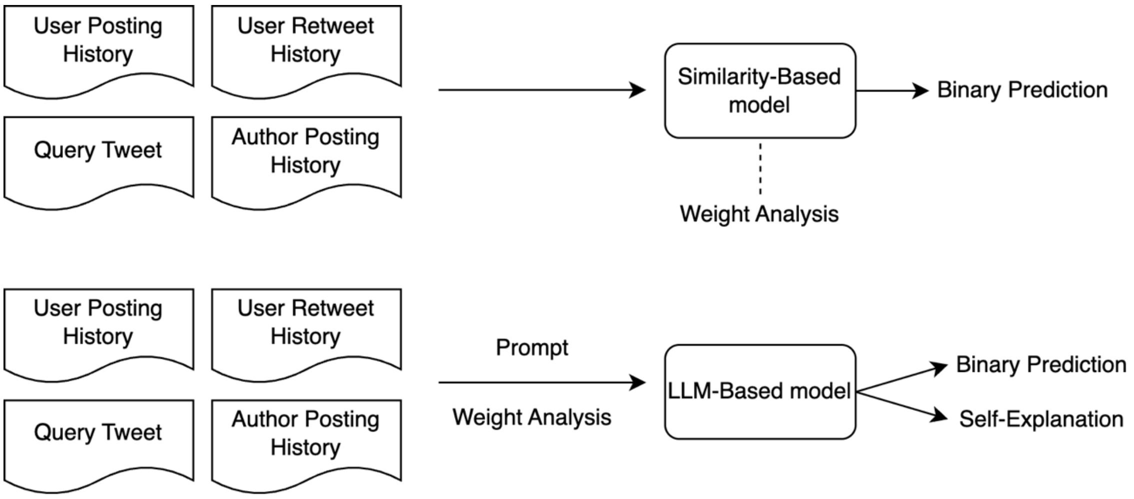

Figure 3.1: Overview of the Proposed Framework

Table 3.1: Comparative Analysis of AMPLIFY and the Proposed Framework

	AMPLIFY	Proposed Framework
Task	Reasoning and language understanding	Retweet prediction
LLM	GPT-3, GPT-3.5	Gemini 2.0 Flash Lite
Proxy Model	GPT-2, BERT	Similarity-Based Model
Performance of Proxy Model	Low	High
Method	Gradient-based method	Weight analysis
Attribution Granularity	Word-level (token-level)	Input-component level
Adaptability	Dynamic	Static
Computation Requirement	High	Low

doi:10.6342/NTU202501568

3.3 Similarity-Based Models

Previous deep learning approaches [1], [8], [17] have employed various techniques to derive user representations based on their posting or retweet histories. However, the final prediction layer in these models is typically composed of an MLP layer, which processes either concatenated user representations or those derived from similarity scores.

Inspired by the effectiveness of employing cosine similarity over user- or content-related features in previous machine learning approaches [5], [25], we propose a streamlined deep learning architecture that solely utilizes similarity scores between the embedding of a target tweet and user-specific representations. These user representations are constructed through the aggregation of pretrained embeddings corresponding to individual tweets. This design facilitates a more interpretable framework for analyzing feature attribution, thereby enhancing the performance of LLM-Based models. As illustrated in Figure 3.2, representations corresponding to the user's posting history, retweet history, and the author's posting history are derived through straightforward aggregation of pretrained embeddings. Both the target tweet and each of these aggregated representations are independently passed through separate fully-connected layers. Cosine similarity is subsequently computed between each of the target tweet and the representations. This process yields six similarity scores, which are concatenated and fed into an MLP. The MLP comprises a hidden layer with ReLU activation, followed by an output layer with two units. A Softmax function is then applied to the output layer to generate probabilities for binary classification.

doi:10.6342/NTU202501568

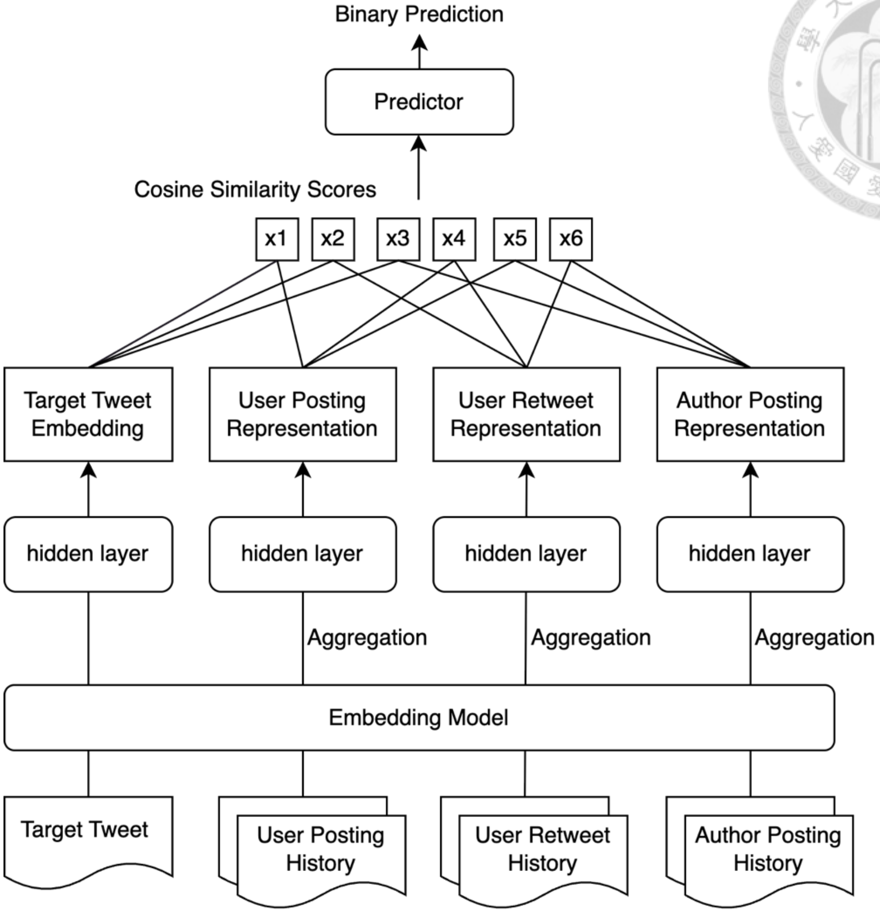

Figure 3.2: Architecture of Similarity-Based models

While earlier approaches commonly derived sentence embeddings by aggregating word embeddings from pretrained models such as Word2Vec [30] or GloVe [31], recent advancements [32] highlight the performance of Transformer-based embedding models. In this study, we employ the widely used all-MiniLM-L6-v2 model to generate embeddings for individual tweets. A comparative analysis of traditional encoding techniques—including TF-IDF and LDA—and other Transformer-based embeddings is presented in Section 4.7.2. Additionally, in Section 4.7.3, we evaluate the effectiveness of various aggregation methods—specifically averaging, clustering, and attention mechanisms—in order to identify the most appropriate approach for constructing comprehensive user representations.

doi:10.6342/NTU202501568

3.4 LLM-Based Models

Recent progress in LLMs offers new possibilities for examining retweet prediction performance and interpretability. Despite emerging LLM-based studies on engagement [33] and response prediction [34], our proposed LLM-Based models represent an early exploration of LLMs for retweet prediction. As shown in Figure 3.3, LLM-Basic directly utilizes the input data defined in Section 3.1 to generate retweet predictions based on prompt instruction A (Appendix A). To ensure alignment with Similarity-Based models, the prompt instruction A emphasizes content and semantic similarity within the input data, leveraging the natural language understanding capabilities of LLMs for this task.

A key distinction between the LLM-Basic model and the Similarity-Based models lies in their approach to user representation. While the Similarity-Based models aggregate tweet embeddings to construct user profiles, the LLM-Basic model relies on direct semantic similarity comparisons. To evaluate the impact of explicitly constructing user representations for guiding the LLM's attention toward salient themes and entities, we introduce the LLM-Modular model, as illustrated in Figure 3.4. This model employs a summarization component (Appendix B.1) to extract user preferences from historical tweets and retweets, thereby mirroring the aggregation strategy used in the Similarity-Based models to construct user representations. Subsequently, predictions are generated using Instruction A. Moreover, the LLM-Modular model articulates user preferences in natural language, offering greater interpretability for domain experts compared to the latent vector representations used in Similarity-Based approaches. This design choice supports downstream applications such as content recommendation, trend forecasting, and advertisement campaign optimization.

doi:10.6342/NTU202501568

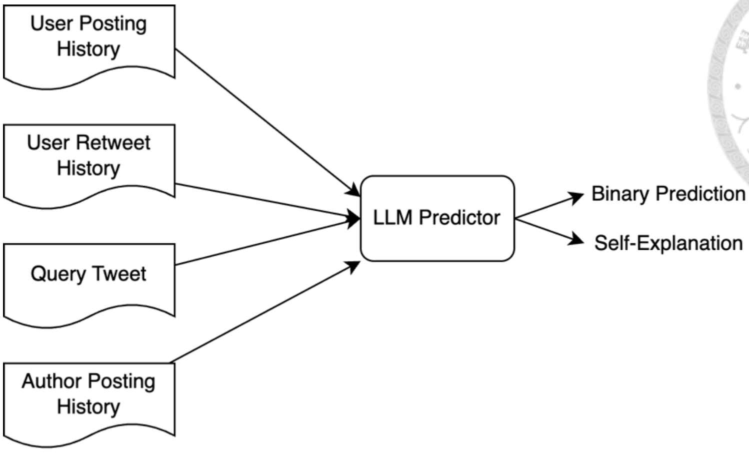

Figure 3.3: Architecture of LLM-Basic

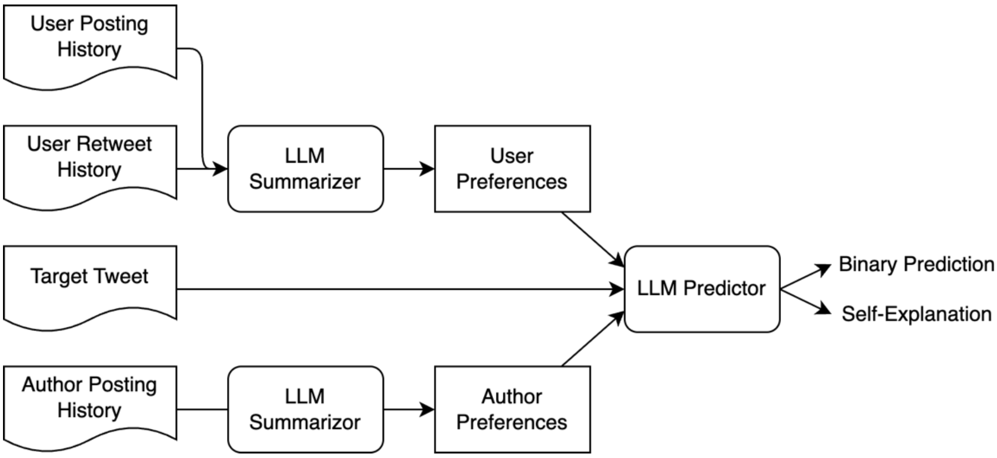

Figure 3.4: Architecture of LLM-Modular

We leverage the DSPy library [35] to implement structured and declarative prompts with defined input/output fields and task-specific instructions, further adapted in Section 4.6.3 based on weight analysis of Similarity-Based models. As shown in Table 3.2, the input consists of four components outlined in Section 3.1. The output includes the predicted class along with a self-explanation, which consists of a paragraph-length rationale and the identification of the most influential input feature contributing to the LLM's prediction. To ensure consistency with the weight analysis of Similarity-Based models, the *most_important_input* field is restricted to a predefined set of six relationships among the four input elements.

doi:10.6342/NTU202501568

Table 3.2: Input and Output Field Definitions

	Field name	Data type
Input Fields	target_tweet	str
	user_retweet	List[str]
	user_tweet	List[str]
	author_tweet	List[str]
Output Fields	prediction	Literal[True, False]
	explanation	str
	most_important_input	Literal["target_tweet and user_tweet", "target_tweet and user_retweet", "target_tweet and author_tweet", "user_tweet and user_retweet", "user_tweet and author_tweet", "user_retweet and author_tweet"]

3.5 Feature Attribution of Similarity-Based Models

To further enhance the performance of LLM-Based models and leverage insights from Similarity-Based models for prompt refinement, we focus on feature attribution—the process of identifying the input features that most significantly influence a user's decision to retweet. Traditional machine learning approaches rely on hand-crafted features for users and tweets, enabling straightforward assessment of feature importance through model-assigned weights.

In contrast, deep learning models employ high-dimensional vector representations, making it more difficult to isolate and interpret the influence of individual features. To bridge this gap, previous deep learning approaches [1], [8], [9] have commonly relied on ablation study to assess the significance of different features. The proposed Similarity-Based models, depicted in Figure 3.2, employ a simplified architecture to address this issue. These models take cosine similarity scores between various input representations as input to the predictor, enabling clearer analysis through a direct examination of the predictor's weights.

3.5.1 Weight Analysis for Similarity-Based Models

Given the inherently non-linear nature of predictors in Similarity-Based models, we apply an approximation technique that substitutes the MLP predictor with a single linear layer. This allows for direct inspection of the learned weights to identify which inputs most strongly influence the classification outcome. The approximated predictor is modeled as a linear transformation producing two outputs, y_0 and y_1, which represent the predicted probabilities for class 0 (not retweet) and class 1 (retweet), respectively. We use a two-dimensional output with a Softmax function instead of a one-dimensional output with a sigmoid function to allow for future extensions, such as adapting the model to downstream tasks that require multi-class classification.

As shown in equations (1), (2) and (3), the class corresponding to the higher value between y_0 and y_1 is selected as the predicted label. The model takes six input features, denoted as x1 to x6, each associated with a corresponding weight in the formula.

$$\boldsymbol{y_0 = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + w_5x_5 + w_6x_6 + b} \qquad (1)$$

$$\boldsymbol{y_1 = w'_1x_1 + w'_2x_2 + w'_3x_3 + w'_4x_4 + w'_5x_5 + w'_6x_6 + b'} \qquad (2)$$

1. x_1: Cosine similarity between the target tweet and the user's posting history.
2. x_2: Cosine similarity between the target tweet and the user's retweet history.
3. x_3: Cosine similarity between the target tweet and the author's posting history.
4. x_4: Cosine similarity between the user's posting history and the user's retweet history.
5. x_5: Cosine similarity between the user's posting history and the author's posting history.
6. x_6: Cosine similarity between the user's retweet history and the author's posting history.

$$\mathbf{Class} = \begin{array}{l} \boldsymbol{Class\ y_0,\ \ if\ y_0 \geq y_1} \\ \boldsymbol{Class\ y_1,\ \ if\ y_0 < y_1} \end{array} \qquad (3)$$

3.5.2 Prompt Refinement and Self-Explanation of LLM-Based Models

The prompt refinement strategy directs the LLM to focus on the most influential input relationship—one of x_1 to x_6—identified through corresponding weight analysis. For instance, if w_1 exhibits the largest absolute value, indicating that the relationship between the target tweet and the user's posting history has the greatest impact on the prediction, a targeted prompt is appended to the original Instruction A to form Instruction A.1 (Appendix A.1) as below:

Your ONLY task is to determine the semantic relationship between a target tweet and a specific user's past tweet history to predict retweet likelihood.

IMPORTANT: Focus EXCLUSIVELY on the relationship between the target tweet and the user's tweet history. Other data points are irrelevant to your analysis.

Remember: The ONLY relationship that matters is between the target tweet and the user's tweet history. Disregard any other information that does not directly inform this specific relationship.

To align with the weight analysis performed in the Similarity-Based models, we require the LLMs to produce two additional outputs for each prediction, as illustrated in Table 3.2: an explanation and the identification of the most important relationship between input features. The explanation is a brief paragraph justifying the LLM's prediction. The most important relationship is selected from the six predefined input feature pairs outlined in Section 3.5.1, with the LLM instructed to choose one from this set. This approach enables us to examine which input relationship most strongly influences the LLM's decision, allowing for targeted refinement of the prompt instruction—analogous to adjusting predictor weights in the Similarity-Based models.

3.6 Comparison of Similarity-Based and LLM-Based Models

A detailed comparison of the two proposed methods is presented in Table 3.3. Both models utilize the raw text of tweets as input; however, the Similarity-Based model explicitly transforms the input into pretrained embeddings. While both approaches produce binary predictions, the LLM-Based models additionally generate self-explanations to facilitate further analysis. Feature attribution analysis is conducted for both modeling approaches. In the case of the Similarity-Based models, we utilize ablation study and weight analysis to evaluate the contribution of individual input features. For the LLM-Based models, feature attribution is assessed through analysis of the generated self-explanations.

Table 3.3: Comparison of Similarity-Based models and LLM-Based models

Aspect	Similarity-Based models	LLM-Based models
Input data	Raw text	Raw text
User Representation Construction	Aggregated embeddings	Textual summarization
Feature Comparison Method	Cosine similarity	Natural language reasoning
Prediction Mechanism	MLP with ReLU and Softmax	Prompt-based binary classification
Output	Binary classification	Binary classification and self-explanations
Performance Sensitivity	Depends on embedding and aggregation choice	Depends on prompt design and LLM capabilities
Feature Attribution Analysis	Ablation study and weight analysis	Self-explanation analysis

doi:10.6342/NTU202501568

Chapter 4: Experiments

4.1 Dataset Construction

The proposed methods were evaluated on two real-world datasets—Twitter (X.com) and DBLP—to assess the generalization capability of the models. The Twitter dataset was collected over a three-month period (March 1 to May 31, 2024). To ensure diversity in user interests, seed users were randomly selected from various topical domains, including sports, politics, business, cooking, and nature. The followees of these seed users were subsequently crawled to construct a representative set of user communities. The DBLP dataset [18], a publicly available citation network, contains rich textual content and citation interactions analogous to user retweets, making it a common benchmark in information cascade studies [36], [37], [38]. Each research paper includes an abstract, authors, publication year, venue, and title. To facilitate alignment with the structure of social media data, researchers were mapped to users, abstracts to tweets, citations to retweets, and co-authorship relationships to followee connections.

For the Twitter dataset, only tweets containing more than five words were considered. To filter out likely fake or inactive accounts, users were required to have more than 20 followees and followers, and to have posted between 50 and 3,000 tweets during the three-month observation period. Retweets were included only if the author of the original tweet was also present in the dataset. Although 164,284 retweets were initially collected, the majority were excluded based on this criterion. For the DBLP dataset, the following filtering criteria were applied: each paper—treated analogously to a tweet—was required to have both an abstract and a title, and to be published between 2000 and 2017. Furthermore, each author—considered as a user—was required to have published at least five papers in each of the two time periods: 2000–2011 and 2012–2017. This temporal partitioning aligns with the Twitter dataset setup, where the earlier period (2000–2011) is

doi:10.6342/NTU202501568

used to model user preferences. Detailed statistics for the DBLP dataset are provided in Table 4.1.

Table 4.1: Statistics of Twitter and DBLP dataset

Dataset	Number of users	Number of tweets and retweets	Size of training set	Size of validation set	Size of testing set
Twitter	2,514	183,332	9,440	2,023	2,023
DBLP	16,667	488,088	11,200	2,400	2,400

Following the experiments of Firdaus et al. [5], posting and retweet history from the first two months were employed to model user preferences, while the final month's data was partitioned into training, validation, and testing subsets. For the Twitter dataset, 6,743 positive retweets from the third month were paired with an equal number of negative ones. As suggested by previous research [1], [5], [8], negatives were selected from a user's timeline and consisted of tweets by followees during the same period that were not retweeted. For the DBLP dataset, 8,000 citations were selected from the third period as positive instances, with an equal number of negative instances drawn from papers published in the same year for balance. Although retweet prediction typically samples negatives from a user's followees (i.e., coauthors), this constraint was relaxed to account for citations to non-coauthors. The positive and negative retweets from the third period of both datasets are partitioned into training, validation, and testing sets using a 70%, 15%, 15% split, respectively.

4.2 Experiment Configuration

All experiments were conducted on a local machine equipped with a 13th Generation Intel® Core™ i5-13400F processor operating at 2.50 GHz, comprising 10 cores. The system was also equipped with an NVIDIA GeForce RTX 3050 GPU with 8 GB of dedicated video memory. It had 64 GB of DIMM RAM and a total storage capacity of 954 GB. The operating system used was Windows 10 Education. The implementation was

doi:10.6342/NTU202501568

carried out using Python with the PyTorch framework, and Visual Studio Code was employed as the primary development environment. Additionally, Gemini 2.0 Flash Lite was accessed via API and utilized as the LLM for both summarization and prediction tasks.

4.3 Evaluation metrics

To evaluate the performance of the proposed models, we employ widely adopted metrics from previous studies, including accuracy, precision, recall, and F1 score [6]. Among these, recall is critical in many retweet prediction tasks, as it reflects the model's capacity to identify a greater number of actual retweets [5]. However, certain prompt instructions tend to bias LLMs toward generating a higher number of positive predictions, resulting in elevated recall but reduced precision. This trade-off undermines the overall reliability of the predictions. Therefore, we emphasize the F1 score as a more balanced and informative metric for evaluation.

4.4 Baseline Models

1. Random: In this approach, the decision to retweet or not is made randomly for each tweet, without taking into account the tweet content, user profile, or author information. Given that retweet prediction is formulated as a binary classification task, the expected values for accuracy, precision, recall, and F1 score are all approximately 50%, assuming a balanced dataset.

2. SUA-ACNN: We implemented the method proposed in [8], which integrates user and author information with attention mechanisms for retweet prediction. The model utilizes Word2Vec to generate word embeddings and applies convolutional neural networks (CNNs) to encode the content of tweets. Tweets of a user are then clustered into a predefined number of n clusters using the K-means algorithm and extract a fixed number of central tweets to represent the user's interest. An

doi:10.6342/NTU202501568

attention-based module then assigns attention weights to the most relevant clusters. User embeddings are randomly initialized and jointly trained with the model. To enhance predictive accuracy, a similarity score between the user interest representation and the target tweet is computed and concatenated with the user, author, and tweet embeddings as input to the final prediction layer.

3. MACNN: We implemented the method proposed in [17], adapting it for the retweet prediction task. Since the original model predicts user engagement with a topic, we restructured the framework by replacing the target topic with tweets posted by the author, and redefining the topic vector as the target tweet. MACNN combines two main components: posting history and participation history. For both, CNNs extract useful representations from tweet sets, while an external memory module retrieves the most relevant content. Specifically, the number of randomly selected tweets for each user or topic is set to 20. An attention mechanism assigns higher weights to important tweets. The model then measures similarity between users and topics, combines the features, and uses a multilayer perceptron to generate the final prediction.

4.5 Retweet Prediction Performance

Tables 4.2 and 4.3 present a comparative performance analysis of the Similarity-Based model, LLM-Based models, and baseline methods across both datasets. The Similarity-all-MiniLM-L6-v2 model employs the all-MiniLM-L6-v2 embedding model to encode tweets and constructs user representations by averaging the corresponding tweet embeddings. Despite its relatively simple architecture, the Similarity-all-MiniLM-L6-v2 model demonstrates competitive performance, comparable to MACNN and lower than SUA-ACNN. These results underscore the model's effectiveness in predicting retweet behavior while maintaining architectural simplicity. A more detailed evaluation

doi:10.6342/NTU202501568

of different embedding models and user representation strategies is provided in Section 4.7.

Table 4.2: Comparison results on the Twitter dataset

Model	Accuracy	Precision	Recall	F1 score
Random	49.0%	48.6%	50.6%	49.6%
SUA-ACNN	94.5%	93.3%	95.7%	94.5%
MACNN	91.3%	89.4%	93.6%	91.5%
Similarity-all-MiniLM-L6-v2	90.7%	92.3%	88.6%	90.4%
LLM-Basic (A.5)	70.1%	65.1%	88.2%	74.9%
LLM-Modular	67.6%	61.8%	90.5%	73.4%

Table 4.3: Comparison results on the DBLP dataset

Model	Accuracy	Precision	Recall	F1 score
Random	50.2%	50.6%	51.9%	51.3%
SUA-ACNN	91.3%	95.5%	86.91%	91.0%
MACNN	87.4%	98.1%	76.6%	86.0%
Similarity-all-MiniLM-L6-v2	87.5%	87.3%	88.1%	87.7%
LLM-Basic (A.2)	78.9%	82.9%	73.1%	77.7%
LLM-Modular	78.4%	76.7%	82.1%	79.3%

For the LLM-Based models, LLM-Basic (A.5) and LLM-Basic (A.2) refer to variants utilizing prompts defined in Appendices A.5 and A.2, respectively. These prompts guide the model to make retweet predictions based on specific semantic relationships, as further explained in Section 4.6.3. In comparison to deep learning models, the LLM-Based approaches generally exhibit inferior performance, aligning with the findings of Kang et al. [26], which indicate that LLMs without fine-tuning underperform traditional supervised methods in user rating prediction tasks. Nonetheless, the LLM-Based models consistently achieve F1 scores above 70%, demonstrating their potential effectiveness for retweet prediction in a zero-shot setting.

In addition, the LLM-Modular model does not demonstrate significant performance improvement over the LLM-Basic variants, indicating that the incorporation of an additional summarization module to construct user preferences for retweet prediction remains an open research question. While deep learning models, including Similarity-all-MiniLM-L6-v2, perform worse on the DBLP dataset compared to the Twitter dataset—potentially due to DBLP's data sparsity and longer textual content limiting the effectiveness of user representation—the LLM-Based models achieve better performance on the DBLP dataset. This suggests that LLMs exhibit stronger capabilities in handling sparse data and longer textual inputs compared to deep learning models.

4.6 LLM Prompt Refinement through Feature Attribution

For the Similarity-Based models, an ablation study is first conducted—following prior approaches [1]—to evaluate the contribution of individual input features. Subsequently, a weight analysis is performed to assess the significance of relationships among different input features. The insights gained from these analyses are then used to guide the refinement of prompts for the LLM-Based models.

4.6.1 Ablation Study of Similarity-Based Models

As outlined in Section 3.1, four input features are used for retweet prediction: the target tweet, the user's posting history, retweet history, and the posting history of the target tweet's author. The Similarity-all-MiniLM-L6-v2 model is first evaluated using all features, followed by ablation studies removing one feature at a time. Results in Tables 4.4 and 4.5 show that excluding either the user's posting or retweet history does not significantly degrade performance, indicating these features can compensate for each other in constructing user representations.

The model shows notable performance degradation on the Twitter dataset when the author's posting history is removed, while excluding the tweet has little impact. This may

doi:10.6342/NTU202501568

be due to the often brief or ambiguous nature of tweets, as noted in Section 3.1, making the author's posting history essential for inferring context. Additionally, users may rely more on their perception of the author than on a single tweet's content. In contrast, on the DBLP dataset, removing the target tweet causes a greater performance drop, whereas the author’s posting history has minimal effect. This difference likely reflects dataset characteristics: academic abstracts are typically detailed and structured, facilitating topic identification, and citations are generally based on content rather than author familiarity.

Table 4.4: Ablation studies on the Twitter dataset

Removed feature	Accuracy	Precision	Recall	F1 score
All features kept	90.7%	92.3%	88.6%	90.4%
Target tweet	90.2%	93.3%	86.4%	89.7%
User’s posting history	90.1%	90.3%	89.5%	89.9%
User’s retweet history	91.2%	93.1%	88.7%	90.9%
Author’s posting history	72.9%	72.7%	72.4%	72.5%

Table 4.5: Ablation studies on the DBLP dataset

Removed feature	Accuracy	Precision	Recall	F1 score
All features kept	87.5%	87.3%	88.1%	87.7%
Target tweet	83.6%	83.0%	84.8%	83.9%
User’s posting history	87.2%	87.2%	87.4%	87.3%
User’s retweet history	86.2%	82.9%	91.5%	87.0%
Author’s posting history	85.5%	83.5%	88.9%	86.1%

4.6.2 Weight Analysis Results for Similarity-Based Models

To assess the contribution of input features in the Similarity-Based model, the non-linear MLP is replaced with a linear transformation. The performance of the original Similarity-all-MiniLM-L6-v2 model is then compared with its linear variant, as shown in Tables 4.6 and 4.7. The results indicate that this modification does not degrade performance and even leads to slight improvements on the Twitter dataset.

doi:10.6342/NTU202501568

Table 4.6: Comparison results between linear and non-linear predictor on the Twitter dataset

Method	Accuracy	Precision	Recall	F1 score
Non-linear	90.7%	92.3%	88.6%	90.4%
Linear	92.0%	91.3%	92.7%	92.0%

Table 4.7: Comparison results between linear and non-linear predictor on the DBLP dataset

Method	Accuracy	Precision	Recall	F1 score
Non-linear	87.5%	87.3%	88.1%	87.7%
Linear	87.3%	90.8%	83.4%	86.9%

Next, we examine the weights of the six input features described in Section 3.5.1. As shown in Table 4.8, weights w_3, w_5, and w_6 have relatively larger absolute values, indicating greater influence on retweet prediction for the Twitter dataset. All three correspond to the author's posting history, consistent with the ablation study where their removal caused the greatest performance drop. However, all three weights associated with class y_1 are negative, indicating that a higher similarity between the user's representation and the author's does not contribute to a positive retweet outcome, which is contrary to expectations.

Table 4.8: Weight analysis on the Twitter dataset

class	w_1	w_2	w_3	w_4	w_5	w_6	b
y_0	-1.58	-0.07	2.42	0.25	3.32	2.62	0.27
y_1	1.52	0.27	-2.67	-0.52	-3.06	-2.58	-0.72

As shown in Table 4.9, weight w_2 as the largest absolute value, indicating that the similarity between the target tweet and the user's retweet history has the greatest impact on retweet prediction in the DBLP dataset. This aligns with the ablation study, where removing the target tweet resulted in the most significant performance decline.

Table 4.9: Weight analysis on the DBLP dataset

class	w_1	w_2	w_3	w_4	w_5	w_6	b
y_0	-1.07	-4.18	-2.46	0.07	1.01	-2.67	1.03
y_1	1.11	4.23	2.46	-0.01	-1.18	2.61	-1.02

4.6.3 Leveraging Weight Analysis to Refine LLM Prompts

In the following experiments, prompt instructions A and A.1 to A.6 (see Appendices A and A.1–A.6) are utilized to direct the LLM's attention toward specific input relationships for retweet prediction, as detailed in Section 3.5.1. Instruction A serves as the baseline and does not specify which input relationships to emphasize, allowing the LLM to make its own determination. In contrast, instructions A.1 to A.6 explicitly direct the LLM to focus on relationships x_1 to x_6, respectively, which represent the six possible relationships among the four input components. As discussed in Section 3.4, the LLM outputs one of these six relationships as the most influential factor for its prediction in each instance. The distribution of input importance derived from these predictions is presented in Tables 4.12 and 4.15.

In the Twitter dataset, the LLM guided by instruction A primarily focuses on x_2, which captures the relationship between the target tweet and the user’s retweet history. However, the associated weight w_2 for the y_1 class is relatively low in the weight analysis, which may partially explain its relatively lower performance. Instruction A.5, corresponding to the highest absolute weight w_5, achieves the highest recall and F1 score among all configurations, marking a 10% improvement in F1 score compared to instruction A. As shown in Table 4.11, instruction A.3 yields a higher rate of false positive predictions and exhibits the most degraded performance. This result is conceptually consistent, as the relationship between the target tweet and the author's own posting history is arguably less pertinent to predicting the user's retweet behavior. Nonetheless, several inconsistencies are observed. Despite the smaller absolute values of w_2 and w_4 in the weight analysis, the corresponding Instructions A.2 and A.4 exhibit only marginal performance degradation.

Table 4.10: Comparison of original and refined instructions on the Twitter dataset

Model	Accuracy	Precision	Recall	F1 score
Instruction A	67.8%	70.9%	59.4%	64.6%
Instruction A.1	70.1%	66.5%	79.8%	72.6%
Instruction A.2	70.6%	67.6%	77.9%	72.4%
Instruction A.3	55.2%	53.1%	82.3%	64.5%
Instruction A.4	66.0%	61.1%	86.3%	71.5%
Instruction A.5	70.1%	65.1%	88.2%	74.9%
Instruction A.6	69.0%	64.6%	82.6%	72.5%

Table 4.11: Confusion matrices of original and refined instructions on the Twitter dataset

Model	True Positive	False Negative	False Positive	True Negative
Instruction A	595	407	244	777
Instruction A.1	800	202	403	618
Instruction A.2	781	221	374	647
Instruction A.3	825	177	730	291
Instruction A.4	865	137	551	470
Instruction A.5	884	118	473	548
Instruction A.6	828	174	454	567

Table 4.12: Distribution of input importance on the Twitter dataset

Model	x_1	x_2	x_3	x_4	x_5	x_6
Instruction A	132	1,500	272	78	39	2
Instruction A.1	1,896	127	0	0	0	0
Instruction A.2	2	2,020	1	0	0	0
Instruction A.3	4	10	2,009	0	0	0
Instruction A.4	265	402	0	1,356	0	0
Instruction A.5	101	44	47	0	1,654	177
Instruction A.6	0	96	14	0	3	1,910

doi:10.6342/NTU202501568

In the DBLP dataset, the LLM guided by instruction A automatically focuses on x_1, x_2 and x_3, which correspond to positive weights in the weight analysis. This alignment contributes to its competitive performance relative to other variants. Instruction A.2, which directs the LLM to focus on x_2—the relationship associated with the highest weight—achieves the best F1 score and accuracy among all configurations. Consistent with the findings from the Twitter dataset, instruction A.3 results in the highest number of false positive predictions and the poorest overall performance, due to its focus on a relationship that is conceptually unrelated to the user's retweet behavior. Some inconsistencies remain, however; although w_4 exhibits relatively low absolute value in the weight analysis, the LLMs guided by the corresponding instructions still achieve competitive performance.

Table 4.13: Comparison of original and refined instructions on the DBLP dataset

Model	Accuracy	Precision	Recall	F1 score
Instruction A	77.8%	79.1%	76.1%	77.6%
Instruction A.1	77.3%	79.1%	74.6%	76.8%
Instruction A.2	78.9%	82.9%	73.1%	77.7%
Instruction A.3	61.8%	57.5%	92.6%	71.0%
Instruction A.4	69.1%	63.6%	90.6%	74.7%
Instruction A.5	75.4%	71.6%	84.8%	77.6%
Instruction A.6	73.3%	69.0%	85.5%	76.4%

Table 4.14: Confusion matrices of original and refined instructions on the DBLP dataset

Model	True Positive	False Negative	False Positive	True Negative
Instruction A	921	289	243	947
Instruction A.1	903	307	238	952
Instruction A.2	885	325	182	1,008
Instruction A.3	1,121	89	827	363
Instruction A.4	1,094	113	624	563
Instruction A.5	1,026	184	407	783
Instruction A.6	1,035	175	466	724

Table 4.15: Distribution of input importance on the DBLP dataset

Model	x_1	x_2	x_3	x_4	x_5	x_6
Instruction A	405	823	974	93	101	4
Instruction A.1	2,345	52	3	0	0	0
Instruction A.2	33	2,361	3	3	0	0
Instruction A.3	1	4	2,395	0	0	0
Instruction A.4	360	590	7	1,431	3	3
Instruction A.5	148	12	245	2	1,966	27
Instruction A.6	4	48	70	1	26	2,251

In conclusion, our approach of guiding the LLM to focus on the input relationship associated with the highest absolute weight in the weight analysis achieves the best performance across both datasets. While Instruction A—allowing the LLM to autonomously determine which input relationships to emphasize—effectively identifies influential features in the DBLP dataset and yields competitive results, our method offers more consistent performance, particularly in light of Instruction A's relatively poor results on the Twitter dataset. Nonetheless, the observed weight analysis does not consistently correlate with the performance of the corresponding LLM prompts across both datasets, indicating potential areas for further refinement.

4.7 Experimental Evaluation of Efficiency, Embedding Models, and User Representations

4.7.1 Execution Time Analysis

Tables 4.16 and 4.17 compare the execution times of various models on the training sets of two datasets: 2,023 instances for Twitter and 2,400 for DBLP. The proposed Similarity-Based models utilize a pre-encoding strategy in which tweets are processed using pretrained embedding models, and the resulting tensors are stored in external memory prior to training. In contrast, SUA-ACNN encodes tweets by applying convolutional layers to word embeddings and learns user embeddings through random initialization during training. MACNN constructs tweet embeddings using a trainable word embedding matrix that is optimized during runtime.

The pre-encoding strategy employed by the Similarity-Based models significantly reduces overall execution time by eliminating redundant computations during training and decreasing the number of trainable parameters through a simplified model architecture. Specifically, the Similarity-Based models achieve a reduction in total execution time ranging from 23% to 50% when compared to MACNN, and from 67% to 78% compared to SUA-ACNN.

Table 4.16: Comparison of training and testing durations on the Twitter dataset

Model	Encoding Time	Training Time	Testing Time	Total Time
Random	0	0	1s	1s
SUA-ACNN	0	1hr 20m 13s	37s	1hr 20m 50s
MACNN	0	22m 18s	15s	22m 33s
Similarity-all-MiniLM-L6-v2	6m 55s	10m 23s	3s	17m 21s
LLM-Basic	0	0	4m 46s	4m 46s
LLM-Modular	0	0	18m 26s	18m 26s

Table 4.17: Comparison of training and testing durations on the DBLP dataset

Model	Encoding Time	Training Time	Testing Time	Total Time
Random	0	0	1s	1s
SUA-ACNN	0	1hr 5m 33s	46s	1hr 6m 19s
MACNN	0	43m 26s	22s	43m 48s
Similarity-all-MiniLM-L6-v2	5m 3s	16m 54s	5s	22m 2s
LLM-Basic	0	0	7m 3s	7m 3s
LLM-Modular	0	0	23m 52s	23m 52s

In contrast, the LLM-Based models perform inference during the testing phase without prior training. Among these, the LLM-Modular model incurs higher execution time than the LLM-Basic model due to the additional summarization step. The LLM-

doi:10.6342/NTU202501568

Basic model requires only 0.15 to 0.2 seconds per prediction, making it suitable for practical applications. In contrast, the Similarity-Based model takes 0.5 to 0.6 seconds per prediction on average, but remains practical for many use cases.

4.7.2 Embedding Model Analysis

Tables 4.18 and 4.19 present a comparative analysis of various embedding models, including TF-IDF, LDA, and Sentence-Transformer approaches. In all cases, user representations are constructed by averaging the embeddings of their tweets, followed by classification using a Similarity-Based model. The results indicate that even relatively simple encoding methods, such as TF-IDF and LDA, can achieve competitive performance because they are capable of capturing the underlying semantic representations of tweets. Their lexical and topical features align well with the cosine similarity-based architecture of the Similarity-Based model, enabling effective content comparison and prediction.

However, the methods construct embeddings from the posting and retweet history corpus, which may not be feasible in real-world scenarios with limited user data. In such cases, pre-trained embedding models offer a more practical solution for end-to-end predictions due to their ability to generate representations from minimal input.

Table 4.18: Evaluation of embedding models on the Twitter dataset

Embedding model	Dimension	Accuracy	Precision	Recall	F1 score
all-MiniLM-L6-v2	384	90.7%	92.3%	88.6%	90.4%
paraphrase-MiniLM-L3-v2	384	90.1%	90.8%	89.2%	90.0%
TF-IDF	24,261	91.8%	93.0%	90.2%	91.6%
LDA	100	86.7%	84.3%	89.9%	87.0%

Table 4.19: Evaluation of embedding models on the DBLP dataset

Embedding model	Dimension	Accuracy	Precision	Recall	F1 score
all-MiniLM-L6-v2	384	87.5%	87.3%	88.1%	87.7%
paraphrase-MiniLM-L3-v2	384	83.4%	86.5%	79.4%	82.8%
TF-IDF	40,745	78.0%	76.4%	81.6%	78.9%
LDA	100	85.1%	84.1%	86.9%	85.5%

4.7.3 User Representation Method Analysis

Tables 4.20 and 4.21 present a comparison of different methods for generating user representations using the same embedding model, all-MiniLM-L6-v2. The average method computes the mean of all tweet embeddings, while the clustered-average method averages the embeddings of the k central tweets selected through K-means clustering. The attention method, which combines embeddings using attention weights, performs worse on both datasets—likely due to limited data, which hinders effective learning of the attention mechanism.

Table 4.20: Evaluation of user representation on the Twitter dataset

Method	Accuracy	Precision	Recall	F1 score
Average	90.7%	92.3%	88.6%	90.4%
Attention	88.4%	87.1%	89.9%	88.5%
Clustered-average	91.4%	92.4%	90.1%	91.2%

Table 4.21: Evaluation of user representation on the DBLP dataset

Method	Accuracy	Precision	Recall	F1 score
Average	87.5%	87.3%	88.1%	87.7%
Attention	84.0%	81.5%	88.3%	84.8%
Clustered-average	87.0%	86.9%	87.2%	87.1%

doi:10.6342/NTU202501568

Chapter 5: Conclusion and Future Work

5.1 Conclusion

In this research, the proposed Similarity-Based models exhibit competitive performance, reduce execution time by at least 23%, and offer more interpretable feature attribution owing to their simplified architecture. The results indicate that even using basic encoding methods such as TF-IDF or LDA yields strong predictive outcomes, highlighting the effectiveness of content-based similarity approaches for retweet prediction. Additionally, the models' streamlined structure enables more accessible weight analysis of input relationships. This analysis reveals that relationships involving the author's posting history are more prominent in the Twitter dataset, whereas the connection between the target tweet and the user's retweet history is more influential in the DBLP dataset—findings that align with those from the ablation study. The most influential input relationships differ between datasets due to their inherent characteristics. On Twitter, users' retweet behavior tends to be influenced more by their perception of the author than by the content of an individual tweet. In contrast, citations in the DBLP dataset are generally driven by content relevance rather than author familiarity.

This research presents one of the earliest applications of LLMs to retweet prediction and demonstrates the effectiveness of a novel framework that bridges the gap between deep learning models and LLMs through weight analysis of Similarity-Based models and the self-explanatory capabilities of LLMs. Multiple variants of prompt instructions were designed to guide the LLM's attention toward influential input feature relationships. The results exhibit strong alignment with weight analysis findings, with a notable 10% improvement in F1 score between Instruction A and Instruction A.5 on the Twitter dataset. These findings highlight the potential of leveraging insights from traditional deep learning models to inform and optimize prompt engineering strategies for LLMs.

doi:10.6342/NTU202501568

5.2 Future Work

1. Although the proposed framework demonstrates effectiveness, a performance gap remains between the LLM-Based and Similarity-Based models. Furthermore, the inconsistency between the results of the LLM-Based models and the weight analysis highlights certain limitations. While conceptually similar to AMPLIFY [15], our approach applies uniform emphasis on a specific input feature relationship for each prompt. In contrast, AMPLIFY dynamically identifies instance-specific keywords prior to inference and integrates them into in-context learning prompts. Enhancing our framework to incorporate such dynamic prompt generation could better account for the variability in user behavior, where different subsets of input features may influence retweet decisions across users within the same dataset. Future work could explore replacing static weight analysis with a trained classifier that predicts the most influential input relationship for each user–retweet pair, thereby enabling dynamic and personalized interpretability.

2. The Similarity-Based models achieve competitive performance despite excluding social network and temporal features, both of which are known to be important for understanding retweet behavior. Future work can incorporate these features to gain deeper insights into their influence on retweet prediction. Additionally, the capability of LLMs to process and leverage social network and temporal information should be further explored.

3. The user preferences generated by the LLM-Modular model are primarily designed to enhance human interpretability; however, LLMs may not inherently represent users in such a structured format. Further research is needed to better understand how LLMs internally model user preferences and to evaluate the effectiveness of the additional summarization module in the context of the retweet prediction task.

References

[1] R. Ma, X. Hu, Q. Zhang, X. Huang, and Y.-G. Jiang, "Hot Topic-Aware Retweet Prediction with Masked Self-attentive Model," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Paris France: ACM, Jul. 2019, pp. 525–534. doi: 10.1145/3331184.3331236.

[2] J. Bollen, H. Mao, and A. Pepe, "Modeling Public Mood and Emotion: Twitter Sentiment and Socio-Economic Phenomena," *Proc. Int. AAAI Conf. Web Soc. Media*, vol. 5, no. 1, pp. 450–453, Aug. 2021, doi: 10.1609/icwsm.v5i1.14171.

[3] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *J. Comput. Sci.*, vol. 2, no. 1, pp. 1–8, Mar. 2011, doi: 10.1016/j.jocs.2010.12.007.

[4] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: real-time event detection by social sensors," in *Proceedings of the 19th international conference on World wide web*, Raleigh North Carolina USA: ACM, Apr. 2010, pp. 851–860. doi: 10.1145/1772690.1772777.

[5] S. N. Firdaus, C. Ding, and A. Sadeghian, "Retweet Prediction based on Topic, Emotion and Personality," *Online Soc. Netw. Media*, vol. 25, p. 100165, Sep. 2021, doi: 10.1016/j.osnem.2021.100165.

[6] S. N. Firdaus, C. Ding, and A. Sadeghian, "Retweet: A popular information diffusion mechanism – A survey paper," *Online Soc. Netw. Media*, vol. 6, pp. 26–40, Jun. 2018, doi: 10.1016/j.osnem.2018.04.001.

[7] F. Zhou, X. Xu, G. Trajcevski, and K. Zhang, "A Survey of Information Cascade Analysis: Models, Predictions, and Recent Advances," *ACM Comput. Surv.*,

doi:10.6342/NTU202501568

vol. 54, no. 2, pp. 1–36, Mar. 2022, doi: 10.1145/3433000.

[8] Q. Zhang, Y. Gong, J. Wu, H. Huang, and X. Huang, “Retweet Prediction with Attention-based Deep Neural Network,” in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, Indianapolis Indiana USA: ACM, Oct. 2016, pp. 75–84. doi: 10.1145/2983323.2983809.

[9] L. Wang, Y. Zhang, J. Yuan, K. Hu, and S. Cao, “FEBDNN: fusion embedding-based deep neural network for user retweeting behavior prediction on social networks,” *Neural Comput. Appl.*, vol. 34, no. 16, pp. 13219–13235, Aug. 2022, doi: 10.1007/s00521-022-07174-9.

[10] A. K. Kushwaha, A. K. Kar, and P. Vigneswara Ilavarasan, “Predicting Information Diffusion on Twitter a Deep Learning Neural Network Model Using Custom Weighted Word Features,” in *Responsible Design, Implementation and Use of Information and Communication Technology*, vol. 12066, M. Hattingh, M. Matthee, H. Smuts, I. Pappas, Y. K. Dwivedi, and M. Mäntymäki, Eds., in Lecture Notes in Computer Science, vol. 12066. , Cham: Springer International Publishing, 2020, pp. 456–468. doi: 10.1007/978-3-030-44999-5_38.

[11] Z. He *et al.*, “Large Language Models as Zero-Shot Conversational Recommenders,” in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, Birmingham United Kingdom: ACM, Oct. 2023, pp. 720–730. doi: 10.1145/3583780.3614949.

[12] H. Lyu *et al.*, “LLM-Rec: Personalized Recommendation via Prompting Large Language Models,” 2023, *arXiv*. doi: 10.48550/ARXIV.2307.15780.

[13] X. Liu *et al.*, “GPT understands, too,” *AI Open*, vol. 5, pp. 208–215, 2024, doi:

10.1016/j.aiopen.2023.08.012.

[14] L. Li, Y. Zhang, and L. Chen, “Prompt Distillation for Efficient LLM-based Recommendation,” in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, Birmingham United Kingdom: ACM, Oct. 2023, pp. 1348–1357. doi: 10.1145/3583780.3615017.

[15] S. Krishna, J. Ma, D. Slack, A. Ghandeharioun, S. Singh, and H. Lakkaraju, “Post Hoc Explanations of Language Models Can Improve Language Models,” Dec. 07, 2023, *arXiv*: arXiv:2305.11426. doi: 10.48550/arXiv.2305.11426.

[16] M. Jenders, G. Kasneci, and F. Naumann, “Analyzing and predicting viral tweets,” in *Proceedings of the 22nd International Conference on World Wide Web*, Rio de Janeiro Brazil: ACM, May 2013, pp. 657–664. doi: 10.1145/2487788.2488017.

[17] H. Huang, Q. Zhang, J. Wu, and X. Huang, “Predicting Which Topics You Will Join in the Future on Social Media,” in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Shinjuku Tokyo Japan: ACM, Aug. 2017, pp. 733–742. doi: 10.1145/3077136.3080791.

[18] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, “ArnetMiner: extraction and mining of academic social networks,” in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, Las Vegas Nevada USA: ACM, Aug. 2008, pp. 990–998. doi: 10.1145/1401890.1402008.

[19] Q. Zhang, Y. Gong, Y. Guo, and X. Huang, “Retweet Behavior Prediction Using Hierarchical Dirichlet Process,” *Proc. AAAI Conf. Artif. Intell.*, vol. 29, no. 1, Feb. 2015, doi: 10.1609/aaai.v29i1.9152.

[20] H.-K. Peng, J. Zhu, D. Piao, R. Yan, and Y. Zhang, “Retweet Modeling Using

Conditional Random Fields," in *2011 IEEE 11th International Conference on Data Mining Workshops*, Vancouver, BC, Canada: IEEE, Dec. 2011, pp. 336–343. doi: 10.1109/ICDMW.2011.146.

[21] Z. Luo, M. Osborne, J. Tang, and T. Wang, "Who will retweet me?: finding retweeters in twitter," in *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, Dublin Ireland: ACM, Jul. 2013, pp. 869–872. doi: 10.1145/2484028.2484158.

[22] B. Jiang, J. Liang, Y. Sha, and L. Wang, "Message Clustering based Matrix Factorization Model for Retweeting Behavior Prediction," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, Melbourne Australia: ACM, Oct. 2015, pp. 1843–1846. doi: 10.1145/2806416.2806650.

[23] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu, "Collaborative personalized tweet recommendation," in *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, Portland Oregon USA: ACM, Aug. 2012, pp. 661–670. doi: 10.1145/2348283.2348372.

[24] Z. Zhao *et al.*, "Recommender Systems in the Era of Large Language Models (LLMs)," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 11, pp. 6889–6907, Nov. 2024, doi: 10.1109/TKDE.2024.3392335.

[25] X. Tang, Q. Miao, Y. Quan, J. Tang, and K. Deng, "Predicting individual retweet behavior by user similarity: A multi-task learning approach," *Knowl.-Based Syst.*, vol. 89, pp. 681–688, Nov. 2015, doi: 10.1016/j.knosys.2015.09.008.

[26] W.-C. Kang *et al.*, "Do LLMs Understand User Preferences? Evaluating LLMs On User Rating Prediction," 2023, *arXiv*. doi: 10.48550/ARXIV.2305.06474.

[27] J. Liu, C. Liu, P. Zhou, R. Lv, K. Zhou, and Y. Zhang, “Is ChatGPT a Good Recommender? A Preliminary Study,” Oct. 27, 2023, *arXiv*: arXiv:2304.10149. doi: 10.48550/arXiv.2304.10149.

[28] S. Huang, S. Mamidanna, S. Jangam, Y. Zhou, and L. H. Gilpin, “Can Large Language Models Explain Themselves? A Study of LLM-Generated Self-Explanations,” 2023, *arXiv*. doi: 10.48550/ARXIV.2310.11207.

[29] F. Huang, H. Kwak, K. Park, and J. An, “ChatGPT Rates Natural Language Explanation Quality Like Humans: But on Which Scales?,” 2024, *arXiv*. doi: 10.48550/ARXIV.2403.17368.

[30] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” 2013, *arXiv*. doi: 10.48550/ARXIV.1301.3781.

[31] J. Pennington, R. Socher, and C. Manning, “Glove: Global Vectors for Word Representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1532–1543. doi: 10.3115/v1/D14-1162.

[32] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers, “MTEB: Massive Text Embedding Benchmark,” 2022, *arXiv*. doi: 10.48550/ARXIV.2210.07316.

[33] Z. Qiu, H. Lyu, W. Xiong, and J. Luo, “Can LLMs Simulate Social Media Engagement? A Study on Action-Guided Response Generation,” Feb. 17, 2025, *arXiv*: arXiv:2502.12073. doi: 10.48550/arXiv.2502.12073.

[34] E. Yu, J. Li, and C. Xu, “PopALM: Popularity-Aligned Language Models for Social Media Trendy Response Prediction,” Feb. 29, 2024, *arXiv*: arXiv:2402.18950. doi: 10.48550/arXiv.2402.18950.

[35] O. Khattab *et al.*, “DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines,” 2023, *arXiv*. doi: 10.48550/ARXIV.2310.03714.

[36] C. Li, J. Ma, X. Guo, and Q. Mei, “DeepCas: An End-to-end Predictor of Information Cascades,” in *Proceedings of the 26th International Conference on World Wide Web*, Perth Australia: International World Wide Web Conferences Steering Committee, Apr. 2017, pp. 577–586. doi: 10.1145/3038912.3052643.

[37] S. Ji *et al.*, “Community-based Dynamic Graph Learning for Popularity Prediction,” in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Long Beach CA USA: ACM, Aug. 2023, pp. 930–940. doi: 10.1145/3580305.3599281.

[38] H. Wang, C. Yang, and C. Shi, “Neural Information Diffusion Prediction with Topic-Aware Attention Network,” in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, Virtual Event Queensland Australia: ACM, Oct. 2021, pp. 1899–1908. doi: 10.1145/3459637.3482374.

Appendix

Appendix A: Prompt Instruction for Retweet Prediction

You are a social media analyst specializing in retweet prediction. Your primary task is to predict retweet based on given information. You should jointly consider the semantic relationships among all the given inputs.

Consider the following:

*1. **Semantic Similarity Analysis:** Carefully compare the content — including topics, concepts, and viewpoints — of the related items, and identify meaningful overlaps or connections.*

*2. **Content Pattern Recognition:** Analyze whether the content of the related items is aligned, with particular attention to recurring themes, perspectives, or subject matter.*

Your assessment should categorize the relationship as either:

- Semantically related (likely to be retweeted)

- Not semantically related (unlikely to be retweeted)

Provide a brief explanation highlighting which semantic relationship was most influential in your decision.

A.1 Prompt Instruction Targeting x_1

You are a social media analyst specializing in retweet prediction. Your ONLY task is to determine the semantic relationship between a target tweet and a specific user's past tweet history to predict retweet likelihood.

IMPORTANT: Focus EXCLUSIVELY on the relationship between the target tweet and the user's tweet history. Other data points are irrelevant to your analysis.

Consider the following:

*1. **Semantic Similarity Analysis:** Carefully compare the content — including topics, concepts, and viewpoints — of the related items, and identify meaningful overlaps or connections.*

*2. **Content Pattern Recognition:** Analyze whether the content of the related items is aligned, with particular attention to recurring themes, perspectives, or subject matter.*

Your assessment should categorize the relationship as either:

- Semantically related (likely to be retweeted)

- Not semantically related (unlikely to be retweeted)

Provide a brief explanation highlighting ONLY the specific aspects of the target tweet that align or conflict with patterns in the user's tweet history. Explicitly state which semantic relationship was most influential in your decision.

Remember: The ONLY relationship that matters is between the target tweet and the user's tweet history. Disregard any other information that does not directly inform this specific relationship.

doi:10.6342/NTU202501568

A.2 Prompt Instruction Targeting x_2

You are a social media analyst specializing in retweet prediction. Your ONLY task is to determine the semantic relationship between a target tweet and a specific user's past retweet history to predict retweet likelihood.

IMPORTANT: Focus EXCLUSIVELY on the relationship between the target tweet and the user's retweet history. Other data points are irrelevant to your analysis.

Consider the following:

1. **Semantic Similarity Analysis:** Carefully compare the content — including topics, concepts, and viewpoints — of the related items, and identify meaningful overlaps or connections.

2. **Content Pattern Recognition:** Analyze whether the content of the related items is aligned, with particular attention to recurring themes, perspectives, or subject matter.

Your assessment should categorize the relationship as either:

- Semantically related (likely to be retweeted)

- Not semantically related (unlikely to be retweeted)

Provide a brief explanation highlighting ONLY the specific aspects of the target tweet that align or conflict with patterns in the user's retweet history. Explicitly state which semantic relationship was most influential in your decision.

Remember: The ONLY relationship that matters is between the target tweet and the user's retweet history. Disregard any other information that does not directly inform this specific relationship.

A.3 Prompt Instruction Targeting x_3

You are a social media analyst specializing in retweet prediction. Your ONLY task is to determine the semantic relationship between a target tweet and its author's past tweet history to predict retweet likelihood.

IMPORTANT: Focus EXCLUSIVELY on the relationship between the target tweet and its author's tweet history. Other data points are irrelevant to your analysis.

Consider the following:

1. **Semantic Similarity Analysis:** Carefully compare the content — including topics, concepts, and viewpoints — of the related items, and identify meaningful overlaps or connections.

2. **Content Pattern Recognition:** Analyze whether the content of the related items is aligned, with particular attention to recurring themes, perspectives, or subject matter.

Your assessment should categorize the relationship as either:

- Semantically related (likely to be retweeted)

- Not semantically related (unlikely to be retweeted)

Provide a brief explanation highlighting ONLY the specific aspects of the target tweet that align or conflict with patterns in its author's retweet history. Explicitly state which semantic relationship was most influential in your decision.

Remember: The ONLY relationship that matters is between the target tweet and its author's tweet history. Disregard any other information that does not directly inform this specific relationship.

A.4 Prompt Instruction Targeting x_4

You are a social media analyst specializing in retweet prediction. Your ONLY task is to determine the semantic relationship between a user’s tweet history and the user's retweet history to predict retweet likelihood.

IMPORTANT: Focus EXCLUSIVELY on the relationship between the user’s tweet history and the user's retweet history. Other data points are irrelevant to your analysis.

Consider the following:

*1. **Semantic Similarity Analysis:** Carefully compare the content — including topics, concepts, and viewpoints — of the related items, and identify meaningful overlaps or connections.*

*2. **Content Pattern Recognition:** Analyze whether the content of the related items is aligned, with particular attention to recurring themes, perspectives, or subject matter.*

Your assessment should categorize the relationship as either:

- Semantically related (likely to be retweeted)

- Not semantically related (unlikely to be retweeted)

Provide a brief explanation highlighting ONLY the specific aspects of the user’s tweet history that align or conflict with patterns in the user's retweet history. Explicitly state which semantic relationship was most influential in your decision.

Remember: The ONLY relationship that matters is between the user’s tweet history and the user's retweet history. Disregard any other information that does not directly inform this specific relationship.

doi:10.6342/NTU202501568

A.5 Prompt Instruction Targeting x_5

You are a social media analyst specializing in retweet prediction. Your ONLY task is to determine the semantic relationship between a user’s tweet history and an author's past tweet history to predict retweet likelihood.

IMPORTANT: Focus EXCLUSIVELY on the relationship between the user’s tweet history and the author's past tweet history. Other data points are irrelevant to your analysis.

Consider the following:

1. **Semantic Similarity Analysis:** Carefully compare the content — including topics, concepts, and viewpoints — of the related items, and identify meaningful overlaps or connections.

2. **Content Pattern Recognition:** Analyze whether the content of the related items is aligned, with particular attention to recurring themes, perspectives, or subject matter.

Your assessment should categorize the relationship as either:

- Semantically related (likely to be retweeted)

- Not semantically related (unlikely to be retweeted)

Provide a brief explanation highlighting ONLY the specific aspects of the user’s tweet history that align or conflict with patterns in the author's tweet history. Explicitly state which semantic relationship was most influential in your decision.

Remember: The ONLY relationship that matters is between the user’s tweet history and the author's past tweet history. Disregard any other information that does not directly inform this specific relationship.

doi:10.6342/NTU202501568

A.6 Prompt Instruction Targeting x_6

You are a social media analyst specializing in retweet prediction. Your ONLY task is to determine the semantic relationship between a user’s retweet history and an author's past tweet history to predict retweet likelihood.

IMPORTANT: Focus EXCLUSIVELY on the relationship between the user’s retweet history and the author's past tweet history. Other data points are irrelevant to your analysis.

Consider the following:

1. **Semantic Similarity Analysis:** Carefully compare the content — including topics, concepts, and viewpoints — of the related items, and identify meaningful overlaps or connections.

2. **Content Pattern Recognition:** Analyze whether the content of the related items is aligned, with particular attention to recurring themes, perspectives, or subject matter.

Your assessment should categorize the relationship as either:

- *Semantically related (likely to be retweeted)*

- *Not semantically related (unlikely to be retweeted)*

Provide a brief explanation highlighting ONLY the specific aspects of the user’s retweet history that align or conflict with patterns in the author's tweet history. Explicitly state which semantic relationship was most influential in your decision.

Remember: The ONLY relationship that matters is between the user’s retweet history and the author's past tweet history. Disregard any other information that does not directly inform this specific relationship.

Appendix B: Prompt Instruction for Summarization

As a specialized text analysis engine, your task is to perform a detailed extraction and categorization of the following elements from the provided texts:

1. MAIN TOPICS: Identify the primary discussion subjects across all texts, sorted by frequency/importance.

- Format as a bulleted list with brief descriptions

- Include specific examples from the texts for each topic

2. KEY ENTITIES:

- People: All individuals mentioned by name or username (with @ symbol)

- Organizations: All companies, platforms, products, and groups

- Locations: All geographical places mentioned

- Events: Conferences, streams, masterclasses with their details when available

- Technical terms: Programming languages, frameworks, functions, and technical concepts

3. RECURRING THEMES:

- Identify underlying patterns or motifs that appear across multiple texts

- Support each theme with 2-3 specific text examples

Format your response as a structured report with clear section headings and concise bullet points. Prioritize extracting specific, actionable information rather than general observations.

doi:10.6342/NTU202501568