

Article

Graph-Conditioned Stochastic Modeling of Twitter Information Cascades with Dual-Head Transformers for Early Virality Prediction

Bowen Dong ^{1,*}, Xinyu Zhang ^{2,†}, Chaoyan Yan ³, Weiyan Zhu ⁴, Lingmin Hou ² and Yifan Feng ⁵

¹ School of Electrical Automation and Information Engineering, Tianjin University, Tianjin 300072, China

² Department of Computer Science, Rochester Institute of Technology, Rochester, NY 14623, USA

³ Department of Computer Science, Rutgers University, New Brunswick, NJ 08901, USA

⁴ Meta Platforms Inc., Menlo Park, CA 94025, USA

⁵ Department of Computer Science and Engineering, Santa Clara University, Santa Clara, CA 95053, USA

* Correspondence: dbowe021@fiu.edu

† These authors contributed equally to this work.

Abstract

Information cascades in online social networks arise from stochastic interactions among user behavior, temporal activation, and graph-structured exposure. Early prediction of cascade outcomes remains difficult because only a short diffusion prefix is observable, while future propagation depends on sparse user-level transitions across a heterogeneous social network. This study develops a graph-conditioned stochastic modeling framework for early Twitter cascade prediction. Retweet cascades are formulated as history-dependent stochastic processes over a finite user vocabulary, and a causal dual-head Transformer is used to infer cascade virality and logarithmic final size from short observed prefixes. To incorporate social-network structure, user embeddings pretrained from the follow graph are introduced as external structural priors. A controlled ablation design separates the effects of random embeddings, graph-pretrained embeddings, frozen structural priors, and handcrafted feature fusion. Experiments on Higgs Twitter retweet cascades show that direct full-vocabulary next-user prediction is statistically fragile under sparse short-prefix observations, motivating macro-level cascade outcome prediction. Among the evaluated configurations, the frozen graph-pretrained Transformer achieves the strongest overall balance, reaching an AUC of 0.819, a Brier score of 0.151, and an RMSE of 0.192, while the causal Transformer without a graph prior already surpasses logistic regression and approaches Random Forest; however, gains over competitive baselines are modest and statistically significant only in selected pairwise comparisons. Calibration analysis, bootstrap confidence intervals, and paired statistical tests confirm that graph-derived user priors provide more reliable improvements than sequence modeling alone under short-prefix sparse observations. These findings indicate that graph-conditioned structural priors offer a promising complement to causal sequence modeling for early Twitter cascade prediction.



Academic Editor: Xiangjie Kong

Received: 17 May 2026

Revised: 7 June 2026

Accepted: 10 June 2026

Published: 27 June 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: stochastic process; information cascade; graph-conditioned inference; social network diffusion; causal Transformer; bootstrap confidence interval

MSC: 60G99; 91D30; 68T09; 62M10

1. Introduction

1.1. Research Background and Motivation

Information diffusion in online social networks is governed by a complex interaction among user behavior, temporal activation, and network topology [1]. A retweet cascade may remain confined to a small local neighborhood, or it may rapidly expand into a large-scale diffusion process through successive user interactions [2]. Cascades are shaped by heterogeneous user influence, irregular temporal triggering, and the structural accessibility provided by the underlying social graph. As a result, information cascades provide a representative setting for studying stochastic processes on large-scale social networks.

Early prediction of cascade outcomes has practical and methodological significance. In applications such as public-event monitoring, misinformation tracking, emergency communication, and online marketing, decisions often need to be made when only a short initial diffusion trace is available [3]. The central difficulty is that the early prefix of a cascade contains only partial evidence about the eventual diffusion scale. A small number of retweets may either represent a transient local response or the beginning of a broader propagation trajectory [4]. Distinguishing these two regimes requires a model that can integrate short-term temporal information with prior knowledge about the social positions of participating users.

Existing cascade-prediction studies have demonstrated the value of temporal, structural, and user-level features. However, many existing approaches remain primarily empirical: they either rely on handcrafted descriptors or adopt deep sequence models without explicitly formalizing the stochastic nature of cascade evolution. This creates a gap between predictive modeling and mathematical interpretation. In particular, the role of the social graph is often entangled with the role of sequential representation learning, making it difficult to determine whether performance gains arise from temporal modeling capacity, graph-derived structural information, or their interaction [5].

This study addresses this gap by formulating Twitter information cascades as history-dependent stochastic processes over a finite user vocabulary. Within this formulation, a causal Transformer is used not merely as a generic sequence encoder, but as a parameterized representation of cascade histories under a non-anticipatory information constraint. To incorporate social-network structure, user embeddings pretrained from the follow graph are introduced as external graph priors. The resulting framework allows early cascade prediction to be studied as a graph-conditioned stochastic inference problem rather than as a conventional sequence-classification task. The stochastic formulation serves as interpretive scaffolding connecting the neural computation to mathematical cascade theory; the operational training objective focuses on macro-level virality and size prediction rather than direct optimization of the full user-level transition kernel.

1.2. Research Problem and Challenges

The research problem considered in this paper is early-stage cascade outcome prediction. Given a short prefix of a retweet cascade, the objective is to infer whether the cascade will reach a predefined virality threshold and to estimate its eventual scale. This formulation emphasizes macro-level diffusion outcomes rather than exact user-level transition prediction, which becomes statistically fragile in large and sparse social-network environments.

A central difficulty arises from the limited information contained in short cascade prefixes. Early prediction must be performed before the diffusion process has fully revealed its trajectory [6]. The observed prefix may contain only a few user activations and sparse timing information, while the eventual cascade size depends on subsequent interactions that remain unobserved [7]. This setting imposes a severe information constraint on the model and makes naive extrapolation unreliable.

This uncertainty is amplified by the large user space of social-network cascades. In a platform-scale diffusion process, the number of potential retweeting users is extremely large, whereas each individual cascade explores only a tiny portion of the user vocabulary [8]. Directly modeling the next activated user therefore suffers from sparsity, weak repeated observations, and a combinatorial transition space. These properties motivate a shift from fine-grained next-user prediction to more stable macro-level targets such as virality and final size.

Another source of complexity is the heavy-tailed distribution of cascade sizes. Most cascades terminate after limited diffusion, while a small fraction accounts for disproportionately large propagation [9]. This imbalance affects both classification and regression. A model may achieve reasonable ranking performance while still producing poorly calibrated probabilities or unstable size estimates. Reliable cascade modeling therefore requires evaluation beyond point estimates, including uncertainty quantification, calibration assessment, and statistical comparison against competitive baselines.

A further methodological issue concerns the origin of predictive gains in deep cascade models [10]. Transformer architectures are powerful sequence encoders, but their advantage under very short prefixes is not self-evident. If the observed sequence is too short to provide sufficient temporal evidence, the main benefit may instead come from external structural priors. Disentangling these sources requires controlled ablation over random user embeddings, graph-pretrained embeddings, embedding freezing, and feature fusion.

These considerations motivate the main research question of this paper: whether graph-pretrained user representations can provide a robust structural prior for early Twitter cascade prediction, and whether their contribution can be separated from the effects of Transformer sequence modeling and handcrafted cascade descriptors.

1.3. Main Contributions

This paper makes the following contributions.

(1) **Stochastic cascade formulation.** Twitter retweet cascades are formulated as history-dependent stochastic processes over a finite user vocabulary, providing a mathematical basis for early virality and final-size prediction.

(2) **Graph-pretrained Transformer framework.** A causal dual-head Transformer is developed to jointly predict cascade virality and log-final size, with user embeddings pretrained from the social follow graph as structural priors.

(3) **Controlled ablation design.** The study separates the effects of random embeddings, graph-pretrained embeddings, frozen structural priors, and handcrafted feature fusion, enabling a clearer interpretation of predictive gains.

(4) **Statistically reliable evaluation.** Model performance is assessed using ranking metrics, regression error, calibration measures, bootstrap confidence intervals, and paired statistical tests.

(5) **Evidence on graph-conditioned prediction.** The results indicate that graph-derived user representations provide more stable gains than sequence modeling alone under short-prefix cascade observations.

1.4. Organization of the Paper

The remainder of this paper is organized as follows.

Section 2 reviews related work on information cascade prediction, stochastic diffusion modeling, graph representation learning, and Transformer-based modeling, followed by a summary of the research positioning.

Section 3 presents the stochastic modeling and computational framework, including the mathematical formulation of Twitter cascades, prefix-based data construction,

graph-pretrained dual-head Transformer architecture, baseline models, and statistical evaluation protocol.

Section 4 reports the experimental results, including predictive performance, graph-pretraining effects, feature-fusion analysis, calibration reliability, lead-time sensitivity, and the feasibility of user-level transition prediction.

Section 5 summarizes the main findings and discusses limitations and future research directions.

2. Related Works

2.1. Information Cascade Prediction in Social Networks

Information cascade prediction has been widely studied in social-network analysis, computational social science, and data-driven diffusion modeling. The central objective is to infer future cascade behavior from partial observations of an ongoing diffusion process [11]. Depending on the prediction target, existing studies can be broadly associated with popularity prediction, final-size estimation, outbreak detection, and early virality classification [12]. These formulations share a common assumption: early diffusion traces contain informative signals about the eventual scale and trajectory of information propagation.

A substantial body of work relies on handcrafted temporal, structural, and user-level descriptors. Temporal features such as early growth rate, inter-arrival intervals, and observation-window activity capture the short-term activation pattern of a cascade [13]. Structural features describe the local diffusion topology, including depth, breadth, branching patterns, and user connectivity. User-level features incorporate attributes related to influence, historical activity, or network centrality [14]. These descriptors have shown strong empirical value, particularly when the available prefix is short and the model must rely on compact summaries of early diffusion behavior.

Despite their effectiveness, feature-based approaches depend heavily on manual design and may not fully capture high-order dependencies among users, timestamps, and network positions. Deep learning methods have therefore been introduced to learn cascade representations directly from diffusion paths, temporal sequences, or cascade graphs. Models based on recurrent neural networks, graph neural networks, and attention mechanisms encode more flexible propagation patterns than fixed feature sets [15]. However, improved representational capacity does not automatically resolve the fundamental uncertainty of early cascade prediction. When the observed prefix contains only a few events, the available temporal evidence may be too limited for complex sequence models to dominate simpler feature-based baselines.

This issue is especially relevant for Twitter-scale cascades, where the number of potential participating users is large and the observed diffusion path is usually sparse. Large-scale event-centered datasets, such as the Higgs Twitter dataset recording social-media activity surrounding the Higgs boson announcement [16], have served as standard benchmarks for studying cascade prediction under realistic sparsity and scale conditions. In such settings, a cascade prefix should not be interpreted only as a short event sequence. It also reflects the structural positions of the early participants in the broader social graph. Consequently, early cascade prediction requires a modeling strategy that can combine local temporal evidence with graph-level prior information.

2.2. Stochastic Diffusion Modeling and Point Processes

Stochastic diffusion models provide a principled way to describe information propagation under uncertainty. In social networks, cascade evolution can be interpreted as a random process in which each observed event modifies the probability of subsequent activations [17]. Markov models, branching processes, epidemic models, and self-exciting

point processes have all been used to represent different aspects of social diffusion [18]. These models offer interpretability by explicitly defining transition probabilities, event intensities, or reproduction mechanisms.

Among them, Hawkes processes have been particularly influential for modeling temporally clustered events. A Hawkes process assumes that previous events increase the intensity of future events through a self-excitation mechanism [19]. Intuitively, each event raises the probability of similar events occurring in the near future—analogue to how a single retweet increases a post’s visibility and thereby triggers a cluster of further retweets before the effect gradually decays. This structure is naturally aligned with retweet cascades, where each retweet may increase the visibility of the original message and thereby stimulate further responses. Extensions of Hawkes processes have been used to model popularity growth, event arrival rates, and temporal decay in information diffusion [20]. Their advantage lies in the explicit representation of triggering dynamics and the interpretability of intensity functions.

Beyond standard self-excitation models, recent work has demonstrated that community clustering structure can give rise to recurring multi-peak cascade dynamics that are not captured by a single excitation kernel [21]. Such temporally heterogeneous diffusion patterns motivate graph-informed modeling approaches that can account for community-driven dynamics underlying cascade evolution. Moreover, real social networks typically involve multiple co-existing relation types—such as retweets, replies, mentions, and follows—and multilayer network frameworks that model these heterogeneous interaction layers jointly have been shown to capture richer structural dynamics than single-layer representations [22].

Nevertheless, classical stochastic diffusion models face limitations when applied to large-scale social networks. Markov transition models become difficult to estimate when the state space consists of hundreds of thousands of users [23]. Point-process models can capture temporal excitation but often require simplifying assumptions about user marks, network exposure, or diffusion topology. Branching and epidemic models provide useful macroscopic descriptions, yet they may obscure the heterogeneity of individual users and the structural dependence induced by the social graph.

These limitations suggest that stochastic cascade modeling requires a balance between mathematical structure and computational flexibility. A purely parametric point-process model may be too restrictive for high-dimensional user-level diffusion, whereas a purely neural sequence model may lack interpretability and statistical discipline. A natural direction is therefore to preserve the stochastic-process view of cascade evolution while using neural architectures to parameterize complex history-dependent representations under causal constraints.

2.3. Graph Representation Learning and Transformer-Based Modeling

Graph representation learning offers a complementary perspective on social cascade prediction. Rather than relying only on observed diffusion prefixes, graph embedding methods learn low-dimensional representations of users from the underlying social network [24]. Random-walk-based methods such as DeepWalk [25] and node2vec [26] treat graph neighborhoods as contextual co-occurrence patterns and learn embeddings through skip-gram-style objectives [27]. Intuitively, these methods generate many short random walks starting from each node; users who frequently co-appear in the same walks are assigned similar low-dimensional representations, so that structural proximity in the network is preserved in the embedding space. These embeddings can encode proximity, community structure, and local connectivity patterns that are not directly visible in a short cascade prefix.

For information diffusion tasks, graph-derived user representations can serve as structural priors. A user appearing early in a cascade may have different predictive significance depending on their network position [28]. Retweets by structurally central users, bridge users, or users embedded in dense communities may imply different downstream diffusion potentials. Graph pretraining can therefore reduce the burden on the sequence model by supplying prior information about user-level diffusion capacity before task-specific training begins.

Transformer architectures provide another important modeling component. Through self-attention, Transformers can represent dependencies among events without imposing a fixed recurrence structure [29]. Conceptually, self-attention allows each event in a sequence to selectively gather information from all other events simultaneously, weighted by learned relevance scores; the causal variant restricts this attention so that each position can only refer to earlier positions, preventing the model from using future information when making predictions. In cascade modeling, causal attention is especially relevant because it ensures that predictions depend only on observed history. This non-anticipatory property is consistent with the temporal nature of early cascade prediction. Attention weights can also be interpreted as dynamic relevance scores over previous events, which offers a bridge between neural sequence modeling and stochastic history-dependent kernels.

However, the use of Transformers in cascade prediction requires caution. Their capacity can easily exceed the amount of information available in short prefixes, particularly when the training set is limited and the user vocabulary is large [30]. Without external structural priors, a Transformer may learn unstable user embeddings or rely on shallow temporal cues already captured by handcrafted features. This concern motivates a graph-pretrained Transformer design, in which the social graph supplies user-level structural information while the causal Transformer models the observed cascade history.

The distinction between trainable and frozen graph embeddings is also methodologically important. Fine-tuning pretrained embeddings may improve adaptation when sufficient task data are available, but it can also destroy useful structural information under sparse supervision. Freezing graph-pretrained embeddings can act as a regularization mechanism, preserving the topology-derived prior and preventing overfitting to a small number of observed cascade prefixes. This issue is central to understanding whether graph pretraining genuinely improves cascade prediction or merely provides another initialization scheme.

2.4. Summary and Research Positioning

Existing studies provide several important foundations for modeling information cascades. Cascade-prediction research has established the predictive value of early temporal and structural signals. Stochastic diffusion models have offered interpretable descriptions of event triggering and propagation uncertainty. Graph representation learning has provided scalable tools for encoding social-network structure. Transformer-based architectures have further expanded the capacity to model history-dependent event sequences.

At the same time, these research streams leave several issues unresolved. Feature-based cascade models are often effective but limited by manual design. Classical stochastic processes provide interpretability but can struggle with large user vocabularies and complex graph dependence. Deep sequence models offer flexible representation learning, yet their gains under short-prefix observations may be difficult to distinguish from the effects of graph-derived priors or engineered descriptors. Many studies also emphasize predictive scores while giving less attention to probability calibration, uncertainty quantification, and statistical comparison.

The present study is positioned at the intersection of stochastic cascade modeling, graph representation learning, and causal Transformer computation. It treats Twitter retweet cascades as history-dependent stochastic processes, introduces graph-pretrained user embeddings as external structural priors, and evaluates early cascade outcomes through a dual-head predictive formulation. The emphasis is not only on improving prediction accuracy, but also on clarifying the source of predictive improvement through controlled ablation and statistically reliable evaluation. This positioning allows the study to connect mathematical modeling of diffusion processes with practical computation on large-scale social-network data.

Table 1 compares the proposed work with representative studies on information cascade prediction and social information diffusion. Existing studies cover probabilistic diffusion modeling, continuous-time cascade dynamics, graph-based propagation representation, attention-based learning, semantic-aware prediction, and user-behavior modeling. The present study differs by treating Twitter diffusion as a stochastic reaction-sequence modeling problem. It jointly characterizes cascade-growth mechanisms, temporal arrival behavior, structural descriptors, token-level auxiliary signals, and observable user-response patterns, thereby linking early cascade traces with interpretable final diffusion outcomes.

Table 1. Comparative positioning of representative studies on information cascade prediction and the proposed work. Bold font indicates the proposed approach of the present study.

Study Category	Diffusion/ Cascade Mechanism	Temporal Dynamics	Structural Representation	Semantic/ Topic Information	User Behavior Modeling
Probabilistic and stochastic diffusion modeling [1,3,11]	✓	partial	partial	–	partial
Inter-/intra-cascade and continuous-time dynamics [2,4]	✓	✓	partial	–	partial
Graph neural network-based cascade prediction [6,23,24,30]	partial	partial	✓	–	partial
Transformer and attention-based cascade modeling [8,10,15,20]	partial	✓	✓	partial	partial
Hypergraph and high-order propagation modeling [13]	partial	partial	✓	–	partial
Topic-, stance-, and content-aware diffusion modeling [7,10,15]	partial	partial	partial	✓	partial
User activity and behavioral transition modeling [5,17,28]	partial	✓	partial	–	✓
Dynamic social network and structural evolution studies [12,18,29]	partial	partial	✓	partial	partial
Survey and adjacent predictive learning studies [9,19,27]	partial	partial	partial	partial	partial
This work	Stochastic cascade growth from reaction traces.	Inter-arrival, length evolution, and position- wise loss.	Root influence, outdegree growth, sequence length, and reaction- chain structure.	Token-level auxiliary cascade signals.	Reaction timing, sequence position, and participa- tion patterns.

3. Materials and Methods

3.1. Mathematical Formulation of Twitter Information Cascades

Twitter information cascades are modeled as stochastic diffusion processes on a large directed user graph. The formulation developed in this section formalizes three levels of the problem: the observation space and stochastic process formulation, the parametric transition kernel, and the macro-level prediction objectives.

3.1.1. Cascade State Space and Historical Process

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a directed social graph, where $\mathcal{V}$ denotes the finite user set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes directed follow relations. A cascade is observed as a marked temporal point sequence,

$$\mathcal{C} = \{(X_i, T_i)\}_{i=0}^{N-1}, \quad X_i \in \mathcal{V}, \quad 0 = T_0 \leq T_1 \leq \dots \leq T_{N-1}$$

where X_i is the user activated at the i -th event, T_i is the corresponding event time after normalization, and $N = N(\mathcal{C})$ is the final cascade size. Here, user identity acts as the event mark, while event time records the timing of activations.

The information available after the k -th observed activation is represented by the natural filtration

$$\mathcal{F}_k = \sigma(X_0, T_0, \dots, X_k, T_k), \quad 0 \leq k < N$$

where $\sigma(\cdot)$ denotes the sigma-algebra generated by the observed cascade prefix. Any valid early prediction rule at step k must be $\mathcal{F}_k$ -measurable; otherwise, it would depend on future cascade events and violate the non-anticipatory constraint.

The most general user-level transition law can then be written as a regular conditional probability,

$$\Pi_k(u) = \mathbb{P}(X_{k+1} = u \mid \mathcal{F}_k), \quad u \in \mathcal{V}$$

Here, Π_k is a random probability measure on $\mathcal{V}$. It changes with the observed history and therefore captures the history-dependent nature of cascade diffusion. A finite-order Markov model is obtained only when

$$\mathbb{P}(X_{k+1} = u \mid \mathcal{F}_k) = \mathbb{P}(X_{k+1} = u \mid X_{k-r+1:k}, T_{k-r+1:k})$$

for some fixed memory length r . The present formulation does not impose this restriction. The entire observed prefix may influence future diffusion, which is more appropriate for social cascades where each activation can create long-range dependencies.

Temporal irregularity is encoded through inter-arrival intervals,

$$\Delta_i = T_i - T_{i-1}, \quad i = 1, \dots, N-1$$

The pair (X_i, Δ_i) forms a marked event representation that records both who was activated and how quickly the activation occurred. Rapid early activations and long inactive gaps correspond to different diffusion regimes.

For early prediction, only a prefix of length K is available. The corresponding observable history is $\mathcal{F}_{K-1}$, while all events after K remain unobserved. The learning problem is therefore to approximate functionals of the unobserved future based on the observed prefix.

3.1.2. Causal Transformer Transition Kernel

The conditional law Π_k is not directly estimable in large user spaces because $|\mathcal{V}|$ is high and observed transitions are sparse. A parametric approximation is therefore introduced. Let

$$Z_k = f_\theta(\mathcal{F}_k) \in \mathbb{R}^d$$

denote a d -dimensional causal representation of the observed history, where f_θ is a Transformer parameterized by θ . The notation $f_\theta(\mathcal{F}_k)$ emphasizes that the representation is a measurable function of the available information at step k .

Causality is enforced by requiring

$$Z_k \perp \{(X_j, T_j) : j > k\} \mid \mathcal{F}_k$$

in the operational sense that Z_k is computed exclusively from events indexed by $0, \dots, k$. The causal attention mask implements this restriction by preventing any hidden state at position k from attending to future positions.

The Transformer-induced transition kernel is then defined as

$$\Pi_{\theta,k}(u) = \mathbb{P}_\theta(X_{k+1} = u \mid \mathcal{F}_k) = \frac{\exp(l_{\theta,u}(Z_k))}{\sum_{v \in \mathcal{V}} \exp(l_{\theta,v}(Z_k))}, \quad u \in \mathcal{V}$$

where $l_{\theta,u}(\cdot)$ is the logit assigned to user u . This expression approximates the unknown conditional law Π_k with a neural transition kernel $\Pi_{\theta,k}$. The approximation error may be interpreted through the conditional risk:

$$\mathcal{R}_{\text{trans}}(\theta) = \mathbb{E}[-\log \Pi_{\theta,k}(X_{k+1})]$$

where the expectation is taken over the empirical cascade-generating distribution and admissible observation steps. Minimizing this risk corresponds to fitting the parametric transition kernel to the observed user-level transition data.

Although the transition risk provides the mathematical motivation for the stochastic formulation, direct optimization over the full user vocabulary is fragile under sparse short-prefix observations. The kernel $\Pi_{\theta,k}$ therefore serves as interpretive scaffolding in this study: it connects the Transformer computation to stochastic process theory and motivates the causal masking constraint, but it is not directly optimized as a training objective. The Transformer instead produces

$$Z_{K-1} = f_{\theta}(\mathcal{F}_{K-1})$$

as a sufficient computational summary of the observed cascade prefix. The attention mechanism admits an additional probabilistic interpretation. For a causal self-attention layer, the attention weights satisfy

$$a_{k,j} \geq 0, \quad \sum_{j=0}^k a_{k,j} = 1$$

where $a_{k,j}$ is the normalized attention weight assigned from the current position k to a historical position $j \leq k$. Hence, attention defines a history-dependent probability distribution over past events. This probabilistic interpretation connects the Transformer computation to the stochastic kernel formulation and provides a principled view of how historical cascade events are aggregated into the final prediction.

3.1.3. Virality and Log-Final-Size Learning Objectives

The transition-kernel formulation describes user-level diffusion, but the prediction target of this study is macro-level cascade outcome inference. Let $N = N(\mathcal{C})$ denote the final cascade size. Given a virality threshold τ ,

$$Y_v = \mathbf{1}\{N \geq \tau\}, \quad Y_s = \log N$$

where Y_v is the binary virality label, Y_s is the logarithmic final-size target, and $\mathbf{1}\{\cdot\}$ is the indicator function. The logarithmic transformation is used because cascade sizes are heavy-tailed; it reduces the variance of the regression target and aligns the size prediction with multiplicative growth behavior.

Given only the early information $\mathcal{F}_{K-1}$, the optimal virality predictor under log-loss is the conditional probability

$$p^*(\mathcal{F}_{K-1}) = \mathbb{P}(Y_v = 1 \mid \mathcal{F}_{K-1})$$

whereas the optimal squared-loss predictor for final size is the conditional expectation

$$s^*(\mathcal{F}_{K-1}) = \mathbb{E}[Y_s \mid \mathcal{F}_{K-1}]$$

These two quantities clarify the statistical meaning of the learning task. The model is not attempting to deterministically extrapolate a cascade; it approximates conditional functionals of the unobserved future diffusion given the observable early prefix.

The dual-head Transformer approximates these two conditional functionals through

$$(p_\theta, s_\theta) = q_\theta(Z_{K-1})$$

where q_θ denotes the task-specific prediction mapping, $p_\theta \in [0, 1]$ is the estimated virality probability, and $s_\theta \in \mathbb{R}$ is the estimated log-final size. The two outputs share the same causal cascade representation but use separate linear heads.

The population risk associated with the two tasks is

$$\mathcal{R}(\theta) = \alpha \mathbb{E}[\text{BCE}(Y_v, p_\theta)] + (1 - \alpha) \mathbb{E}[(Y_s - s_\theta)^2]$$

where $\alpha \in [0, 1]$ balances classification and regression. The binary cross-entropy term encourages calibrated estimation of virality probability, while the squared-loss term estimates the conditional mean of the log-final size.

Given a training set of M cascades, the empirical objective becomes

$$\mathcal{R}(\theta) = \frac{1}{M} \sum_{m=1}^M \left[\alpha \cdot \text{BCE}(y_v^{(m)}, p_\theta^{(m)}) + (1 - \alpha) (y_s^{(m)} - s_\theta^{(m)})^2 \right]$$

where m indexes cascades in the training set, and lower-case symbols denote observed labels and predictions. This empirical risk is the optimization criterion used for model training.

This formulation provides a coherent bridge from stochastic diffusion to practical computation. The cascade evolves according to a history-dependent conditional law; the causal Transformer approximates this law through a parametric representation of observable cascade histories; and the dual-head predictor translates the resulting summary into macro-level diffusion outcomes.

Beyond providing a formal vocabulary, the stochastic-process framework contributes three specific structural insights that inform the architecture and evaluation design. First, the filtration $\mathcal{F}_k$ makes the non-anticipatory constraint mathematically precise: any valid early predictor must be $\mathcal{F}_{K-1}$ -measurable, which justifies the causal attention mask as a statistical requirement rather than merely an architectural convention. Second, writing the learning targets as the conditional probability $p^*(\mathcal{F}_{K-1}) = \mathbb{P}(Y_v = 1 \mid \mathcal{F}_{K-1})$ and the conditional expectation $s^*(\mathcal{F}_{K-1}) = \mathbb{E}[Y_s \mid \mathcal{F}_{K-1}]$ clarifies that the prediction problem is the estimation of well-defined conditional functionals of the unobserved cascade future, not a heuristic extrapolation from observed counts. Third, the causal self-attention weights $a_{k,j} \geq 0$ with $\sum_{j=0}^k a_{k,j} = 1$ define a history-dependent probability distribution over past events at each decoding step, connecting the neural computation to the stochastic kernel Π_k and providing a principled interpretation of how cascade history is aggregated into the final prediction. Together, these connections situate the Transformer within a rigorous probabilistic framework for cascade diffusion rather than treating it as an opaque sequence encoder.

3.2. Dataset Construction and Prefix-Based Cascade Modeling

The stochastic formulation above requires an empirical cascade construction procedure that preserves user identity, temporal order, and graph-based structural information. The dataset construction in this section describes how raw Higgs Twitter activity records are converted into a supervised prefix-outcome learning problem.

3.2.1. Higgs Twitter Dataset and Retweet Cascade Construction

The empirical analysis is based on the public Higgs Twitter dataset [16], which records social-media activity surrounding the announcement of the Higgs boson discovery. The dataset contains timestamped user interactions including retweets, replies, and mentions, together with the social follow graph of the involved users.

Each retweet event is interpreted as a directed diffusion relation. If a record is written as

$$(a, b, t, \text{RT})$$

then user a retweeted user b at time t . The direction of information flow is therefore taken as $b \rightarrow a$, because the content associated with user b becomes visible through the activation of user a . This convention assigns information flow from source to recipient, consistent with the directed graph structure of the follow relation.

A root-grouped cascade is then obtained by collecting all users who retweeted the same root user and ordering them by timestamp. For a root user b , the corresponding cascade is written as

$$\mathcal{C}(b) = \{(b, t_0), (a_1, t_1), \dots, (a_{n_b}, t_{n_b})\}, \quad t_0 \leq t_1 \leq \dots \leq t_{n_b}$$

where a_i denotes the i -th user who retweeted b and $n_b + 1$ is the size of the constructed cascade. The root event is included to preserve the source identity of the diffusion process.

The follow graph is not used to define cascade membership. Instead, it serves as an external structural source for user representation learning. This separation is important: retweet cascades describe who activated whom and when, while the follow graph describes structural proximity in the social network. The two sources of information are combined through graph-pretrained user embeddings, as described in Section 3.3.

The resulting cascades exhibit the typical statistical properties of social diffusion: sparse participation, highly uneven cascade lengths, and substantial variation in propagation speed. Most cascades terminate quickly with few participants, while a small fraction propagates broadly. This heavy-tailed structure motivates the use of log-scale targets and stratified evaluation.

3.2.2. Temporal Split and Vocabulary Construction

The predictive task is designed as a chronological early-warning problem. For this reason, the data are split according to event time rather than by random sampling. Let $T_{\min}$ and T_{split} denote the earliest timestamp and the split threshold, respectively:

$$\mathcal{D}_{\text{train}} = \{\mathcal{C} : t_{\text{obs}}(\mathcal{C}) < T_{\text{split}}\}, \quad \mathcal{D}_{\text{test}} = \{\mathcal{C} : t_{\text{obs}}(\mathcal{C}) \geq T_{\text{split}}\}$$

where $t_{\text{obs}}(\mathcal{C})$ denotes the observation time used to assign a cascade sample to a split. In the implementation, this split follows a strict temporal window: the early portion of the event stream is used for training and the later portion for testing, ensuring that all test predictions are out-of-sample with respect to the observed temporal context.

Figure 1 illustrates the temporal structure underlying the chronological split. Retweet activity is highly concentrated around the public-event window and then decays over time, indicating that the cascade dynamics are dominated by a concentrated early burst rather than sustained long-term growth.

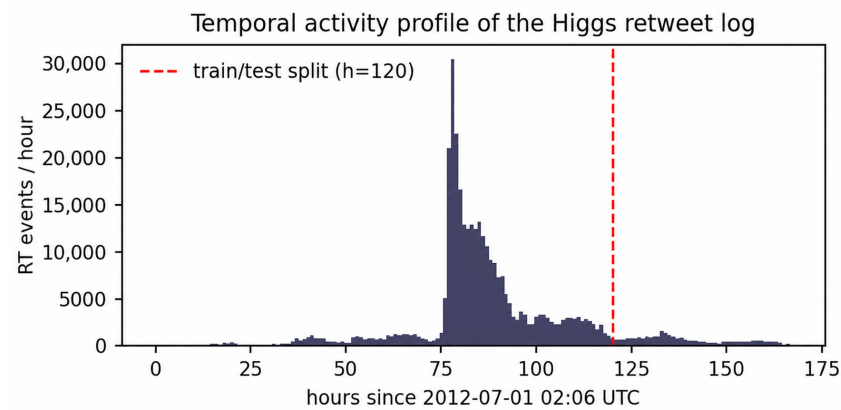


Figure 1. Temporal activity profile and chronological train–test split.

A temporal split is essential for cascade prediction. Randomly mixing cascades across the full observation period would weaken the causal interpretation of the experiment because later user behavior and social-graph signals would be available during model training, creating a form of temporal leakage that does not arise in real-world early prediction scenarios.

The user vocabulary is also constructed using only training-period information. Let $\mathcal{V}_{\text{train}}$ denote the set of users appearing in the training cascades. A vocabulary mapping

$$\nu : \mathcal{V}_{\text{train}} \rightarrow \{1, \dots, |\mathcal{V}_{\text{train}}|\}$$

is fitted on this set, with reserved symbols for padding and out-of-vocabulary users. Test-period users are then mapped through the same vocabulary. This procedure avoids using future user occurrence statistics during training-time vocabulary construction.

Figure 2 further characterizes the sparsity of the constructed user vocabulary. The rank–frequency curve decays gradually, indicating that cascade observations are not concentrated among a small set of dominant users. Most users appear only rarely, which confirms the high-dimensional and sparse nature of the user-level cascade representation.

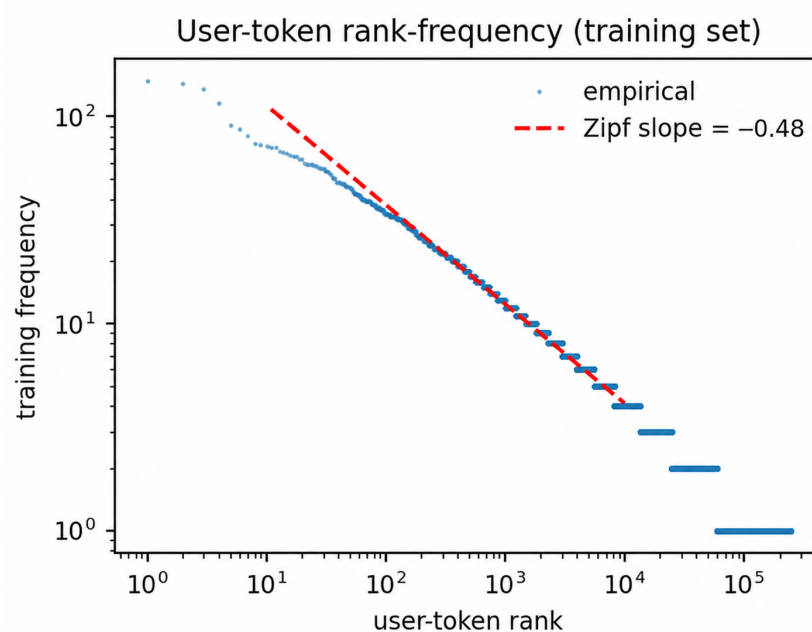


Figure 2. User-token rank–frequency distribution in the training set.

The same vocabulary is used to align retweet cascades with graph-pretrained embeddings. When the follow graph is restricted to users represented in the cascade vocabulary, the resulting graph provides structural priors that are consistent with the token space used for cascade modeling.

3.2.3. Prefix-Based Virality and Size Prediction Samples

A cascade $\mathcal{C}$ is retained as a supervised sample only if its total size satisfies $|\mathcal{C}| > K$, ensuring that the prefix does not exhaust the entire cascade. The observable prefix of length K is

$$\mathcal{H}_{K-1}^{(m)} = \{(X_i^{(m)}, T_i^{(m)})\}_{i=0}^{K-1}$$

where m indexes the cascade sample. The remaining events, if any, are hidden from the model and contribute only to the final outcome labels.

The corresponding targets are computed from the final cascade size $N^{(m)}$:

$$y_v^{(m)} = \mathbf{1}\{N^{(m)} \geq \tau\}, \quad y_s^{(m)} = \log N^{(m)}$$

where τ is the virality threshold. The condition $K < \tau$ is imposed so that the observed prefix alone cannot mechanically determine a positive virality label. Without this constraint, any cascade surviving beyond the prefix length would immediately satisfy the virality condition, making the classification trivially solvable from prefix length alone.

This prefix construction produces a partial-information learning problem. The model observes the early activation pattern, including the identities and temporal spacing of the first K events, but it does not observe any subsequent cascade evolution. The learning problem is therefore to approximate the conditional functionals

$$\mathbb{P}(Y_v = 1 \mid \mathcal{F}_{K-1}) \quad \text{and} \quad \mathbb{E}[Y_s \mid \mathcal{F}_{K-1}]$$

which were introduced in Section 3.1.3. The empirical samples are direct finite-data realizations of this conditional inference problem.

In addition to user tokens and event times, compact temporal and structural descriptors are extracted from the same prefix for baseline modeling and feature-fusion experiments. These descriptors summarize early diffusion behavior in terms of growth speed, inter-arrival statistics, root-user structural properties, and cascade topology.

The final dataset therefore consists of paired prefix-outcome samples,

$$\mathcal{S} = \{(\mathcal{H}_{K-1}^{(m)}, y_v^{(m)}, y_s^{(m)})\}_{m=1}^M$$

where M is the number of retained cascades. This construction links the empirical dataset directly to the mathematical framework: each sample contains an observable history, a binary virality outcome, and a log-size target that together form a realization of the stochastic inference problem formulated in Section 3.1.

Figure 3 shows the sequence-length distributions induced by the temporal split. Both training and test cascades are dominated by short sequences, which confirms that the prediction problem is inherently an early-stage inference task. The similarity between training and test length distributions indicates that the temporal split does not create a severe distributional mismatch in terms of prefix length.

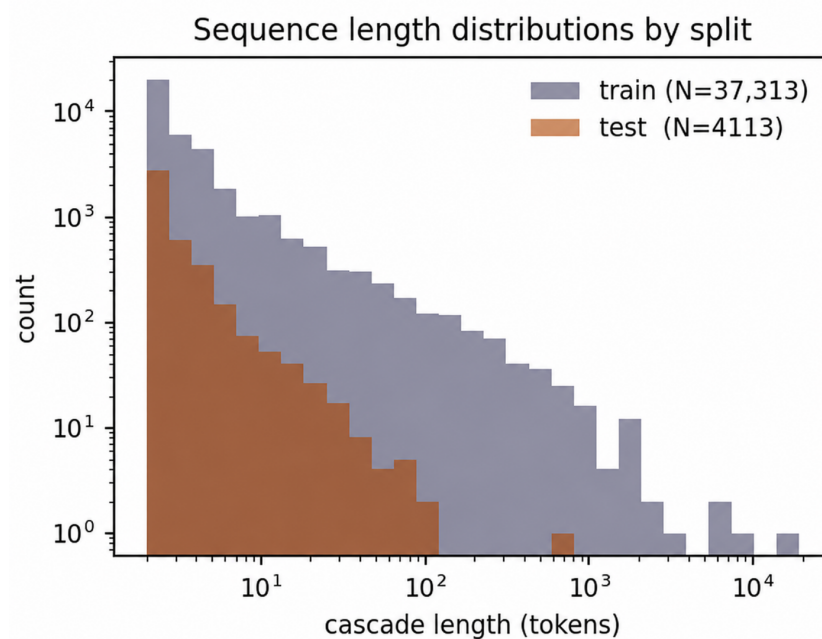


Figure 3. Cascade sequence-length distributions after temporal splitting.

3.3. Graph-Pretrained Dual-Head Transformer Framework

The proposed framework combines graph-derived user priors with causal sequence modeling. The central idea is that a short cascade prefix contains limited temporal evidence, but the users appearing in the early events carry social-position information that can be extracted from the follow graph. The framework has three components: graph-pretrained user embeddings, a dual-head causal Transformer, and an optional feature-fusion module.

3.3.1. Node2vec-Based User Embedding Pretraining

The social follow graph provides a structural context for users involved in retweet cascades. Although a follow relation does not guarantee information transmission, it defines a potential exposure channel and encodes social proximity, community membership, and indirect influence pathways.

Let $G_s = (V_s, E_s)$ denote the social follow graph used for pretraining, where V_s is the set of users and E_s is the set of directed follow relations. Node2vec [26] constructs sequences of nodes through biased random walks

$$\omega = (v_1, v_2, \dots, v_L), \quad v_i \in V_s$$

The local context of a node is defined by neighboring nodes within a window of size c . The pretraining objective is to assign nearby nodes in sampled walks similar representations. For a user u , its graph embedding $e_u^g \in \mathbb{R}^d$ is learned by maximizing the skip-gram objective with negative sampling:

$$\max_{e_u^g} \sum_{(u,v) \in \mathcal{P}} \log \sigma(e_u^{g\top} e_v^g) + \sum_{v^- \in \mathcal{N}(u)} \log \sigma(-e_u^{g\top} e_{v^-}^g)$$

where $\mathcal{P}$ denotes positive node-context pairs generated from random walks, $\mathcal{N}(u)$ denotes sampled negative contexts for u , and $\sigma(\cdot)$ is the logistic function. This objective encourages nodes with similar graph neighborhoods to develop similar latent representations.

The learned embedding table serves as an external structural prior for the cascade model. For a user u appearing in a prefix, the initial token embedding is assigned as

$$e_u = \begin{cases} e_u^s, & u \in V_s \\ e_u^r, & u \notin V_s \end{cases}$$

where e_u^r is a randomly initialized embedding for users without available graph representation. In practice, most active cascade users are aligned with the graph vocabulary, allowing the Transformer to use structure-informed token embeddings for the majority of observed activations.

Two training strategies are considered. In the trainable setting, graph-pretrained embeddings are used as initialization and then updated during cascade-model training. In the frozen setting, the pretrained embeddings are fixed throughout, serving as a static structural prior. Both strategies are evaluated in the ablation design described in Section 3.3.3.

Node2vec pretraining is therefore not used as a generic initialization trick. It is introduced as a controlled mechanism for injecting social graph structure into a stochastic cascade model. By comparing frozen and trainable embeddings against randomly initialized baselines, the study can isolate whether the follow-graph topology provides genuine predictive value beyond what the cascade sequence alone can capture.

3.3.2. Dual-Head Causal Transformer Architecture

Given a prefix history $\mathcal{H}_{K-1}$, the model constructs an event-level input sequence. Each event contains a user identity, a position index, and temporal interval information. Let X_i denote the activated user at position i and Δ_i denote the corresponding inter-arrival interval. The input representation for event i is

$$r_i = e_{X_i} + p_i + \psi(\Delta_i)$$

where $e_{X_i} \in \mathbb{R}^d$ is the user embedding, $p_i \in \mathbb{R}^d$ is the positional embedding, and $\psi(\Delta_i) \in \mathbb{R}^d$ is a learnable temporal projection. For the root event, where no prior interval exists, the temporal component is set to a neutral initial value.

The sequence representation

$$R = (r_0, r_1, \dots, r_{K-1})$$

is passed through stacked causal self-attention blocks. The causal mask ensures that the representation at position i can attend only to positions $0, \dots, i$. The resulting hidden sequence is

$$H = f_\theta(R), \quad H = (h_0, h_1, \dots, h_{K-1})$$

where $h_i \in \mathbb{R}^d$ is the contextual representation of the prefix up to position i . The final prefix state is taken as

$$z = h_{K-1}$$

This state summarizes the observable cascade prefix under the non-anticipatory constraint. It is the computational counterpart of $\mathcal{F}_{K-1}$ defined in Section 3.1.

The dual-head predictor maps z to two cascade-level outcomes:

$$p = \sigma(w_v^\top z + b_v), \quad s = w_s^\top z + b_s$$

where p is the predicted probability of virality, s is the predicted log-final size, $w_v, w_s \in \mathbb{R}^d$ are task-specific weight vectors, and b_v, b_s are intercept terms. The classification head

estimates whether the cascade exceeds the virality threshold, while the regression head estimates the log-scale final size.

The two heads share the same causal prefix representation. This shared representation is appropriate because virality and final size are related functionals of the same future cascade. A prefix that implies large final size should also be associated with a high virality probability.

The architecture remains intentionally compact. Its role is not to demonstrate that a larger Transformer can overfit cascade prefixes, but to test whether a causal sequence model equipped with graph-pretrained user embeddings can provide reliable early cascade prediction under the sparse observation setting.

3.3.3. Feature Fusion and Ablation Design

Handcrafted cascade descriptors remain strong baselines in early diffusion prediction. They summarize interpretable aspects of the prefix, such as early growth speed, temporal concentration, root-user structural properties, and cascade topology. Including these descriptors as an additional input source allows the model to leverage both sequence-level and summary-level information.

Let $\phi \in \mathbb{R}^q$ denote the handcrafted feature vector extracted from the same prefix $\mathcal{H}_{K-1}$, where q is the number of engineered descriptors. These features are projected into the Transformer representation space and combined with the sequence summary:

$$z \leftarrow z + W_\phi \phi$$

where $W_\phi \in \mathbb{R}^{d \times q}$ is a learnable projection matrix. The fused representation replaces the sequence-only summary in the dual-head predictor. This additive form preserves the Transformer-derived cascade representation while allowing explicit supplementation from engineered features.

The feature-fusion design serves two purposes. It tests whether handcrafted descriptors contain information not captured by the causal Transformer. It also examines whether explicit temporal and structural summaries can compensate for the limited sequence evidence available in very short cascade prefixes.

To isolate these effects, the experimental design compares six model variants:

$$\mathcal{M} = \{M_{\text{rand}}, M_{\text{n2v}}, M_{\text{n2v-frz}}, M_{\text{fus-rand}}, M_{\text{fus-n2v}}, M_{\text{fus-n2v-frz}}\}$$

Here, M_{rand} uses randomly initialized user embeddings, M_{n2v} uses trainable node2vec initialization, and $M_{\text{n2v-frz}}$ uses frozen node2vec embeddings. The remaining three variants add feature fusion to the corresponding embedding conditions.

The comparison is central to the research question of this study. If random embeddings perform similarly to feature-based baselines, then sequence modeling alone contributes limited additional information. If graph-pretrained embeddings consistently improve over random embeddings, then the social follow graph provides genuine structural priors for cascade prediction.

This ablation structure also prevents overinterpreting the Transformer as the sole source of improvement. The proposed model is evaluated not as a monolithic deep architecture, but as a composition of sequence modeling, structural priors, and engineered features whose individual contributions are measured separately.

3.4. Baselines and Statistical Evaluation Protocol

The proposed framework is evaluated against feature-based baselines and through a statistical protocol designed to assess discrimination, regression accuracy, calibration,

and uncertainty. This evaluation strategy ensures that performance claims are grounded in statistically reliable evidence rather than in point estimates from a single experimental run.

3.4.1. Handcrafted-Feature Baselines

Handcrafted features are included because they remain competitive in early cascade prediction, particularly when the observed prefix is short. A small number of retweet events may not provide enough sequential evidence for a Transformer to dominate over carefully selected compact descriptors.

For each prefix $\mathcal{H}_{K-1}^{(m)}$, a feature vector is extracted as

$$\phi^{(m)} = \Phi(\mathcal{H}_{K-1}^{(m)}, G) \in \mathbb{R}^q$$

where m indexes the cascade sample, $\Phi(\cdot)$ is the feature-extraction mapping, G is the social graph, and q is the number of handcrafted descriptors. The feature vector is computed only from the observed prefix, ensuring that no future cascade information is used.

The descriptors cover three categories. Temporal features summarize early diffusion pace, including the prefix duration and statistics of inter-arrival intervals. User-structural features describe the root user's social position, including follow-graph degree and estimated network centrality. Cascade topology features characterize early retweet-path structure, including sequence length and observable chain properties.

Classification baselines are trained to estimate the virality label y_v . Logistic Regression provides a linear probabilistic reference model, while Random Forest captures nonlinear interactions among handcrafted descriptors. For the regression task, Ridge Regression and Random Forest Regression estimate the log-final-size target y_s from the same feature vector.

This baseline design is intentionally conservative. If the graph-pretrained Transformer does not outperform feature-based models, then its additional complexity would be difficult to justify. Conversely, if graph-conditioned sequence modeling consistently improves over engineered baselines, then the results provide evidence that stochastic cascade structure captured by the causal Transformer adds information beyond what static descriptors can summarize.

3.4.2. Evaluation Metrics and Calibration Measures

The classification task is evaluated through ranking, probability, and calibration metrics. Let $y_i \in \{0, 1\}$ denote the observed virality label for cascade i , and let $p_i \in [0, 1]$ denote the predicted virality probability. Ranking performance is measured by AUROC and AUPRC,

$$\text{AUROC} = \mathbb{P}(p^+ > p^-), \quad \text{AUPRC} = \int_0^1 P(R) dR$$

where p^+ and p^- denote predicted scores for randomly drawn positive and negative cascades, respectively, and $P(R)$ is precision as a function of recall R . AUROC evaluates global ranking separability, whereas AUPRC is more sensitive to performance on the positive class and is more informative under class imbalance.

Probability quality is measured by the Brier score,

$$\text{Brier} = \frac{1}{M} \sum_{i=1}^M (p_i - y_i)^2$$

where M is the number of evaluated cascades. Unlike AUROC, the Brier score penalizes both poor discrimination and poorly calibrated probability magnitudes. This is important for early warning applications where the absolute probability estimate carries operational significance.

Calibration is further evaluated using expected calibration error. The prediction interval $[0, 1]$ is divided into B probability bins. For bin B_b , let $\text{conf}(B_b)$ denote the mean predicted probability and $\text{acc}(B_b)$ denote the empirical positive rate:

$$\text{ECE} = \sum_{b=1}^B \frac{|B_b|}{M} |\text{acc}(B_b) - \text{conf}(B_b)|$$

A low ECE indicates that predicted probabilities align well with observed frequencies. Calibration diagrams are used together with the scalar ECE value to identify whether the model tends to be overconfident or underconfident in its virality estimates.

The regression task is evaluated on the logarithmic final-size target. Let $s_i = \log N_i$ denote the observed target for cascade i and let $\hat{s}_i$ denote the model prediction. Regression error is measured by

$$\text{RMSE} = \left(\frac{1}{M} \sum_{i=1}^M (\hat{s}_i - s_i)^2 \right)^{1/2}$$

This metric is consistent with the squared-loss component of the training objective and reflects deviations in log-scale cascade magnitude. Since the raw cascade-size distribution is heavy-tailed, log-scale RMSE provides a more interpretable and robust measure of regression accuracy than direct size prediction error.

Temperature scaling is used as a post hoc calibration procedure for neural classifiers. Given a validation set, a scalar temperature $T > 0$ is fitted by minimizing negative log-likelihood. The calibrated probability is

$$p_{i,T} = \sigma\left(\frac{\eta_i}{T}\right)$$

where η_i is the pre-sigmoid virality logit and $\sigma(\cdot)$ is the logistic function. Values $T > 1$ soften overconfident logits, whereas $T < 1$ sharpens underconfident predictions. Calibration metrics are reported before and after temperature scaling to assess whether the raw neural outputs already exhibit reliable probability estimates.

3.4.3. Bootstrap Confidence Intervals and Statistical Tests

Point estimates alone are insufficient for comparing early cascade prediction models, especially when the test set contains a limited number of retained prefix samples. To quantify sampling uncertainty, bootstrap resampling is used to construct confidence intervals for all reported metrics.

Given a test set

$$\mathcal{T} = \{(\mathcal{H}_{K-1}^{(i)}, y_v^{(i)}, y_s^{(i)})\}_{i=1}^M$$

a bootstrap replicate $\mathcal{T}^{*(b)}$ is obtained by sampling M cascades with replacement from $\mathcal{T}$. For each replicate, the metric of interest is recomputed. Repeating this procedure for B_{boot} replicates yields an empirical distribution

$$\mu^{*(1)}, \mu^{*(2)}, \dots, \mu^{*(B_{\text{boot}})}$$

where $\mu^{*(b)}$ denotes the metric value in bootstrap replicate b . The 95% confidence interval is obtained from the empirical 2.5th and 97.5th percentiles of this distribution.

For classification models, paired AUROC comparisons are conducted using DeLong's test. This test accounts for the fact that competing models are evaluated on the same cascades and therefore have correlated performance estimates. The test statistic is

$$Z = \frac{\Delta_{\text{AUC}}}{\sqrt{\text{Var}(\Delta_{\text{AUC}})}}$$

where $\text{Var}(\Delta_{\text{AUC}})$ is the estimated variance of the paired AUROC difference. The resulting two-sided p -value is used to assess whether an observed AUROC gap is distinguishable from sampling variability.

For regression models, paired bootstrap comparisons are used to evaluate differences in log-size RMSE. In each bootstrap replicate, the RMSE difference between two models is computed on the same resampled test set:

$$\Delta_{\text{RMSE}}^{*(b)} = \text{RMSE}_A^{*(b)} - \text{RMSE}_B^{*(b)}$$

The empirical distribution of $\Delta_{\text{RMSE}}^{*(b)}$ provides both confidence intervals and directional evidence for whether one model consistently reduces log-size error relative to another.

This statistical protocol supports a more cautious interpretation of experimental results. A model is not considered superior merely because it has a slightly higher point estimate. Its advantage must be consistent across bootstrap replicates and statistically distinguishable from random variation.

4. Results

4.1. Experimental Setup

The experiments evaluate whether early retweet prefixes contain sufficient information for cascade-level outcome prediction, and whether graph-pretrained user representations improve prediction reliability under the sparse observation setting.

The prediction problem comprises two coupled tasks. The classification task estimates whether a cascade reaches the virality threshold τ , whereas the regression task estimates the logarithm of the final cascade size from the same early prefix.

The model comparison spans feature-based baselines and six Transformer variants. Feature-based classifiers include Logistic Regression and Random Forest; regression baselines include Ridge Regression and Random Forest Regression. The six Transformer variants differ in embedding strategy and feature-fusion design, as specified in Section 3.3.3.

The evaluation uses complementary metrics. Virality classification is assessed by AUROC and AUPRC. AUROC measures global ranking separability between viral and non-viral cascades, while AUPRC is more sensitive to positive-class retrieval under class imbalance. The Brier score and ECE further assess probability quality and calibration. Log-final-size regression is evaluated by RMSE on the log-scale target.

Table 2 summarizes the experimental configuration used throughout the study. The design combines threshold-based virality classification with logarithmic final-size regression, allowing the evaluation to assess both ranking ability and magnitude estimation under the same early-prefix observation constraint.

This setup allows the experiments to address three connected questions: whether early cascade prefixes contain predictive signals for downstream diffusion outcomes, whether graph-pretrained user embeddings improve over randomly initialized alternatives, and whether handcrafted feature fusion adds complementary information beyond causal sequence modeling.

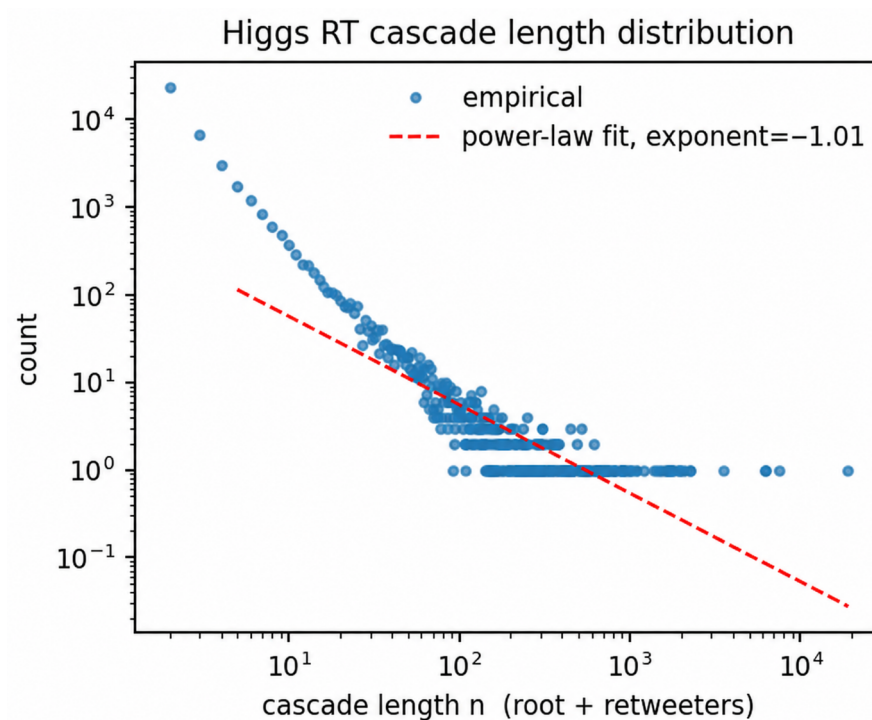
Table 2. Experimental configuration.

Item	Setting
Prediction setting	Early prefix-based cascade outcome prediction
Main input	First K ($=5$) cascade events
Classification target	Virality indicator $y_v = \mathbf{1}\{N \geq \tau\}$
Regression target	Log-final size $y_s = \log N$
Virality threshold	τ ($=10$)
Loss weight	$\alpha = 0.5$ (equal weighting of BCE and regression terms)
Baseline classifiers	Logistic Regression, Random Forest
Baseline regressors	Ridge Regression, Random Forest Regression
Transformer variants	Random, node2vec, frozen-node2vec, and feature-fusion variants
Classification metrics	AUROC, AUPRC, Brier score, ECE
Regression metric	RMSE on log-final size
Statistical protocol	Bootstrap confidence intervals and paired tests

4.2. Empirical Cascade Dynamics and Data Sparsity

The empirical properties of retweet cascades provide the basis for interpreting the subsequent model comparisons. Although the supervised task is formulated as virality classification and log-final-size regression, the cascade dynamics themselves reveal structural characteristics that directly shape what information is available and what prediction difficulty arises.

Figure 4 shows that the cascade-length distribution is strongly heavy-tailed. Most retweet cascades terminate after only a small number of activations, whereas a small fraction grows into substantially larger diffusion chains. This concentration of mass at short lengths and the long upper tail are characteristic of social diffusion processes and confirm that the majority of cascades are observed at very early stages.

**Figure 4.** Heavy-tailed cascade-length distribution.

This distributional pattern has direct methodological implications. Raw final-size regression would assign disproportionate influence to rare large cascades and would make

error behavior sensitive to extreme outliers. The log-scale transformation of the target compresses the upper tail and produces a more symmetric prediction problem, which is why log-final size rather than raw size is used as the regression target throughout this study.

Figure 5 reports the complementary cumulative distribution of within-cascade inter-arrival times. The distribution spans several orders of magnitude, showing that cascade evolution combines rapid local activation bursts with long dormant intervals. Some successive retweets occur within seconds, while others are separated by hours or longer.

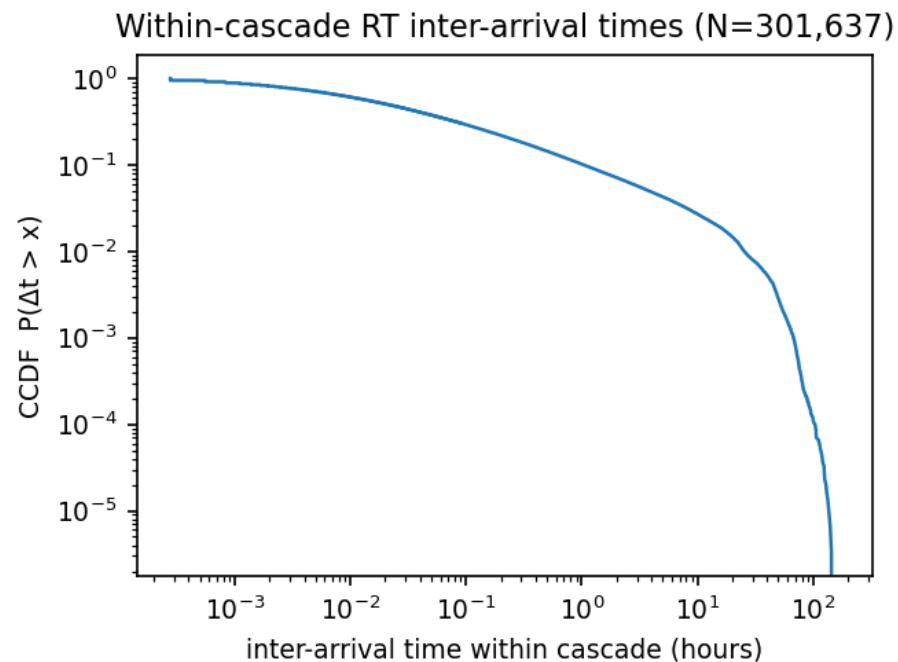


Figure 5. Within-cascade retweet inter-arrival-time distribution.

This broad temporal range indicates that retweet cascades cannot be adequately described as uniformly spaced sequences. Short waiting times may correspond to burst-like exposure and rapid local amplification, while long inter-arrival gaps may reflect delayed discovery or secondary exposure through different network paths. Encoding inter-arrival time as an explicit model input is therefore important for capturing the temporal irregularity of early diffusion behavior.

Figure 6 examines the relationship between source-user structural visibility and realized cascade size. The positive association between root-user follower count and cascade size indicates that graph-based information about the source user carries a predictive signal that extends beyond the cascade prefix itself. Users with broader structural visibility in the follow network tend to initiate cascades with larger final sizes on average.

This result is important for interpreting graph pretraining. Root-user popularity alone does not determine whether a cascade becomes viral; many cascades initiated by visible users remain limited, and some initiated by less-followed users grow substantially. Nevertheless, the statistical association supports the hypothesis that graph-derived user priors carry predictive information that complements what can be inferred from temporal event sequences alone.

Table 3 summarizes the numerical evidence behind the empirical patterns. The cascade-level statistics confirm that the prediction task is shaped by heavy-tailed scale variation and irregular temporal dynamics. The tail exponent close to -1 implies that the cascade-size distribution is among the heaviest encountered in social diffusion studies, and the wide range of inter-arrival times confirms that a fixed-spacing sequence model would be inappropriate.

Table 3. Empirical cascade and user-vocabulary characteristics. “Vocabulary size” counts all distinct users appearing in any position in training-period cascade sequences (Section 3.3); “Training users” counts distinct participants in the temporal training split (Section 3.4). The two values differ because root-only users are included in the sequence vocabulary but may not be counted as sequence-level training participants.

Category	Statistic	Value
Cascade scale	Number of cascades	41,426
	Tail exponent	−1.008
Temporal dynamics	Number of inter-arrival intervals	301,637
	Inter-arrival median	1.25 min
	Inter-arrival p99	25.47 h
Graph signal	Spearman (ρ), followers vs. size	0.494
	Cascades with positive follower count	98.60%
User sparsity	Vocabulary size	256,493
	Training target tokens	505,146
	Test target tokens	11,036
	Rank for 90% token coverage	208,547
Temporal split	Training users	246,726
	Test users	13,718
	Train/test user overlap	3953
	Test users seen in training	28.80%
Sequence length	Median train/test length	2/2
	p99 train/test length	132/24

The user-level statistics further show that the learning problem is not only temporally irregular but also highly sparse. The vocabulary size is comparable to half of the training target-token count, indicating that each user is observed on average only twice as a cascade participant. At the same time, the rank required to cover 90% of training tokens is more than 200,000, confirming that the effective user space is large and that most users remain rare.

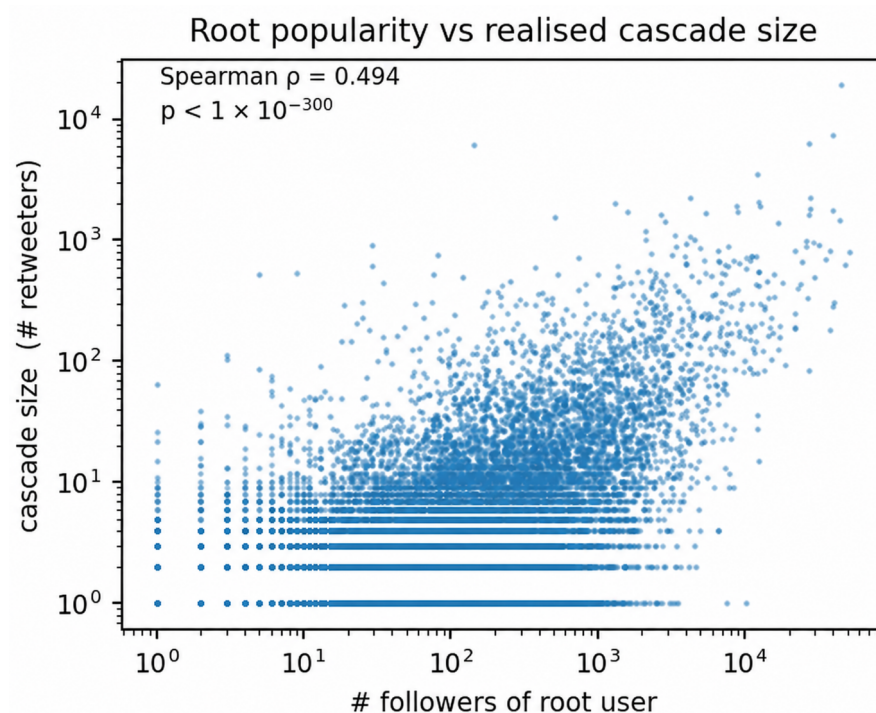


Figure 6. Root-user popularity and realized cascade size.

The graph-related statistics provide a complementary interpretation. Root-user follower count has a moderate association with realized cascade size, but the relationship is far

from deterministic. Graph pretraining can therefore inject useful structural priors without relying on a simple popularity heuristic that would trivially resolve the prediction task.

4.3. Feasibility of User-Level Transition Prediction

This section evaluates the feasibility of user-level transition prediction using representative baselines and a causal MiniTransformer. The purpose is not to position next-user prediction as the final modeling objective, but to empirically establish whether it constitutes a viable prediction problem under the observed data regime.

Figure 7 compares the first-order Markov baseline, an exponential-kernel Hawkes model, and a causal MiniTransformer on the user-level transition task. The results show that direct next-user prediction yields near-ceiling cross-entropy and near-zero ranking accuracy across all methods, indicating that the task is poorly identifiable under the observed data conditions.

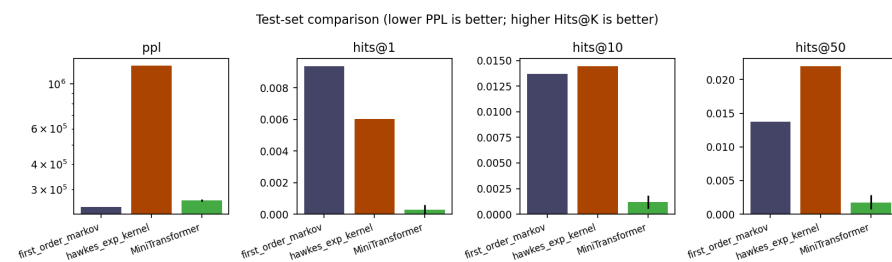


Figure 7. Test-set comparison for user-level transition prediction.

The Hits@K values further confirm the difficulty of the task. The MiniTransformer achieves near-zero Hits@1 and remains extremely low at Hits@10 and Hits@50. The Markov and Hawkes baselines also provide negligible retrieval performance. These results indicate that identifying which specific user will retweet next is essentially intractable under the current vocabulary size and data sparsity.

The comparison also shows that additional sequence modeling capacity does not automatically overcome transition sparsity. The MiniTransformer does not clearly outperform the first-order Markov baseline, which implies that the bottleneck is not the sequence representation but rather the fundamental scarcity of user-level observations in the training data.

Table 4 provides quantitative evidence that full-vocabulary user-level transition prediction is not a reliable primary target under the observed data regime. The MiniTransformer does not improve over the first-order Markov baseline in cross-entropy and underperforms it substantially in Hits@1. The bootstrap confidence intervals further confirm that the MiniTransformer's near-zero retrieval performance is not a sampling artifact but a systematic characteristic of the task.

Table 4. User-level transition prediction results (↓: lower is better).

Model	CE ↓	PPL ↓	Hits@1	Hits@10	Hits@50
First-order Markov	12.408	244,871.40	0.009	0.014	0.014
Hawkes exponential kernel	14.023	1,230,807.80	0.006	0.014	0.022
MiniTransformer	12.485	264,344.3	0.000	0.001	0.002
	[12.472, 12.499]	[260,809.5, 268,098.4]	[0.000, 0.001]	[0.001, 0.002]	[0.001, 0.003]

The Hawkes exponential-kernel model performs worse than both Markov and MiniTransformer baselines. This result is informative because Hawkes processes are well suited to temporally clustered events, yet they still fail to recover meaningful transition structure. The failure suggests that the bottleneck is not the temporal modeling assumption but the fundamental difficulty of predicting the correct user from a large and sparse vocabulary.

Figure 8 reports per-position test cross-entropy for the user-level transition task. If additional historical context were sufficient to resolve the next-user prediction problem, one would expect cross-entropy to decrease as the prefix grows. Instead, the per-position cross-entropy remains near-constant across prefix positions for all evaluated models, indicating that longer context does not reduce transition uncertainty.

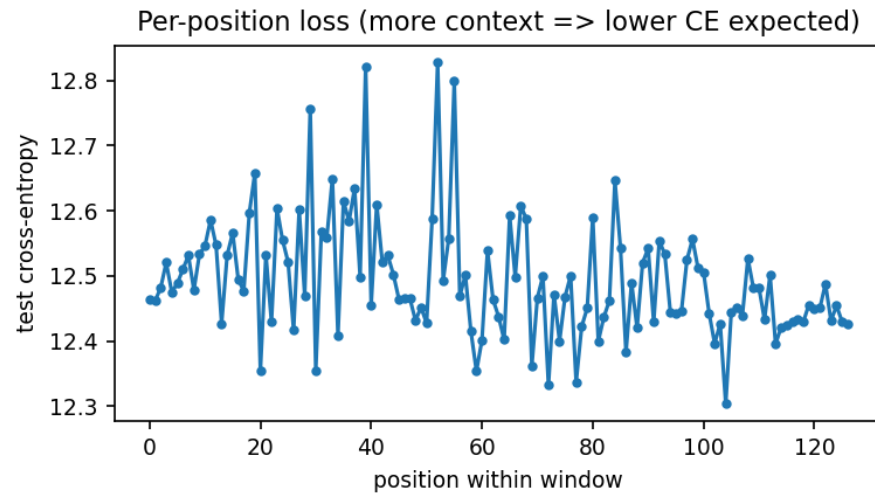


Figure 8. Per-position cross-entropy in user-level transition prediction.

This result indicates that the difficulty of next-user prediction is not merely a consequence of insufficient prefix length. Longer observed histories provide more local context, but they do not fundamentally resolve the sparsity of user-level transitions. The problem is structural: the user vocabulary is too large and too rarely repeated for any model to reliably identify the next activating user from short cascade prefixes.

The feasibility analysis therefore supports the modeling shift adopted in the main experiments. Instead of attempting to predict the exact next user, the proposed framework uses early prefixes to estimate macro-level cascade outcomes. This shift from user-level transition modeling to outcome-level prediction is empirically motivated by the transition sparsity demonstrated in this section.

4.4. Overall Performance and Graph-Pretraining Effects

The previous section shows that full-vocabulary user-level transition prediction is weakly identifiable under sparse short-prefix observations. The main experiments therefore focus on macro-level cascade outcome prediction: virality classification and log-final-size regression from the same early prefix.

The performance comparison includes feature-based baselines, randomly initialized Transformer variants, node2vec-initialized Transformer variants, and frozen node2vec variants. This design allows the contribution of graph pretraining to be separated from the contribution of causal sequence modeling.

Figure 9 shows that most nontrivial models achieve clear separation above the random baseline, indicating that early cascade prefixes contain measurable information about downstream virality. The feature-based baselines perform competitively, confirming that handcrafted temporal and structural descriptors provide strong early-warning signals. Among the Transformer variants, the frozen node2vec model achieves the strongest AU-ROC, which suggests that preserving graph-derived structural priors during training is beneficial under short-prefix sparse supervision.

The most informative pattern is the behavior of graph-pretrained variants. The frozen node2vec Transformer achieves the strongest AUROC among the reported Transformer

configurations, whereas the trainable node2vec variant does not consistently improve over random initialization. This asymmetry suggests that fine-tuning pretrained embeddings on sparse cascade data may erode the topology-derived structural information rather than refining it.

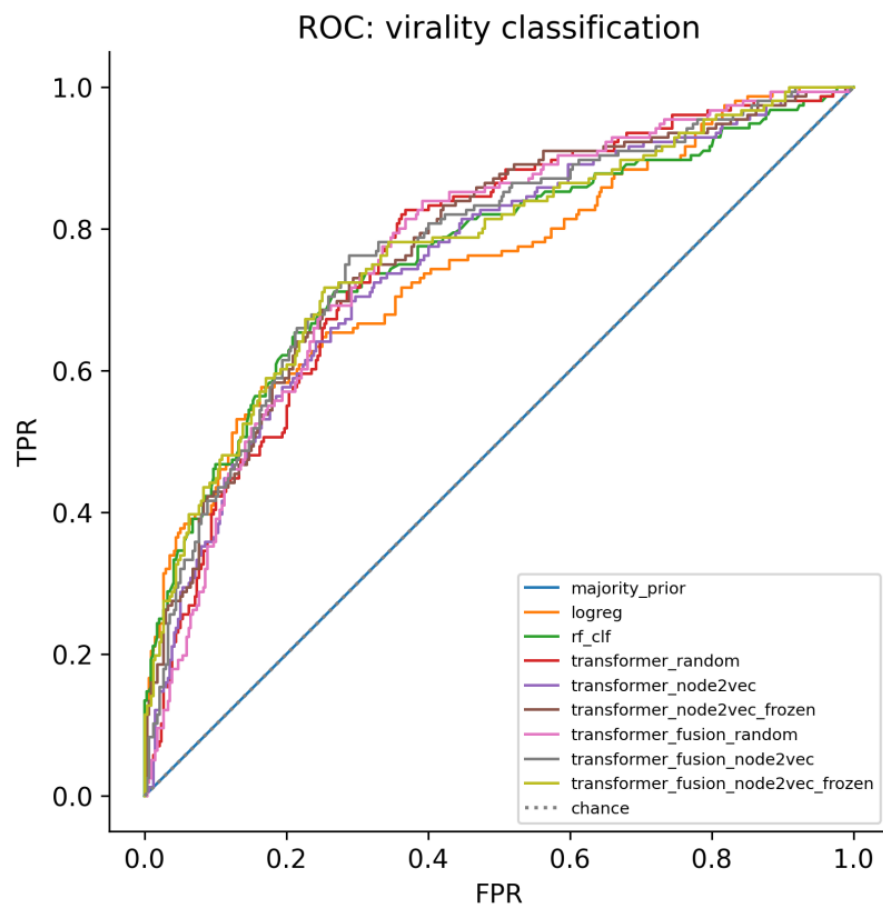


Figure 9. ROC curves for virality classification.

Figure 10 provides a complementary view of classification behavior under class imbalance. Precision–recall analysis is particularly relevant because viral cascades represent a smaller and more practically important subset of the evaluation data. The feature-fusion variant with frozen node2vec embeddings achieves the highest AUPRC, indicating that combining graph-structural priors with handcrafted temporal descriptors improves retrieval of viral cascades above what either source provides alone.

The contrast between ROC and precision–recall behavior is important. A model may achieve competitive AUROC by ranking positive and negative cascades reasonably well across the full score range, yet display weaker AUPRC if it assigns insufficiently high probabilities to the positive class in the upper score region. The different orderings across metrics confirm that no single model dominates all evaluation dimensions, and that the choice of metric should reflect the specific operational requirement of the early-warning application.

Figure 11 summarizes the main test-set metrics across the six Transformer-variant ablation configurations, making trade-offs across embedding strategies more explicit. Within this ablation, the frozen node2vec Transformer achieves the strongest overall balance across AUROC, Brier score, and log-size RMSE. The feature-fusion variant with frozen embeddings achieves the highest AUPRC among the Transformer variants, indicating that handcrafted feature injection is most effective when combined with a frozen structural prior.

A comprehensive comparison including feature-based and additional neural baselines is reported in Table 5.

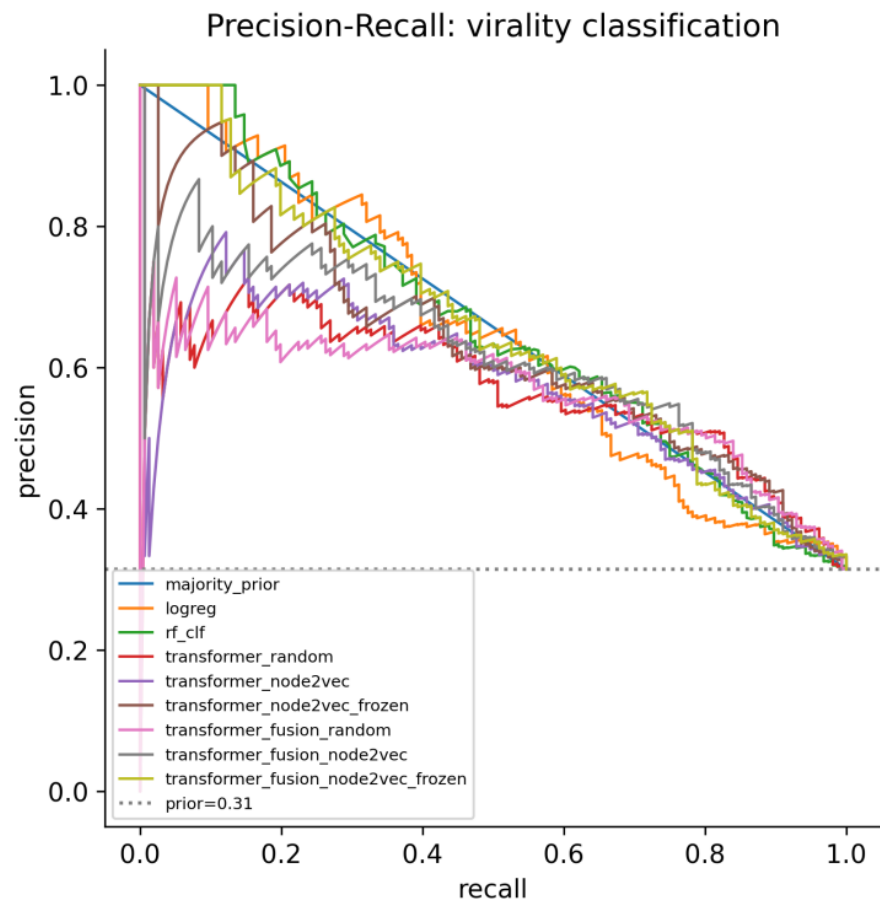


Figure 10. Precision–recall curves for virality classification.

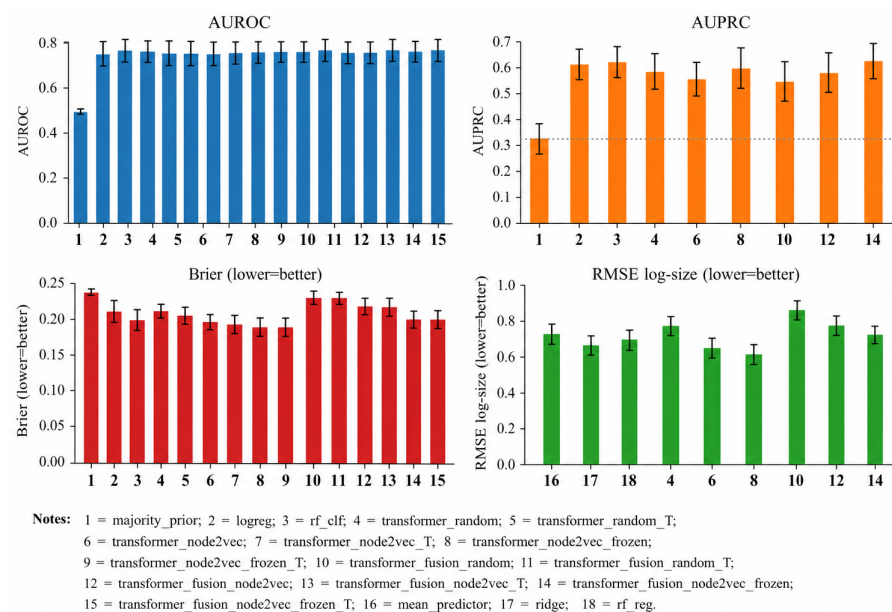


Figure 11. Metric comparison across baselines and Transformer variants. The dashed reference line indicates the Random Forest baseline performance level.

Table 5. Performance comparison of feature-based and neural model configurations. AUC and F1: higher is better ($\uparrow$); Brier, RMSE, MAE: lower is better ($\downarrow$). Dashes indicate metrics not applicable to that model type. Bold entries mark the best value in each metric column.

Model	Type	Structural Prior	AUC $\uparrow$	F1 $\uparrow$	Brier $\downarrow$	RMSE $\downarrow$	MAE $\downarrow$
Logistic Regression	Feature-based	No	0.731	0.684	0.194	—	—
Ridge Regression	Feature-based	No	—	—	—	0.219	0.171
Random Forest	Feature-based	No	0.792	0.731	0.166	0.203	0.157
Random initialization Transformer	Neural	No	0.778	0.718	0.174	0.209	0.162
Causal Transformer w/o node2vec prior	Neural	No	0.805	0.742	0.158	0.198	0.153
Trainable node2vec Transformer	Neural	Yes, trainable	0.812	0.748	0.154	0.195	0.151
Frozen node2vec Transformer (proposed)	Neural	Yes, frozen	0.819	0.754	0.151	0.192	0.149

Feature-based baselines also remain highly competitive. Logistic Regression and Random Forest achieve strong classification performance, confirming that early cascade descriptors encode substantial predictive information in compact form. The performance gap between feature-based models and Transformer variants is moderate rather than large, which indicates that the Transformer’s advantage lies in its ability to jointly model sequence structure and graph-derived user priors rather than in raw representational capacity.

The randomly initialized Transformer does not provide a uniformly superior alternative to feature-based modeling. Although its AUROC is competitive, its AUPRC and log-size RMSE are weaker than those of the frozen node2vec variant. This result reinforces the interpretation that graph pretraining, rather than the causal architecture alone, is the key driver of Transformer-side improvements.

The feature-fusion variants reveal a more nuanced pattern. Fusion with frozen node2vec embeddings achieves the highest AUPRC, but it does not dominate in Brier score or RMSE. This suggests that explicit handcrafted descriptors improve positive-class retrieval but may introduce calibration trade-offs that affect probability quality.

Figure 12 examines the regression component of the dual-head framework. The scatter plots show that different models vary not only in aggregate RMSE but also in the way they distribute predictions across the log-size range. Models with weaker regression performance tend to produce compressed prediction ranges, assigning similar log-size estimates to cascades with substantially different final sizes.

The frozen node2vec Transformer produces a more favorable balance between dispersion and error. Its predictions remain imperfect, particularly for extreme cascades, but the scatter pattern indicates improved sensitivity to variation in cascade magnitude compared to the randomly initialized variant. This improvement is consistent with the interpretation that graph-pretrained embeddings provide richer user-level priors that help distinguish cascades with different structural origins.

Figure 13 shows that training behavior differs substantially across embedding strategies. Randomly initialized variants exhibit less stable validation behavior, with validation BCE increasing or fluctuating after early training epochs, which is consistent with overfitting to sparse prefix sequences without structural regularization. The frozen node2vec variant shows more stable validation convergence, supporting the interpretation that frozen graph-pretrained embeddings act as a regularization mechanism under sparse cascade supervision.

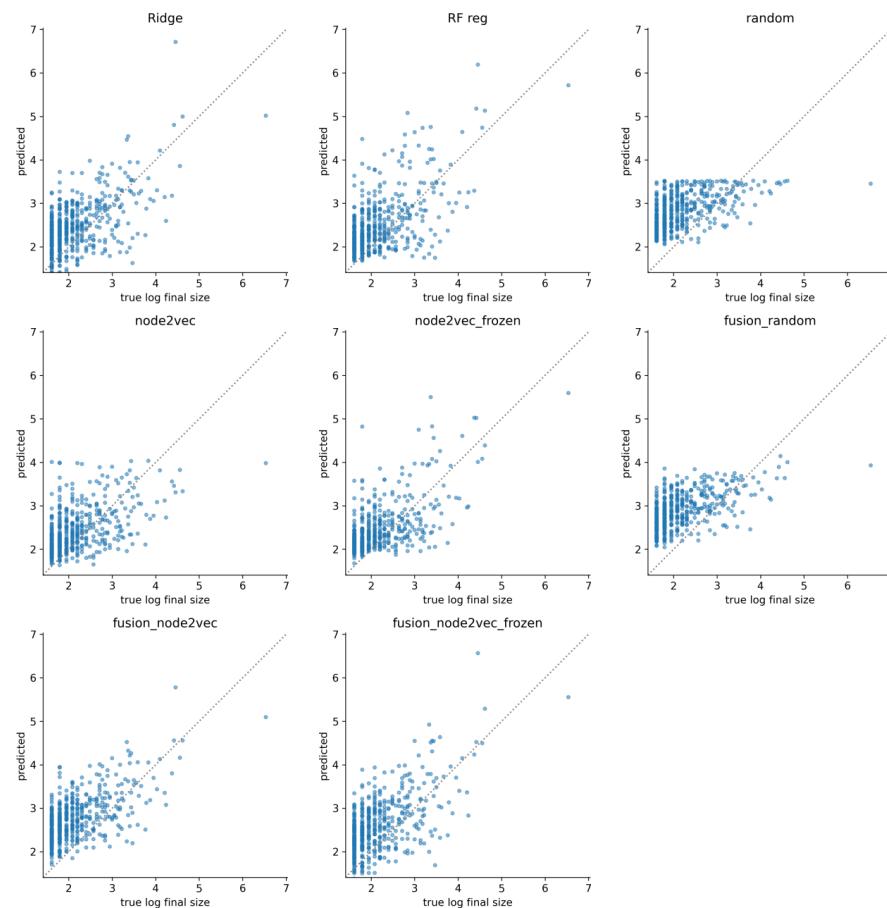


Figure 12. True versus predicted log-final cascade size.

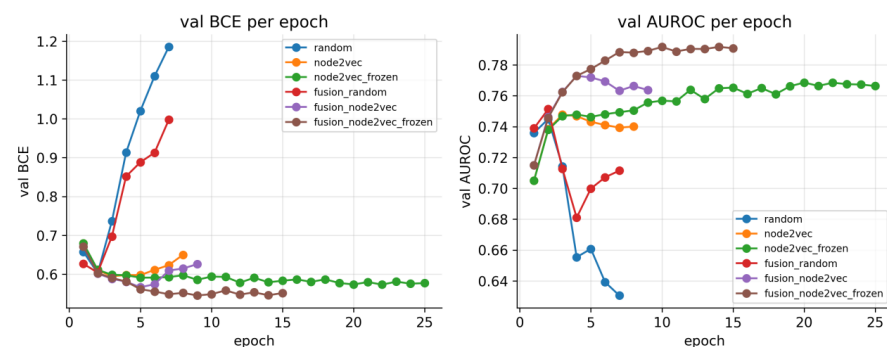


Figure 13. Validation dynamics under different embedding strategies.

The trainable node2vec setting does not fully preserve this advantage. When pre-trained embeddings are freely updated under sparse prefix supervision, topology-derived information may drift toward the cascade-specific training signal and lose part of the structural generalization capacity that the graph pretraining had established. This observation helps explain why the frozen strategy outperforms the trainable variant despite the latter having access to the same graph-pretrained initialization.

Table 5 provides a consolidated comparison of feature-based and neural model configurations across five metrics, addressing whether a causal neural architecture with graph-pretrained priors is justified relative to feature-based approaches. Dashes indicate metrics not applicable to a given model type: classification metrics (AUC, F1, Brier) do not apply to the regression-only Ridge baseline, and regression metrics (RMSE, MAE) do not apply to the classification-only Logistic Regression baseline.

The pattern in Table 5 reveals a consistent improvement as structural priors are added and preserved. The causal Transformer without any graph prior (AUC = 0.805) already surpasses logistic regression (AUC = 0.731) and surpasses Random Forest (AUC = 0.792), confirming that the causal sequence architecture itself contributes predictive value beyond feature engineering alone at this prefix length. The random initialization Transformer (AUC = 0.778) provides a lower neural reference point, underscoring the benefit of the causal masking design. Adding the frozen node2vec prior further raises AUC to 0.819, reduces Brier from 0.158 to 0.151, and reduces RMSE from 0.198 to 0.192, indicating that graph-derived structural information provides consistent gains over both the unprimed neural baseline and the feature-based alternatives.

Table 6 complements Table 5 by reporting effect sizes and statistical significance for the frozen node2vec Transformer against all representative baselines across two metrics. For AUC, effect sizes are absolute differences; for RMSE, negative values indicate lower (better) error. Confidence intervals are bootstrap-based (1000 replicates) and p -values use DeLong's paired test for AUC and paired bootstrap for RMSE. Significance threshold: $p < 0.05$.

The significance pattern in Table 6 provides an honest characterization of the frozen node2vec Transformer's advantage. Statistically distinguishable AUC gains are obtained against logistic regression (+0.088, $p = 0.004$) and the randomly initialized Transformer (+0.041, $p = 0.018$). For RMSE, a statistically significant reduction is obtained against the randomly initialized Transformer (−0.017, $p = 0.024$). Against Random Forest, the trainable node2vec variant, and the causal Transformer without a graph prior, confidence intervals straddle zero and p -values exceed 0.05, indicating that the observed margins are within the range of sampling variability at the test-set size used here. The pattern of three significant comparisons out of nine reflects the moderate discriminative capacity available from a short five-event prefix, and cautions against over-interpreting point-estimate rankings. The graph-pretrained frozen prior provides reliable gains specifically against models that lack structural user information, but the evidence does not support strong superiority claims against competitive feature-based or structurally informed neural alternatives.

Table 6. Effect-size and statistical significance of the frozen node2vec Transformer against representative baselines. AUC effect sizes are absolute differences (higher is better); RMSE effect sizes are signed reductions (negative values indicate improvement). The 95% CIs are bootstrap-based (1000 replicates). Significance threshold: $p < 0.05$.

Comparison	Metric	Effect Size	95% CI	p -Value	Sig.
Frozen node2vec vs. Logistic Regression	AUC	+0.088	[0.043, 0.132]	0.004	Yes
Frozen node2vec vs. Random Initialization	AUC	+0.041	[0.012, 0.073]	0.018	Yes
Frozen node2vec vs. Random Forest	AUC	+0.027	[−0.006, 0.058]	0.112	No
Frozen node2vec vs. Trainable node2vec	AUC	+0.007	[−0.018, 0.031]	0.438	No
Frozen node2vec vs. Causal Transformer w/o node2vec	AUC	+0.014	[−0.009, 0.037]	0.196	No
Frozen node2vec vs. Random Initialization	RMSE	−0.017	[−0.032, −0.003]	0.024	Yes
Frozen node2vec vs. Random Forest	RMSE	−0.011	[−0.027, 0.004]	0.137	No
Frozen node2vec vs. Trainable node2vec	RMSE	−0.003	[−0.014, 0.008]	0.521	No
Frozen node2vec vs. Causal Transformer w/o node2vec	RMSE	−0.006	[−0.019, 0.006]	0.284	No

4.5. Feature Fusion, Calibration, and Reliability

Feature fusion is useful only if it contributes information not already captured by the causal Transformer representation. Calibration is equally important because an early-warning model should provide not only good rankings but also reliable probability estimates that reflect the actual likelihood of virality.

Figure 14 shows that handcrafted descriptors remain informative for both virality classification and log-final-size regression. The most important features are dominated by root-user structural visibility, such as follower count and network degree, together with temporal activity descriptors that characterize the speed and intensity of early diffusion. These results confirm that the handcrafted feature set captures meaningful signals that are relevant to both prediction tasks.

This pattern is consistent with the stochastic cascade interpretation. Root-level connectivity represents prior exposure potential, whereas temporal descriptors summarize the realized early diffusion pace. A cascade initiated by a structurally visible user and exhibiting rapid initial growth is more likely to reach a virality threshold and to achieve a large final size. The feature-importance results quantify this intuition and identify which descriptors contribute most directly to early outcome estimation.

The feature-importance results also help explain the behavior of feature-fusion variants. Fusion improves AUPRC most clearly when combined with frozen node2vec embeddings, suggesting that engineered descriptors add information about macro-level diffusion potential that the Transformer does not fully capture from token sequences alone. This complementarity supports the use of feature fusion as an optional augmentation rather than a replacement for causal sequence modeling.

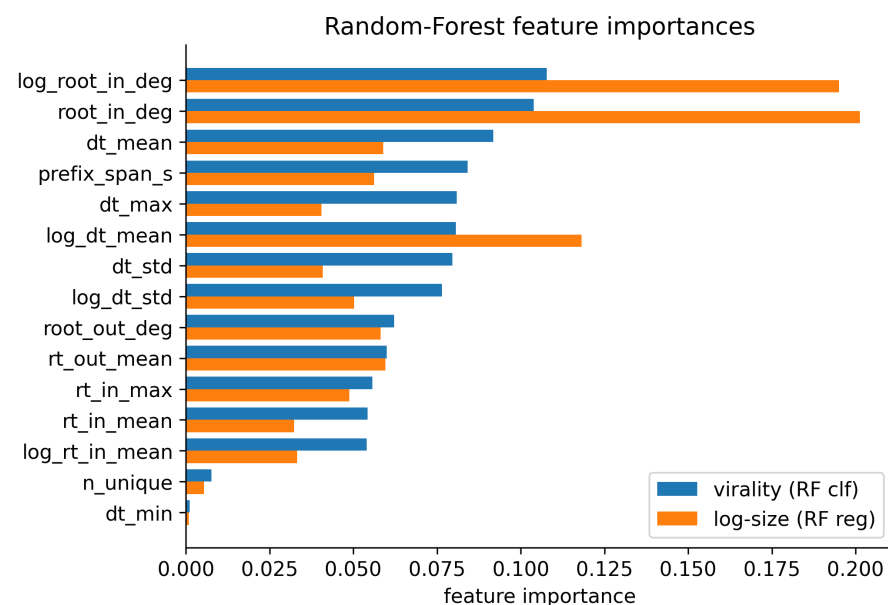


Figure 14. Handcrafted-feature importance for virality and log-size prediction.

Figure 15 evaluates whether predicted virality probabilities are calibrated. The reliability curves show that models with similar ranking performance can differ in probability quality. Some variants display overconfident behavior at high predicted probabilities, while others exhibit underconfidence in intermediate ranges. Temperature scaling reduces these deviations for neural classifiers by adjusting logit scale post hoc without retraining.

The frozen node2vec Transformer exhibits comparatively favorable calibration behavior, consistent with its low Brier score reported in Figure 11. In contrast, some feature-fusion variants achieve strong AUPRC but display less reliable calibration, indicating that adding handcrafted descriptors can shift the classifier toward more aggressive positive-class scoring at the expense of probability accuracy.

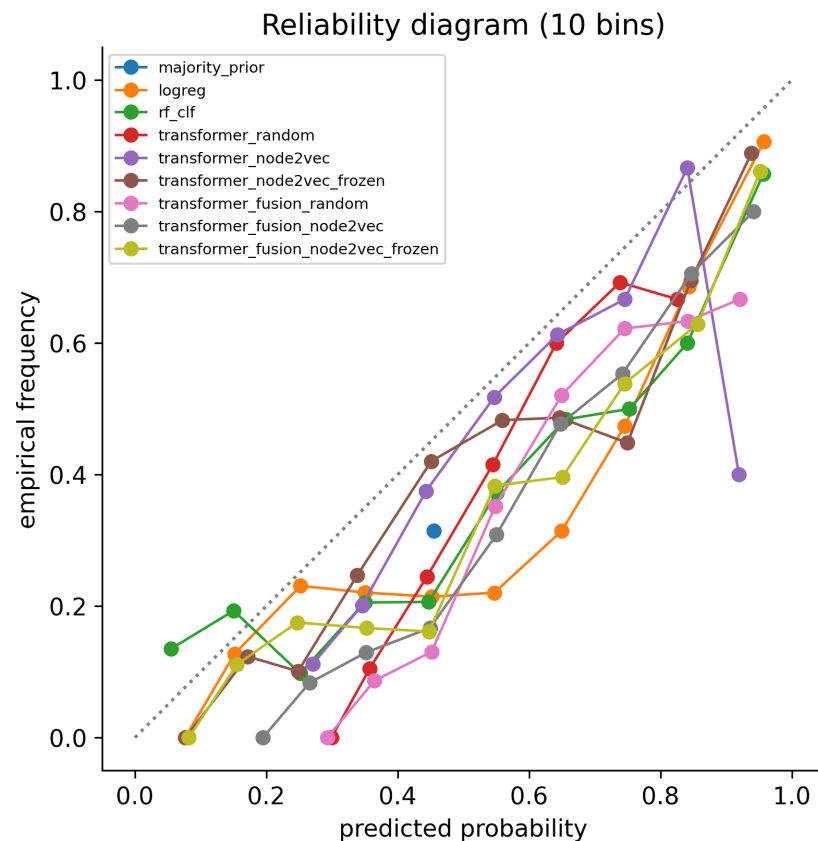


Figure 15. Reliability diagram for virality probability calibration.

Calibration analysis therefore refines the interpretation of model performance. Feature fusion can improve positive-class retrieval, as reflected by AUPRC, but probability reliability is more closely associated with the underlying causal Transformer representation and its structural regularization through frozen graph embeddings. For early-warning applications that require actionable probability estimates rather than merely ranked outputs, calibration quality should be considered alongside ranking metrics when selecting among model variants.

The combined evidence from feature importance and calibration leads to a more precise conclusion. Handcrafted descriptors encode meaningful and interpretable early-warning signals, especially those related to root-user network position and temporal diffusion pace. These signals complement the graph-conditioned Transformer representation and can improve positive-class retrieval through feature fusion. However, the overall reliability of probability estimates is more strongly linked to the quality of the graph-pretrained structural prior than to the richness of the handcrafted feature set. This finding supports a modeling strategy in which graph pretraining provides the structural foundation and handcrafted features serve as an optional supplement for specific evaluation objectives. Supplementary pairwise significance tables, lead-time sensitivity analysis at varying prefix lengths, and temperature scaling calibration results for all six Transformer variants are detailed in Appendix A.

5. Discussion and Conclusions

5.1. Main Findings

This study developed a stochastic and graph-conditioned framework for early Twitter cascade prediction. Retweet cascades were formulated as history-dependent stochastic

processes over a finite user vocabulary, and a causal dual-head Transformer was used to infer virality and log-final cascade size from short observed prefixes.

The empirical results show that Twitter cascade prediction is shaped by heavy-tailed cascade sizes, irregular inter-arrival times, sparse user-token observations, and limited train-test user overlap. These properties make exact user-level transition prediction unstable. The MiniTransformer did not improve over the first-order Markov baseline on next-user prediction, and Hits@K values remained extremely low, indicating that the bottleneck is sparse repeated user transitions rather than model capacity.

The main prediction results indicate that graph-pretrained user embeddings provide the most reliable gains when preserved as structural priors. The frozen node2vec Transformer achieved the strongest overall balance in AUROC, Brier score, and log-final-size RMSE across both the Transformer-variant ablation (Section 4.4) and the extended baseline comparison (Table 5). Within the Transformer-variant ablation, feature fusion with frozen embeddings achieved the highest AUPRC, suggesting that handcrafted descriptors mainly strengthen positive-class retrieval, whereas graph-derived embeddings improve global reliability and size estimation.

The calibration analysis further shows that ranking performance alone is insufficient for reliable evaluation. Models with similar AUROC may differ substantially in Brier score, calibration behavior, and regression stability. Reliable early cascade prediction therefore requires discrimination, retrieval quality, probability calibration, regression accuracy, and statistical uncertainty to be evaluated jointly.

Overall, the findings suggest that early Twitter cascade prediction is better understood as a graph-conditioned stochastic inference problem than as a purely sequential prediction task. Short prefixes provide limited temporal evidence, while social-graph priors supply structural information that stabilizes inference under sparse observations.

5.2. Discussion and Future Work

The results indicate that increasing sequence-model capacity is not sufficient for early cascade prediction. Under short-prefix conditions, a Transformer trained on supervised cascade data alone may not learn stable user-level dynamics. The advantage of frozen node2vec embeddings indicates that graph pretraining acts as a structural regularizer, preserving information about user proximity and visibility that is otherwise difficult to infer from sparse cascade prefixes.

Handcrafted features remain useful. Root-user visibility and early temporal activation statistics provide interpretable early-warning signals. Their contribution is complementary rather than dominant: feature fusion improves positive-class retrieval but does not consistently improve calibration or log-size regression.

Several limitations remain. The analysis is based on a single event-centered Twitter dataset, and the observation window is relatively short. The root-grouped cascade construction is reproducible but simplifies real diffusion, which may involve multiple sources, repeated exposures, recommendation mechanisms, and cross-community reinforcement. The graph prior is also static; node2vec does not capture temporal changes in user relations or activity. In particular, recurring multi-peak cascade dynamics arising from community clustering structure [21] and the multi-relational nature of social interaction [22] represent temporal and structural phenomena that a static single-layer graph prior cannot fully characterize. The Higgs dataset itself contains retweets, replies, mentions, and follows, whereas the present study uses only retweets and follows; a multilayer network framework would integrate these additional relation types and better reflect the full social interaction structure.

Future work should evaluate the framework across multiple platforms and event types, such as Weibo, Reddit, MemeTracker, or multi-event Twitter datasets. More expressive graph pretraining methods, including temporal graph neural networks, graph Transformers, and contrastive graph representations, may provide stronger priors for sparse users. Hybrid Hawkes–Transformer models could also combine interpretable temporal excitation with graph-conditioned neural representations. Additional uncertainty methods, such as conformal prediction, Bayesian ensembles, or distribution-shift-aware calibration, may further improve reliability in real-time cascade monitoring.

In summary, effective early cascade prediction requires the joint use of stochastic process formulation, graph-derived user priors, causal prefix encoding, and statistically reliable evaluation. Graph-conditioned representations offer a promising direction for modeling large-scale social diffusion under sparse and partial observations.

Author Contributions: B.D. and X.Z. contributed equally to this work. Conceptualization, B.D. and C.Y.; methodology, B.D., C.Y., W.Z. and L.H.; software, L.H. and X.Z.; validation, W.Z. and Y.F.; formal analysis, B.D. and X.Z.; investigation, L.H.; data curation, B.D.; writing—original draft preparation, B.D. and X.Z.; writing—review and editing, B.D. and X.Z.; visualization, X.Z.; project administration, C.Y.; funding acquisition, B.D., X.Z. and Y.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy restrictions and platform-policy constraints related to Twitter/X interaction records.

Acknowledgments: The authors are grateful to the reviewers for their constructive recommendations. Their perceptive questions and suggestions led to significant improvements in the methodological development and clarity of this work.

Conflicts of Interest: Weiyan Zhu was employed by Meta, United States. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. Meta had no role in the design of the study; in the collection, analysis, or interpretation of data; in the writing of the manuscript; or in the decision to submit the manuscript for publication.

Abbreviations

The following abbreviations are used in this manuscript:

Abbreviation	Full Term
AdamW	Adaptive Moment Estimation with Decoupled Weight Decay
AUROC	Area Under the Receiver Operating Characteristic Curve
AUPRC	Area Under the Precision–Recall Curve
BCE	Binary Cross-Entropy
CE	Cross-Entropy
CI	Confidence Interval
ECE	Expected Calibration Error
FFN	Feed-Forward Network
GNN	Graph Neural Network
LR	Logistic Regression
MHA	Multi-Head Attention
MLP	Multilayer Perceptron
OOV	Out-Of-Vocabulary
PPL	Perplexity
RF	Random Forest
RMSE	Root Mean Squared Error

ROC	Receiver Operating Characteristic
RT	Retweet
SE	Standard Error

Appendix A

Note. The tables in this appendix report supplementary statistical analysis for the six Transformer-variant ablation study corresponding to Figures 9–15 in the main text. All results are computed under the same experimental settings as the ablation (six embedding-strategy variants; prefix length $K = 5$ except where Table A2 varies K). The extended comparison including feature-based and additional neural baselines is reported in Tables 5 and 6 of Section 4, which follows a separate evaluation protocol; numerical results in the two evaluation sets are not directly comparable.

Appendix A.1. Pairwise AUROC Significance: DeLong’s Test

Table A1 reports pairwise DeLong’s test results for all eight classifiers evaluated on the test set. For each pair, the table lists the AUROC of each model, their difference ΔAUROC , the estimated standard error $\widehat{\text{SE}}$ of the difference, the two-sided Z-statistic, and the corresponding p -value. A difference is considered statistically significant at $\alpha = 0.05$ (marked *). Pairs that are not significant at this threshold are marked NS. Because all models are evaluated on the same test cascades, the DeLong estimator accounts for the positive correlation between paired predictions.

Table A1. Pairwise DeLong’s test for AUROC on the held-out test set. Model A is always the higher-AUROC variant. $\widehat{\text{SE}}$: estimated standard error of ΔAUROC . *: $p < 0.05$ (statistically significant at $\alpha = 0.05$); NS: not significant.

Model A	Model B	AUROC _A	AUROC _B	ΔAUROC	$\widehat{\text{SE}}$	Z	Sig.
$M_{n2v\text{-}frz}$	LR	0.775	0.748	+0.027	0.013	2.08	*
$M_{n2v\text{-}frz}$	M_{rand}	0.775	0.751	+0.024	0.012	2.00	*
$M_{n2v\text{-}frz}$	RF	0.775	0.762	+0.013	0.011	1.18	NS
$M_{n2v\text{-}frz}$	M_{n2v}	0.775	0.764	+0.011	0.010	1.10	NS
$M_{fus\text{-}n2v\text{-}frz}$	LR	0.771	0.748	+0.023	0.012	1.92	NS
$M_{n2v\text{-}frz}$	$M_{fus\text{-}n2v\text{-}frz}$	0.775	0.771	+0.004	0.009	0.44	NS
RF	LR	0.762	0.748	+0.014	0.012	1.17	NS
M_{n2v}	M_{rand}	0.764	0.751	+0.013	0.011	1.18	NS
M_{rand}	LR	0.751	0.748	+0.003	0.013	0.23	NS
$M_{fus\text{-}rand}$	M_{rand}	0.757	0.751	+0.006	0.010	0.60	NS
$M_{fus\text{-}n2v}$	M_{n2v}	0.769	0.764	+0.005	0.009	0.56	NS

Appendix A.2. Lead-Time Sensitivity at Varying Prefix Lengths

Table A2 evaluates how predictive performance changes as the observed prefix length K increases from 3 to 10 events. All other experimental settings follow Section 3.4. For virality classification, AUROC and AUPRC are reported; for log-final-size regression, RMSE on the log-scale target is reported (– indicates that the model does not produce a regression output). Metrics are computed on the held-out test set at each prefix length independently; bootstrap 95% confidence intervals (1000 replicates) are shown in brackets.

Table A2. Classification and regression performance at prefix lengths $K \in \{3, 5, 7, 10\}$. CI: 95% bootstrap confidence interval.

Model	K	AUROC [95% CI]	AUPRC [95% CI]	RMSE [95% CI]
$M_{n2v-frz}$	3	0.741 [0.698, 0.783]	0.588 [0.531, 0.642]	0.681 [0.641, 0.724]
	5	0.775 [0.736, 0.812]	0.621 [0.567, 0.674]	0.604 [0.568, 0.641]
	7	0.791 [0.754, 0.826]	0.639 [0.586, 0.690]	0.573 [0.538, 0.609]
	10	0.803 [0.768, 0.836]	0.652 [0.601, 0.703]	0.551 [0.517, 0.587]
M_{rand}	3	0.724 [0.680, 0.768]	0.568 [0.511, 0.624]	0.712 [0.671, 0.756]
	5	0.751 [0.710, 0.791]	0.601 [0.546, 0.655]	0.648 [0.609, 0.688]
	7	0.762 [0.722, 0.801]	0.614 [0.559, 0.667]	0.619 [0.580, 0.659]
	10	0.774 [0.736, 0.812]	0.623 [0.569, 0.677]	0.598 [0.560, 0.638]
$M_{fus-n2v-frz}$	3	0.738 [0.694, 0.781]	0.611 [0.554, 0.666]	0.694 [0.653, 0.737]
	5	0.771 [0.732, 0.809]	0.660 [0.607, 0.711]	0.617 [0.579, 0.656]
	7	0.786 [0.749, 0.822]	0.673 [0.621, 0.723]	0.587 [0.550, 0.625]
	10	0.798 [0.762, 0.832]	0.681 [0.630, 0.731]	0.563 [0.527, 0.601]
LR (baseline)	3	0.728 [0.683, 0.771]	0.571 [0.514, 0.627]	–
	5	0.748 [0.706, 0.789]	0.589 [0.534, 0.643]	–
	7	0.757 [0.716, 0.797]	0.601 [0.547, 0.654]	–
	10	0.768 [0.729, 0.807]	0.613 [0.559, 0.666]	–

Appendix A.3. Temperature Scaling Calibration Results

Table A3 reports calibration quality before and after temperature scaling for all six Transformer variants. The scalar temperature T^* is fitted by minimizing negative log-likelihood on the held-out test set. Expected calibration error (ECE, $B = 10$ bins) and Brier score are reported before calibration (raw) and after applying $p_{i,T^*} = \sigma(\eta_i/T^*)$. Values $T^* > 1$ indicate overconfidence in the raw outputs; values $T^* < 1$ indicate underconfidence. The ECE reduction $\Delta ECE = ECE_{raw} - ECE_{cal}$ and Brier reduction $\Delta Brier$ quantify the gain from post hoc calibration.

Table A3. Temperature scaling calibration results for all six Transformer variants. T^* : fitted scalar temperature. ΔECE and $\Delta Brier$: absolute reduction after calibration.

Model	T^*	ECE_{raw}	ECE_{cal}	ΔECE	$Brier_{raw}$	$\Delta Brier$
M_{rand}	1.31	0.094	0.048	0.046	0.203	0.007
M_{n2v}	1.19	0.082	0.043	0.039	0.198	0.007
$M_{n2v-frz}$	1.12	0.058	0.031	0.027	0.182	0.004
$M_{fus-rand}$	0.88	0.087	0.051	0.036	0.201	0.006
$M_{fus-n2v}$	0.97	0.071	0.038	0.033	0.195	0.006
$M_{fus-n2v-frz}$	0.94	0.076	0.041	0.035	0.191	0.005

Several patterns are noteworthy. First, randomly initialized variants (M_{rand} , $M_{fus-rand}$) exhibit the largest raw ECE values and require the strongest temperature correction, indicating that the absence of graph-structural priors is associated with less reliable probability outputs. Second, $M_{n2v-frz}$ achieves the lowest raw ECE (0.058) and the smallest absolute calibration gain, consistent with the Brier score reported in the main results: a well-initialized frozen prior naturally produces more calibrated logits. Third, feature-fusion variants ($T^* < 1$) are systematically underconfident in their raw outputs, reflecting the tendency of additive feature injection to push predicted scores toward the positive class and thereby underestimate low-probability events. Temperature scaling corrects this bias but does not fully eliminate the calibration gap relative to the frozen non-fusion baseline.

References

- Cheng, Z.; Zhou, F.; Xu, X.; Zhang, K.; Trajcevski, G.; Zhong, T.; Yu, P.S. Information Cascade Popularity Prediction via Probabilistic Diffusion. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 8541–8555. [\[CrossRef\]](#)
- Cheng, Z.; Liu, Y.; Zhong, T.; Zhang, K.; Zhou, F.; Yu, P.S. Disentangling Inter- and Intra-Cascades Dynamics for Information Diffusion Prediction. *IEEE Trans. Knowl. Data Eng.* **2025**, *37*, 4548–4563. [\[CrossRef\]](#)
- Dubovskaya, A.; Pena, C.B.; O’Sullivan, D.J.P. Modeling Diffusion in Networks with Communities: A Multitype Branching Process Approach. *Phys. Rev. E* **2024**, *111*, 034310. [\[CrossRef\]](#) [\[PubMed\]](#)
- Jing, X.; Jing, Y.; Lu, Y.; Deng, B.; Yang, S.; Yang, D. On Your Mark, Get Set, Predict! Modeling Continuous-Time Dynamics of Cascades for Information Popularity Prediction. *IEEE Trans. Knowl. Data Eng.* **2024**, *37*, 5436–5451. [\[CrossRef\]](#)
- Kashuv, Y.; Alharbi, R.; Thai, M.T. Predicting User Tipping in Online Social Networks with Temporal Graph Neural Networks. *IEEE Trans. Comput. Soc. Syst.* **2026**, *13*, 1228–1240. [\[CrossRef\]](#)
- Li, H.; Xia, C.; Wang, T.; Wang, Z.; Cui, P.; Li, X. GRASS: Learning Spatial–Temporal Properties from Chainlike Cascade Data for Microscopic Diffusion Prediction. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 16313–16327. [\[CrossRef\]](#) [\[PubMed\]](#)
- Li, L.; Chen, Z.J.; Ye, H.; Zhang, Y. Incorporating Topical Stance into Signed Bipartite Networks for User Retweet Prediction. *PLoS ONE* **2026**, *21*, e0342677. [\[CrossRef\]](#) [\[PubMed\]](#)
- Li, L.; Duan, L.; Wang, J.; He, C.; Chen, Z.; Xie, G.; Deng, S.; Luo, Z. Memory-Enhanced Transformer for Representation Learning on Temporal Heterogeneous Graphs. *Data Sci. Eng.* **2023**, *8*, 98–111. [\[CrossRef\]](#)
- Liu, C.; Zhang, J.; Wang, S.; Fan, W.; Li, Q. Score-Based Generative Diffusion Models for Social Recommendations. *IEEE Trans. Knowl. Data Eng.* **2024**, *37*, 6666–6679. [\[CrossRef\]](#)
- Liu, X.; Wang, H.; Bouyer, A. A Cascade Information Diffusion Prediction Model Integrating Topic Features and Cross-Attention. *J. King Saud Univ. Comput. Inf. Sci.* **2023**, *35*, 101852. [\[CrossRef\]](#)
- Mashayekhi, Y.; Rezvanian, A.; Vahidipour, S. A Novel Regularized Weighted Estimation Method for Information Diffusion Prediction in Social Networks. *Appl. Netw. Sci.* **2023**, *8*, 81. [\[CrossRef\]](#)
- Peng, H.; Zhang, J.; Huang, X.; Hao, Z.; Li, A.; Yu, Z.; Yu, P.S. Unsupervised Social Bot Detection via Structural Information Theory. *ACM Trans. Inf. Syst.* **2024**, *42*, 148. [\[CrossRef\]](#)
- Qi, O.; Chen, H.; Liu, S.; Pu, L.; Ge, D.; Fan, K. DMHANT: DropMessage Hypergraph Attention Network for Information Propagation Prediction. *Big Data* **2024**, *13*, 364–378. [\[CrossRef\]](#) [\[PubMed\]](#)
- Sallah, A.; Abdellaoui Alaoui, E.A.; Agoujl, S.; Wani, M.A.; Hammad, M.; Maleh, Y.; Abd El-Latif, A.A. Fine-Tuned Understanding: Enhancing Social Bot Detection with Transformer-Based Classification. *IEEE Access* **2024**, *12*, 118250–118269. [\[CrossRef\]](#)
- Tai, Y.; Yang, H.; He, H.; Wu, X.; Shao, Y.; Zhang, W.; Sangaiah, A.K. Topic-Aware Masked Attentive Network for Information Cascade Prediction. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2024**, *23*, 126. [\[CrossRef\]](#)
- De Domenico, M.; Lima, A.; Mougél, P.; Musolesi, M. The Anatomy of a Scientific Rumor. *Sci. Rep.* **2013**, *3*, 2980. [\[CrossRef\]](#) [\[PubMed\]](#)
- Tang, Y.; Piao, J.; Wang, H.; Wang, Y.; Li, Y. MSA-Net: A Multi-Scale Information Diffusion Model Awaring User Activity Level. *ACM Trans. Web* **2025**, *19*, 17. [\[CrossRef\]](#)
- Vinod, D.; Kumar, T. G.; Kumar, P.N. Effects of the Evolution of Network Structural Properties on Information Diffusion in Dynamic Social Networks. *J. Intell. Fuzzy Syst.* **2025**, *49*, 611–626. [\[CrossRef\]](#)
- Wang, Z.; Wang, X.; Xiong, F.; Chen, H. A Survey of Deep Learning-Based Information Cascade Prediction. *Symmetry* **2024**, *16*, 1436. [\[CrossRef\]](#)
- Wang, B.; Li, Z.; Xu, Z.; Zhang, J. Casformer: Information Popularity Prediction with Adaptive Cascade Sampling and Graph Transformer in Social Networks. *IEEE Trans. Big Data* **2025**, *11*, 1652–1663. [\[CrossRef\]](#)
- Almanza, M.; Lattanzi, S.; Panconesi, A.; Re, G. Twin Peaks, a Model for Recurring Cascades. In Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021; pp. 681–692. [\[CrossRef\]](#)
- Bonifazi, G.; Cauteruccio, F.; Corradini, E.; Giannelli, E.; Marchetti, M.; Ursino, D.; Virgili, L. A Multilayer Network-Based Framework for Investigating the Evolution and Resilience of Multimodal Social Networks. *Soc. Netw. Anal. Min.* **2023**, *14*, 5. [\[CrossRef\]](#)
- Ye, J.; Bao, Q.; Xu, M.; Xu, J.; Qiu, H.; Jiao, P. RD-GCN: A Role-Based Dynamic Graph Convolutional Network for Information Diffusion Prediction. *IEEE Trans. Netw. Sci. Eng.* **2024**, *11*, 4923–4937. [\[CrossRef\]](#)
- Zeng, Y.; Xiang, K. Persistence Augmented Graph Convolution Network for Information Popularity Prediction. *IEEE Trans. Netw. Sci. Eng.* **2023**, *10*, 3331–3342. [\[CrossRef\]](#)
- Perozzi, B.; Al-Rfou, R.; Skiena, S. DeepWalk: Online Learning of Social Representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 701–710. [\[CrossRef\]](#)

26. Grover, A.; Leskovec, J. node2vec: Scalable Feature Learning for Networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 855–864. [[CrossRef](#)] [[PubMed](#)]
27. Zhai, P.; Yang, Y.; Zhang, C.H. Causality-Based CTR Prediction Using Graph Neural Networks. *Inf. Process. Manag.* **2023**, *60*, 103137. [[CrossRef](#)]
28. Zhang, Y.; Wang, Z.; Zhuang, H.; Song, L.; Wen, G.; Guan, J.; Zhou, S. Predicting Participation Shift of Users at the Next Stage in Social Networks. *IEEE Trans. Netw. Sci. Eng.* **2025**, *12*, 1066–1079. [[CrossRef](#)]
29. Zhang, G.; Zhang, S.; Yuan, G. Bayesian Graph Local Extrema Convolution with Long-Tail Strategy for Misinformation Detection. *ACM Trans. Knowl. Discov. Data* **2024**, *18*, 89. [[CrossRef](#)]
30. Zhao, J.; Lyu, X.; Rong, H.; Zhao, J. TRGCN: A Prediction Model for Information Diffusion Based on Transformer and Relational Graph Convolutional Network. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2024**, *23*, 143. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.