FUTUREWAY: ADAPTIVE FUTURE-INFORMED WAYPOINTS FOR LONG-HORIZON DIFFUSION PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-horizon diffusion prediction aims to forecast how information spreads on social networks, which requires resolving local ambiguity and correcting trajectory drift during autoregressive rollout. Yet existing methods primarily learn next-participant prediction on ground-truth histories, leaving them poorly equipped to recover from rollout errors. To address this challenge, we propose FUTUREWAY, a framework that turns recorded futures into decision-specific guidance on model-generated histories. Future information can help distinguish locally plausible participants and correct decisions after rollout deviation, but the useful guidance horizon depends on the current diffusion state. During training, multi-horizon future-informed teachers provide complementary corrective signals. A prefix-only projection analysis characterizes the optimal approximation to future-informed guidance using observable history, motivating student-aware horizon weighting that balances corrections beyond a pretrained anchor against compatibility with the evolving student. Representation-aware on-policy self-distillation transfers this guidance into student decisions and representations. At inference, FUTUREWAY uses only the current cascade prefix and social graph without future information. Experiments on three real-world benchmarks show consistent improvements in both long-horizon future-participant coverage and order-sensitive trajectory recovery, with an average relative gain of 10.56% over the strongest comparators.

1 INTRODUCTION

Long-horizon diffusion prediction aims to forecast the future trajectory of an information cascade (*i.e.*, the sequence of users who become active over time) from an observed prefix and the underlying social graph. Accurate long-horizon forecasts can support timely decisions in applications such as viral marketing and misinformation intervention (Kempe et al., 2015; Bak-Coleman et al., 2022). Existing diffusion prediction methods are primarily optimized for *next-participant prediction*: given a social graph and the observed cascade history, they predict which user will become active next (Sun et al., 2022; Qiao et al., 2023; Zhong et al., 2024; Shao et al., 2025). Accordingly, a long-horizon trajectory is typically obtained by repeatedly applying such a predictor autoregressively, appending each predicted user to the history and using the extended sequence to predict the next one.

Information diffusion on social networks is shaped by various factors, such as social relations and user interests. The same observed prefix often admits several locally plausible next participants, while using only the observed cascade makes it difficult to identify exactly which one is correct. Prediction errors are consequently difficult to avoid. During multi-step autoregressive forecasting, however, an early error can affect more than a single prediction: once an incorrect participant is appended to the history, it influences subsequent decisions, potentially reinforcing the deviation and steering the predicted rollout toward a different trajectory or community. Figure 1 illustrates this progressive divergence from a shared observed prefix. Figure 2(a) further shows that Hit@10 drops from $\mathbf{36}\%$ at the first prediction step to only $\mathbf{7}\%$ by step 15 as the model continuously conditions on its own generated history. A key limitation lies in how existing predictors are trained. Most methods optimize *local next-step prediction* on *ground-truth prefixes*, which provides limited supervision for long-horizon multi-step trajectories and for the model's own generated, deviated prefixes. Long-horizon forecasting therefore calls for two key capabilities: (1) *stronger local discrimination* to identify the correct participant among several similarly plausible candidates; and (2) *stronger global correction* to recover subsequent predictions after an early mistake and limit further drift.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

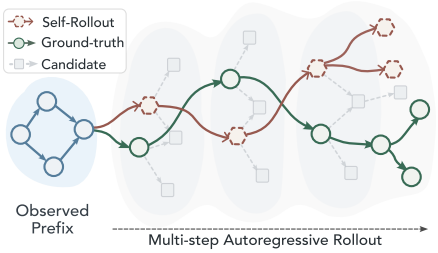


Figure 1: At each step, the same prefix may support several locally plausible users. Once the rollout takes a wrong branch, accumulated errors over time cause the trajectory to diverge.

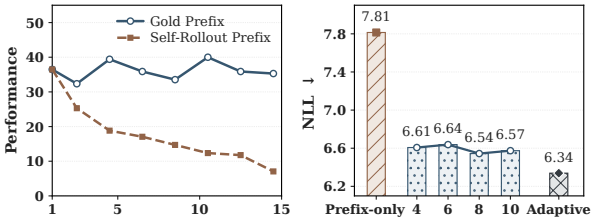


Figure 2: (a) Autoregressive accuracy consistently declined over the rollout steps. (b) Future-informed guidance can consistently lower next-participant negative log-likelihood (NLL; lower is better), with oracle adaptive guidance providing a further benefit.

These observations motivate a different perspective: If the model can proactively understand future diffusion waypoints, such future information can serve as *discriminative guidance* when several next participants are locally plausible, and as *corrective guidance* when earlier prediction errors have already shifted the rollout away from the ground-truth trajectory. Empirically, Figure 2(b) shows that on model-generated prefixes (most already deviate from the true trajectory), incorporating future information reduces the average negative log-likelihood (NLL) of the next participant from 7.81 without future information to 6.54–6.64 across different fixed future horizons. The results suggest that future information provides useful signals for resolving local ambiguity after rollout deviation.

A natural way to exploit recorded futures during training is to use multiple future participants as labels (Feng et al., 2026). In this formulation, future participants serve as *prediction targets*, specifying which users should be predicted as an unordered set. This differs from using future information as *decision-specific guidance* to determine which participant should be selected next given the history currently generated by the model. This distinction matters in two respects. First, supervising multiple future targets on ground-truth prefixes does not explicitly teach the model how to make decisions on its own generated, potentially deviated histories. Second, treating multiple future participants as positive targets encourages coverage of future participants rather than discrimination of the exact next participant. Such supervision therefore does not explicitly indicate which locally plausible candidate should be selected, especially on a biased trajectory generated by the model.

The key requirement is therefore not simply to provide more future labels, but to use future information as decision-specific guidance *at the states the model actually encounters during its own rollout*. Yet even when future guidance is provided on these generated states, a second question remains: *how far into the future should the model look?* The usefulness of future information naturally depends on the current diffusion state. For example, near a community transition, a short horizon may mainly reveal participants from the current community, whereas a longer horizon can expose the next community to become active. Looking too far ahead, however, may introduce additional transitions that are less relevant to the current decision. Different horizons may indicate different correction signals for the same generated history. The *Adaptive* oracle in Figure 2(b) further validates this state-dependent headroom, achieving a lower NLL than the best fixed horizon. Effective future guidance must therefore determine not only *where* to provide guidance—on the model's own generated histories—but also *which* future horizon should guide each decision as diffusion progresses.

We introduce FUTUREWAY, a framework that turns recorded futures into decision-specific guidance for local discrimination and trajectory correction. During training, *multi-horizon future-informed teachers* guide next-participant decisions and recovery on the predictor's own generated histories. Our *prefix-only projection analysis* characterizes the optimal prefix-only approximation to this guidance, separating transferable supervision from irreducible future dependence and explaining state-dependent horizon utility. Motivated by this analysis, *student-aware horizon weighting* adapts guidance to the state and evolving student, balancing guidance novelty against student compatibility. *Representation-aware on-policy self-distillation* transfers this guidance into both decisions and representations used for horizon weighting. Successive rounds refresh rollout histories and the student comparison distribution, adapting both supervision and its assessment. These components couple *where to learn*, *which future guidance to use*, and *how to transfer it*, addressing the mismatch between training on ground-truth prefixes and forecasting on self-generated histories. Inference preserves the original backbone and requires only the current cascade and social graph. Experiments on three real-world benchmarks consistently improve future-participant coverage and order-sensitive trajectory recovery, with an average relative gain of $\mathbf{10.56}\%$ over the strongest comparators.

2 PROBLEM STATEMENT

Let $G = (V, E)$ denote a social graph, where V is the set of users and E represents social relations. An information cascade is a temporally ordered sequence $C^\star = ((v_1^\star, \tau_1^\star), \ldots, (v_T^\star, \tau_T^\star))$, where $v_i^\star$ denotes the i-th participating user and $\tau_i^\star$ is the corresponding participation time, with $\tau_1^\star \leq \cdots \leq \tau_T^\star$. For readability, we omit timestamps from subsequent notation when they are not essential and write $C^\star = (v_1^\star, \ldots, v_T^\star)$; temporal information remains part of the observed cascade.

Long-Horizon Diffusion Prediction. Conventional information diffusion prediction is typically formulated as next-participant prediction: given an observed prefix $C_t^\star = (v_1^\star, \ldots, v_t^\star)$ and the social graph G, a predictor $p_\theta(\cdot \mid C_t^\star, G)$ estimates which user will participate next using only information available up to the current step (Sun et al., 2022; Qiao et al., 2023; Zhong et al., 2024; He et al., 2025). During rollout, the current prefix includes previously generated participants.

Long-horizon diffusion prediction extends this task to forecasting a sequence of future participants. Given an observed prefix $C_t^\star$ with $1 \leq t < T$, let H denote a configurable prediction budget, measured in future participation steps. The target continuation is $(v_{t+1}^\star, \ldots, v_{t+H}^\star)$, where $1 \leq H \leq T - t$. Starting from $\widehat{C}_t = C_t^\star$, the predictor autoregressively generates, for $s = t, \ldots, t + H - 1$:

$$\widehat{v}_{s+1} = \mathrm{Decode}\Big(p_\theta(\cdot \mid \widehat{C}_s, G)\Big), \qquad \widehat{C}_{s+1} = \widehat{C}_s \oplus \widehat{v}_{s+1}, \tag{1}$$

where Decode selects a participant from the predicted distribution and $\oplus$ appends that participant to the history. Thus, $\widehat{C}_s$ consists of the observed prefix followed by the model's own predictions. The resulting forecast $(\widehat{v}_{t+1}, \ldots, \widehat{v}_{t+H})$ aims to recover both future participants and their participation order. An early error therefore affects not only the current step but also the context for every subsequent decision, potentially steering the rollout toward a different trajectory or community.

Standard Next-Participant Training. Standard diffusion predictors are typically trained on observed cascade histories using the actual next participant as supervision:

$$\mathcal{L}_{\mathrm{NTP}}(\theta) = -\mathbb{E}_{C^\star, t}\left[\log p_\theta\left(v_{t+1}^\star \mid C_t^\star, G\right)\right]. \tag{2}$$

The expectation is over observed cascades and valid prefix lengths $1 \leq t < T$. This objective supervises individual next-participant decisions on *ground-truth histories*, whereas autoregressive inference conditions on histories increasingly composed of the model's own predictions. It neither directly optimizes the resulting multi-step trajectory nor teaches recovery from errors in these generated histories. When several next participants are locally plausible, an incorrect choice can therefore propagate through subsequent predictions. Effective long-horizon forecasting calls for both *local discrimination*, to distinguish the correct next participant among plausible candidates, and *global correction*, to recover subsequent predictions after an early mistake and limit further trajectory drift.

3 RELATED WORK

Microscopic diffusion prediction has primarily been formulated as *next-participant prediction*. Existing methods mainly improve cascade representations through hypergraph learning, disentanglement, contrastive objectives, geometric dynamics, retrieval, or scalable graph computation (Yuan et al., 2021; Sun et al., 2022; Cheng et al., 2023; Jiao et al., 2024; Shao et al., 2025; Qiao et al., 2023; Wang et al., 2024; Zhong et al., 2024; Feng et al., 2025). They retain the one-step objective of predicting the next participant from an observed history and achieve improved performance. A second line incorporates final cascade size, *i.e.*, popularity prediction, as a macro-level auxiliary objective (Yang et al., 2019; Jiao et al., 2024). Both signals can improve cascade representations, but do not provide multi-step trajectory evolution guidance for individual decisions during rollout. More recently, GRID directly predicts the set of future participants from an observed prefix using multi-label ranking supervision (Feng et al., 2026). This is a different prediction problem: GRID performs one-shot, order-free set prediction from a ground-truth prefix, without modeling the intermediate trajectory. Our setting instead predicts step by step on the model's own generated histories, where each decision changes the state encountered next. Thus, multi-label targets defined on ground-truth prefixes cannot supervise how the model should act after rollout deviation. The key distinction of FUTUREWAY is therefore not simply to use more future information, but to use future evolution as *decision-level corrective supervision*, which enables the model to learn stronger local discrimination and global correction capabilities without future information at inference. Appendix D provides more details about diffusion prediction and knowledge distillation techniques.

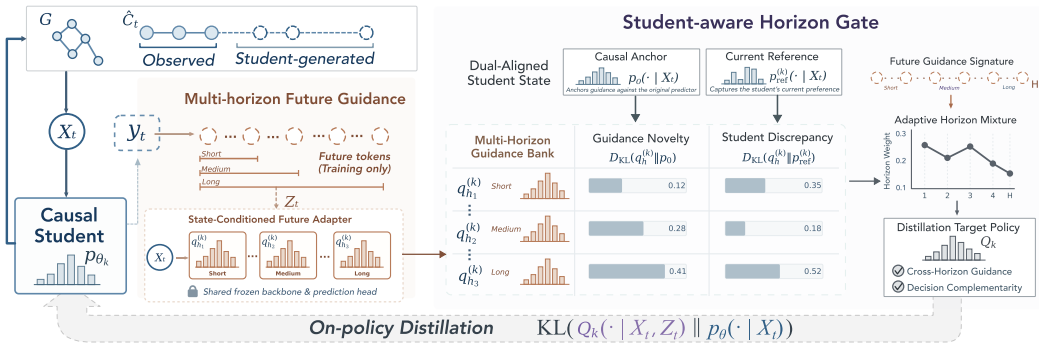


Figure 3: **Overview of FUTUREWAY**: At step t, the student autoregressively extends the observed cascade using the social graph, forming a diffusion state X_t. The student is *prefix-only*: its predictions depend only on the current cascade prefix and social graph, without access to future participants. We construct a corrective target y_t and future windows Z_t from the ground-truth cascade. Teachers with different window lengths provide multi-horizon future-informed guidance. We compare each teacher's guidance with a fixed pretrained anchor and a student comparison distribution obtained by decoding the round-start student representation through the frozen pretrained prediction head. Adaptive weights favor guidance that differs from the anchor while better matching the student comparison distribution, balancing *guidance novelty* and *student compatibility*. The teacher mixture supervises student predictions and representations; only the student is retained at inference.

4 METHODOLOGY

We first build a commonly used GNN–Transformer prediction backbone (Sun et al., 2022; He et al., 2025). Given a state $X = (C, G)$ consisting of the current cascade prefix C and social graph G, a GNN first encodes G into structure-aware user embeddings. A causal Transformer (Vaswani et al., 2017) then processes the participant embeddings in C in cascade order to obtain the cascade representation $\mathbf{s}_\theta(X)$, which a prediction head maps to the next-participant distribution $p_\theta(\cdot \mid X)$. Here, θ comprises the trainable backbone and prediction-head parameters. The pretrained parameters $\bar{\theta}$ are obtained using the objective in Eq. (2). Appendix B.1 provides a notation table for reference.

4.1 MULTI-HORIZON FUTURE-INFORMED TEACHERS

As discussed above, FUTUREWAY uses future information to guide local next-participant decisions and longer-term correction at model-generated states. We therefore construct future-informed teachers that operate on states visited by the student and additionally access ground-truth future information during training. Each teacher refines the prefix-only prediction at the same state, using subsequent ground-truth participants to guide the next-participant distribution. The *student* learns from this guidance using only the current cascade prefix and social graph.

Consider a positive integer horizon h, which controls the amount of future information available to the teacher. For a model-generated state $X_t = (\widehat{C}_t, G)$ associated with a ground-truth cascade $C^\star$, we first remove participants already present in $\widehat{C}_t$ from $C^\star$ while preserving the order of the remaining participants, yielding $R_t = (r_{t,0}, \ldots, r_{t,L_t-1})$. When $L_t > 0$, we define $y_t = r_{t,0}$ as the *corrective target*, *i.e.*, the earliest ground-truth participant not yet present in the generated cascade. This choice follows the ground-truth order and provides consistent re-entry supervision at model-generated states. For a ground-truth prefix, the same construction reduces to the ordinary next-participant target. The subsequent h remaining participants form the future window:

$$Z_{t,h} = (r_{t,1}, \ldots, r_{t,h}), \qquad L_t \geq h + 1. \tag{3}$$

Thus, y_t specifies the immediate corrective target that is excluded from teacher input to avoid label leakage, while $Z_{t,h}$ provides subsequent ground-truth context for that decision.

Given $(X_t, Z_{t,h})$, the teacher injects this future information through a *State-conditioned Attention Adapter*. Let $\bar{\mathbf{H}}_t$ contain the prefix representations produced by the frozen pretrained backbone for X_t, arranged as columns, and let $\bar{\mathbf{s}}_t = \mathbf{s}_{\bar{\theta}}(X_t)$ denote its prediction representation. The columns of $\bar{\mathbf{E}}(Z_{t,h})$ contain the future-participant embeddings in the same frozen user embedding space. Using

standard scaled dot-product attention (Vaswani et al., 2017), we compute

$$\mathbf{s}_{t,h}^{\mathrm{T}} = F_h\Big(\mathbf{u}_{t,h} + \mathrm{Attn}\Big(W_h^Q\mathbf{u}_{t,h},\, W_h^K\mathbf{M}_{t,h},\, W_h^V\mathbf{M}_{t,h}\Big)\Big). \tag{4}$$

Here, $\mathbf{u}_{t,h} = \mathbf{u}_h^{(0)} + U_h\bar{\mathbf{s}}_t$ is a state-conditioned query and $\mathbf{M}_{t,h} = [\,\bar{\mathbf{H}}_t\ \bar{\mathbf{E}}(Z_{t,h})\,]$ contains both prefix and future-participant representations (both with learnable temporal encodings). $\mathbf{u}_h^{(0)}$, U_h, W_h^Q, W_h^K, and W_h^V are learned parameters, and F_h is a learned map to the prediction representation space. The resulting teacher distribution is $q_h(\cdot \mid X_t, Z_{t,h}) = \mathrm{softmax}(\bar{W}\mathbf{s}_{t,h}^{\mathrm{T}})$, where $\bar{W}$ is the frozen pretrained prediction head. Conditioning the query on $\bar{\mathbf{s}}_t$ lets the teacher interpret future information in the current prediction context.

For each training round, we collect aligned samples from both ground-truth prefixes and model-generated histories produced by the frozen student snapshot. Each horizon-specific adapter is trained to predict the corrective target from the current state and its future window: $\mathcal{L}_{\mathrm{T},h} = -\mathbb{E}_{(X,Y,Z_h)}\big[\log q_h(Y \mid X, Z_h)\big]$. For efficiency, only the lightweight adapter parameters are updated; the pretrained backbone, user embeddings, and prediction head remain frozen. Training directly on model-generated histories exposes the teacher to states that deviate from ground-truth prefixes and require corrective guidance. Adapter parameters are retained across rounds, and the updated teacher $q_h^{(k)}$ is frozen during student optimization in round k.

Figure 2 shows that different horizons can provide complementary corrective signals. We therefore construct teachers for a finite horizon set $\mathcal{H}$, sharing the frozen backbone, user embeddings, and prediction head while maintaining a separate lightweight adapter for each horizon. For an aligned pair $(X_t, C^\star)$, the available teachers are indexed by $\mathcal{H}_t = \{h \in \mathcal{H} : L_t \geq h+1\}$.

4.2 Prefix-Only Projection of Future Guidance

The multi-horizon teachers use future information that is unavailable to the student at prediction time. This raises two questions: **(Q1)** *what part of future-informed guidance can be transferred to a prefix-only predictor?* and **(Q2)** *why should the useful future horizon depend on the current state?* We address both before introducing student-aware horizon weighting.

For **Q1**, consider a future-guidance distribution $Q(\cdot \mid X, Z)$, where X denotes the current prefix state and Z collects the available future windows. Although a prefix-only predictor $p(\cdot \mid X)$ cannot reproduce every future-conditioned prediction, forward-KL distillation admits an exact population projection that is related to the conditional-mean property of Bregman losses (Banerjee et al., 2005).

Theorem 1. *Fix a distribution ν over aligned pairs (X, Z), with state marginal ν_X, and a future-guidance distribution $Q(\cdot \mid X, Z)$ over participants in V. Let $\mathcal{R}_Q(p) = \mathbb{E}_\nu[D_{\mathrm{KL}}(Q(\cdot \mid X, Z)\|p(\cdot \mid X))]$, $A \mid X, Z \sim Q(\cdot \mid X, Z)$, and $p_Q^\star(a \mid X) = \mathbb{E}_\nu[Q(a \mid X, Z) \mid X]$. Then, for every prefix-only predictor p with finite risk,*

$$\mathcal{R}_Q(p) = \underbrace{I_Q(A; Z \mid X)}_{\textit{irreducible future dependence}} + \underbrace{\mathbb{E}_{\nu_X} D_{\mathrm{KL}}\big(p_Q^\star(\cdot \mid X)\|p(\cdot \mid X)\big)}_{\textit{prefix-only approximation gap}}, \tag{5}$$

where I_Q is computed under $\nu(X, Z)Q(A \mid X, Z)$. Consequently, $p_Q^\star$ is the unique population optimum over unrestricted prefix-only distributions up to ν_X-null sets.

The proof is provided in Appendix B.3. Theorem 1 decomposes the distillation risk into irreducible future dependence and a discrepancy from the *prefix-only projection*. Thus, distillation does not require the student to recover Z; its population objective targets the conditional average of future-informed predictions. This projection has a direct statistical interpretation. If $\nu = P_{X,Z}$ is the marginal of the aligned-data distribution P and the teacher is ideal, $Q(a \mid X, Z) = P(Y = a \mid X, Z)$, we obtain $p_Q^\star(a \mid X) = \mathbb{E}_{P(Z|X)}[P(Y = a \mid X, Z)] = P(Y = a \mid X)$. Hence, future observations can sharpen the training signal while the optimal projection depends only on X.

For **Q2**, learned teachers with different horizons can induce different prefix-only projections. For a fixed round, restrict aligned samples to common support where every $h \in \mathcal{H}$ is available; all expectations below use this restricted distribution. Suppressing the round index, let $p_h^\star(\cdot \mid X) = \mathbb{E}[q_h(\cdot \mid X, Z_h) \mid X]$ for each $h \in \mathcal{H}$. For two prefix-only predictors p_a and p_b with finite distillation risks for every horizon, define the horizon-specific *projected transfer gain* $\Delta_h(X) = D_{\mathrm{KL}}(p_h^\star\|p_a) -$

$D_{\mathrm{KL}}(p_h^\star \| p_b)$. A positive value means that the change from p_a to p_b improves agreement with the prefix-only projection induced by horizon h. Applying Theorem 1 conditionally on X gives

$$\Delta_h(X) = \mathbb{E}_{Z_h|X}[D_{\mathrm{KL}}(q_h \| p_a) - D_{\mathrm{KL}}(q_h \| p_b)], \tag{6}$$

because the horizon-specific irreducible future dependence cancels. Thus, the realized KL difference is conditionally unbiased for the projected transfer gain. This characterization also explains why a single globally fixed horizon can be insufficient. For finite $\mathcal{H}$, if no single horizon maximizes $\Delta_h(X)$ almost surely, then $\mathbb{E}_X[\max_h \Delta_h(X)] > \max_h \mathbb{E}_X[\Delta_h(X)]$: the horizon with the largest projected transfer gain varies across states. This analysis is an exact characterization of prefix-only distillation and a conditional oracle comparison that motivates state-dependent horizon weighting.

4.3 Student-Aware Horizon Weighting

The analysis above motivates comparing teacher guidance with both the pretrained predictor and the current student representation. We therefore use two prefix-only distributions: a fixed pretrained anchor and a student comparison distribution that isolates representation changes. Their divergences from each teacher serve as proxies for *guidance novelty* and *student compatibility*.

Distillation proceeds in rounds with $\theta_0 = \bar{\theta}$. At the beginning of round k, we freeze the student snapshot θ_{k-1} for rollout collection, teacher-adapter updates, and guidance construction. Let $p_0 = p_{\bar{\theta}}$ denote the pretrained anchor. To isolate representation changes, we decode student representations using the frozen pretrained head $\bar{W}$. For any student parameters θ, define

$$p_{\mathrm{ref}}^{\theta}(\cdot \mid X) = \mathrm{softmax}\big(\bar{W}\mathbf{s}_\theta(X)\big), \qquad p_{\mathrm{ref}}^{(k)} = p_{\mathrm{ref}}^{\theta_{k-1}}. \tag{7}$$

Here, the subscript "ref" denotes the student comparison distribution obtained using the frozen pretrained head; it does not denote a ground-truth distribution. Thus, the difference between $p_{\mathrm{ref}}^{(k)}$ and p_0 reflects representation change relative to pretraining under the same prediction head. For each horizon $h \in \mathcal{H}_t$, we define its *student-aware transfer score*

$$g_{t,h}^{(k)} = D_{\mathrm{KL}}\Big(q_h^{(k)} \| p_0\Big) - D_{\mathrm{KL}}\Big(q_h^{(k)} \| p_{\mathrm{ref}}^{(k)}\Big), \tag{8}$$

with the teacher evaluated at $(X_t, Z_{t,h})$ and both prefix-only distributions at X_t. The score is positive precisely when the student comparison distribution is closer to the teacher in forward KL than the pretrained anchor is. Equivalently, $g_{t,h}^{(k)} = \mathbb{E}_{a \sim q_h^{(k)}}[\log p_{\mathrm{ref}}^{(k)}(a \mid X_t) - \log p_0(a \mid X_t)]$, so teacher entropy cancels and the score measures relative fit to the teacher under the same prediction head.

By Eq. (6), the score is an observable counterpart of the projected transfer gain. Under the common-support distribution defined above, $\mathbb{E}_{Z_h|X}[g_h^{(k)}] = D_{\mathrm{KL}}(p_h^\star \| p_0) - D_{\mathrm{KL}}(p_h^\star \| p_{\mathrm{ref}}^{(k)})$. Here, $g_h^{(k)}$ denotes the score as a function of (X, Z_h). Its conditional expectation measures the current student representation's relative fit to the horizon-specific projection.

Because $g_{t,h}^{(k)}$ is computed from a realized future window, aggressively selecting the largest score can be sensitive to sample-level variation. Moreover, our empirical analysis shows that different horizons share substantial global structure while retaining complementary top-ranked predictions (Appendix A.4). We therefore retain a uniform component and use the student-aware scores to modulate the remaining guidance. For $\mathcal{H}_t \neq \varnothing$,

$$w_{t,h}^{(k)} = \frac{1-\rho}{|\mathcal{H}_t|} + \rho \frac{\exp(g_{t,h}^{(k)}/\tau_s)}{\sum_{j \in \mathcal{H}_t} \exp(g_{t,j}^{(k)}/\tau_s)}, \qquad Q_k = \sum_{h \in \mathcal{H}_t} w_{t,h}^{(k)} q_h^{(k)}. \tag{9}$$

Here, $\rho \in [0,1]$ controls how strongly the guidance departs from uniform aggregation and $\tau_s > 0$ controls the concentration of the adaptive component. In the first round, $p_{\mathrm{ref}}^{(1)} = p_0$, so all scores are zero and all available horizons receive equal weights, $w_{t,h}^{(1)} = 1/|\mathcal{H}_t|$. Let $Z_t = \{Z_{t,h}\}_{h \in \mathcal{H}_t}$ collect the available windows; Q_k depends on both X_t and Z_t. This weighting rule requires no additional learned parameters and is used only during training; inference uses only the prefix-only student.

4.4 Representation-Aware On-Policy Self-Distillation

Based on the future-informed policy Q_k, we can guide the student on model-generated states where long-horizon errors accumulate. We therefore use roundwise on-policy distillation. Here, self-distillation transfers guidance from adapters of *the same backbone*; rollout states are collected using

the student snapshot at the beginning of each round and held fixed within that round. In round k, the frozen snapshot θ_{k-1} induces a distribution ν_k over rollout tuples (X, Y, Z) aligned with the ground-truth cascade, restricted to states with $\mathcal{H}_t \neq \varnothing$ for the loss below.

The student is initialized from θ_{k-1} and optimized with corrective supervision and two complementary knowledge-distillation (KD) signals. Matching Q_k through p_θ updates the full predictor, whereas matching it through p^θ_{ref} supervises the representation through the frozen head $\bar{W}$. This shapes the representation used by horizon weighting without requiring hidden-state matching:

$$\mathcal{L}^{(k)}_{\text{OPSD}}(\theta) = \mathbb{E}_{(X,Y,Z)\sim\nu_k}\Big[-\log p_\theta(Y \mid X) + \lambda D_{\text{KL}}(\text{sg}[Q_k]\|p_\theta) + \mu D_{\text{KL}}(\text{sg}[Q_k]\|p^\theta_{\text{ref}})\Big]. \quad (10)$$

Here, λ and μ are the nonnegative policy- and representation-distillation coefficients, respectively; sg denotes stop-gradient, and Q_k is fixed during student optimization. Conditioning arguments are omitted inside the KL terms. After obtaining θ_k, we refresh the rollouts, update the adapters, and reconstruct the student comparison distribution and guidance for the next round. Both teacher guidance and horizon weights thus track the evolving student distribution, while the pretrained anchor p_0 remains fixed throughout training. At inference, only the prefix-only predictor $p_\theta(\cdot \mid \widehat{C}_t, G)$ is retained. Any future information is used only during training; inference requires only the current cascade prefix and social graph. Algorithm 4 summarizes the training and inference workflow.

5 Experiments

Datasets. As summarized in Table 1, we evaluate on three real-world information diffusion datasets (Twitter, Weibo, and Quora), each containing timestamped cascades and an associated social graph. Cascades are chronologically partitioned into TRAIN, VALID, and TEST sets with an 8:1:1 ratio. All data partitions and user vocabularies are fixed across random seeds and methods. Appendix E.1 provides further dataset details.

Table 1: Dataset statistics.

Dataset	Users	Links	Cascades	Avg. Len.
Quora	4,578	55,698	1,267	34.60
Twitter	12,627	619,262	3,435	32.64
Weibo	39,043	369,920	7,963	147.74

Evaluation Protocol. For a cascade of length T, we observe the first $\max(1, \lfloor 0.3T \rfloor)$ participants and predict the subsequent trajectory in a budget-constrained greedy manner: each predicted user is appended to the history before the next prediction, with padding tokens excluded from prediction. To ensure comparable evaluation at every horizon, each dataset uses a fixed TEST cohort containing sufficient future participants for its largest horizon. This is referred to as retrospective, fixed-budget continuation evaluation on eligible cascades. Specifically, Twitter evaluates $H \in \{3, 5, 10\}$ on 123 cascades, Weibo evaluates $H \in \{10, 15, 20\}$ on 350 cascades, and Quora evaluates $H \in \{3, 5\}$ on 101 cascades; we denote these cohorts by $\mathcal{D}_T$, $\mathcal{D}_W$, and $\mathcal{D}_Q$, respectively. At inference, the deployed student receives only the observed/generated history and the social graph, with no access to any future information. Appendix E.2 specifies the autoregressive evaluation protocol.

Baselines and Metrics. We compare FutureWay with MS-HGAT (Sun et al., 2022), RotDiff (Qiao et al., 2023), SILN (Feng et al., 2025), PMRCA (He et al., 2025), CARE (Zhong et al., 2024), and GRID (Feng et al., 2026), all evaluated under the same long-horizon rollout protocol. The primary metrics are SetRecall@H, which measures recovery of the next H ground-truth participants irrespective of order, and nLCS@H, which additionally measures how well their relative participation order is preserved. Both metrics compare the first H generated users with the first H ground-truth users and are macro-averaged over the corresponding fixed TEST cohort. SR denotes SetRecall; both SR and nLCS are reported as percentages, and all reported gains and drops are relative changes unless stated otherwise. Appendices E.3, E.4, and E.2 provide baseline adaptation details, model-selection settings, and formal metric definitions, respectively.

5.1 Main Comparison

Prediction Performance. Table 2 shows that FutureWay achieves the highest mean SetRecall and nLCS across all evaluated datasets and horizons, averaging a 10.56% relative improvement over the strongest comparator in each of the 16 settings. The gains persist at the longest horizons, where predictions depend increasingly on generated histories. Improvements over Backbone further support future-informed training within the same architecture, without future access at inference.

Table 2: Long-horizon prediction performance on three datasets. SR and nL denote SetRecall and nLCS. Results are reported as mean ± standard deviation on 5 independent runs. Higher is better.

Method	Twitter						Quora				Weibo					
	SR@3	SR@5	SR@10	nL@3	nL@5	nL@10	SR@3	SR@5	nL@3	nL@5	SR@10	SR@15	SR@20	nL@10	nL@15	nL@20
MSHGAT	13.82 ±0.72	15.32 ±0.63	18.78 ±0.88	13.39 ±0.68	14.37 ±0.59	15.82 ±0.74	3.83 ±0.44	4.32 ±0.35	3.56 ±0.36	3.96 ±0.31	5.05 ±0.28	5.37 ±0.25	5.69 ±0.24	4.33 ±0.27	4.18 ±0.20	4.02 ±0.18
RotDiff	14.53 ±0.65	16.07 ±0.61	19.46 ±0.79	14.09 ±0.61	15.15 ±0.66	16.55 ±0.69	4.03 ±0.36	4.59 ±0.33	3.83 ±0.38	4.24 ±0.36	5.31 ±0.24	5.61 ±0.21	5.94 ±0.26	4.55 ±0.23	4.39 ±0.18	4.19 ±0.17
CARE	14.96 ±0.73	16.62 ±0.58	20.05 ±0.84	14.53 ±0.65	15.61 ±0.60	17.02 ±0.71	4.29 ±0.47	4.83 ±0.30	4.03 ±0.36	4.44 ±0.27	5.58 ±0.22	5.91 ±0.20	6.23 ±0.22	4.76 ±0.21	4.59 ±0.17	4.39 ±0.15
SILN	15.77 ±0.67	17.56 ±0.59	21.19 ±0.80	15.39 ±0.62	16.52 ±0.70	18.03 ±0.65	4.69 ±0.54	5.19 ±0.26	4.29 ±0.47	4.75 ±0.24	6.12 ±0.21	6.31 ±0.17	6.69 ±0.20	5.09 ±0.22	4.96 ±0.13	4.76 ±0.14
PMRCA	15.28 ±0.71	17.07 ±0.62	20.67 ±0.86	14.91 ±0.66	16.13 ±0.68	17.58 ±0.67	4.55 ±0.49	5.31 ±0.26	4.42 ±0.44	4.83 ±0.23	5.98 ±0.23	6.47 ±0.19	6.75 ±0.21	5.12 ±0.23	5.06 ±0.14	4.81 ±0.13
GRID	15.50 ±0.65	17.30 ±0.60	20.93 ±0.81	15.12 ±0.62	16.33 ±0.65	17.79 ±0.63	4.55 ±0.54	5.23 ±0.23	4.42 ±0.44	4.83 ±0.23	6.03 ±0.20	6.39 ±0.18	6.86 ±0.20	5.20 ±0.21	5.00 ±0.13	4.83 ±0.12
Backbone	15.99 ±0.69	17.79 ±0.57	21.43 ±0.82	15.61 ±0.65	16.75 ±0.77	18.26 ±0.62	4.62 ±0.62	5.27 ±0.23	4.36 ±0.54	4.87 ±0.23	6.07 ±0.24	6.43 ±0.18	6.81 ±0.22	5.17 ±0.25	5.03 ±0.12	4.85 ±0.13
FUTUREWAY	**16.59** ±0.73	**18.37** ±0.75	**22.73** ±1.01	**16.15** ±0.56	**17.40** ±0.65	**19.37** ±0.61	**5.15** ±0.50	**6.14** ±0.59	**4.95** ±0.52	**5.74** ±0.54	**6.83** ±0.11	**7.62** ±0.14	**8.09** ±0.15	**5.86** ±0.11	**5.77** ±0.07	**5.49** ±0.09

Figure 4: Runtime scalability across rollout length and teacher horizons.

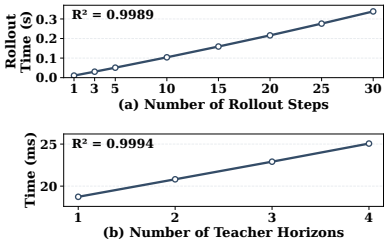


Table 3: Ablation study on five independent runs. We report the two longest-horizon metrics on all three datasets.

Variant	Twitter		Quora		Weibo	
	SR@10	nL@10	SR@5	nL@5	SR@20	nL@20
Best Single Horizon	21.37(↓6.0%)	18.11(↓6.5%)	5.54(↓9.8%)	5.15(↓10.3%)	7.86(↓2.8%)	5.37(↓2.2%)
Uniform Horizons	21.54(↓5.2%)	18.20(↓6.0%)	5.90(↓3.9%)	5.31(↓7.5%)	7.66(↓5.3%)	5.26(↓4.2%)
Learnable Mixture	21.33(↓6.2%)	18.11(↓6.5%)	5.35(↓12.9%)	4.95(↓13.8%)	7.79(↓3.7%)	5.29(↓3.6%)
Random Future	20.93(↓7.9%)	17.89(↓7.6%)	5.19(↓15.5%)	4.44(↓22.6%)	6.75(↓16.6%)	4.76(↓13.3%)
w/o Representation KD	20.29(↓10.7%)	18.02(↓7.0%)	4.95(↓19.4%)	4.75(↓17.2%)	7.91(↓2.2%)	5.41(↓1.5%)
w/o Policy KD	21.37(↓6.0%)	18.39(↓5.1%)	4.95(↓19.4%)	4.55(↓20.7%)	7.74(↓4.3%)	5.35(↓2.6%)
On-Policy CE-Only	22.06(↓2.9%)	18.80(↓2.9%)	5.15(↓16.1%)	4.95(↓13.8%)	7.46(↓7.8%)	5.14(↓6.4%)
w/o Rollout Refresh	21.43(↓5.7%)	18.37(↓5.2%)	4.75(↓22.6%)	4.55(↓20.7%)	7.73(↓4.4%)	5.41(↓1.5%)
Full Model	**22.73**	**19.37**	**6.14**	**5.74**	**8.09**	**5.49**

Efficiency Analysis. Figure 4(a) shows approximately linear rollout-time scaling, reaching about 0.33 seconds for 30 steps on the Twitter TEST cohort. Panel (b) measures guidance computation on 188 training states: runtime rises from approximately 19 to 25 ms as teacher horizons increase from one to four. This measures guidance overhead only; Appendix A.6 reports further component benchmarks. Only the student is retained at inference: mean per-batch latency exceeds Backbone by 1.70 ms on Twitter and 0.76 ms on Weibo and is nearly unchanged on Quora (Table 7).

5.2 Ablation Study

In Table 3, we conduct a comprehensive ablation study to examine three aspects of the design: *(1) Which guidance to use?* The full model outperforms *Best Single Horizon*, *Uniform Horizon Mix.*, and *Learnable Mixture* across datasets. These comparisons support combining multiple future perspectives and adapting their contributions to the current student. In *Learnable Mixture*, an MLP predicts state-dependent horizon weights by student prediction joint-training. Its lower performance supports the effectiveness of the proposed student-aware weighting rule. Moreover, *Random Future*, which replaces future participants with randomized users, also lowers performance, supporting the importance of informative future context for corrective guidance. *(2) How to transfer guidance?* Removing either *Policy KD* or *Representation KD* lowers performance, supporting complementary supervision of the prediction distribution and the student representation decoded through the frozen head. The full model also outperforms *On-Policy CE-Only*, which retains corrective labels. This suggests that teacher distributions provide useful supervision beyond identifying a single corrective participant at each generated state. *(3) Where to learn?* Removing *rollout refresh* also degrades performance, with a particularly pronounced effect on Quora. This supports updating the training states as the student changes, so that supervision remains relevant to its current rollout behavior. The relative impact of these components varies across datasets, but the full model achieves the highest reported scores on all six ablation metrics.

5.3 Sensitivity Analysis

Figure 5 reveals two patterns on Twitter. First, moderate adaptive mixture weights improve both metrics over uniform weighting ($\rho = 0$), while intermediate distillation weights (λ, μ) generally perform well. Increasing these coefficients further provides no consistent additional benefit. Second, increasing rollout depth L yields diminishing gains, while additional on-policy rounds K improve both metrics overall. At larger rollout depths, coverage largely saturates while nLCS continues to improve slightly. This is consistent with deeper supervision improving relative-order recovery after coverage saturates. Gains from additional rounds support iterative refinement; the rollout-refresh ablation more directly supports updating generated training histories.

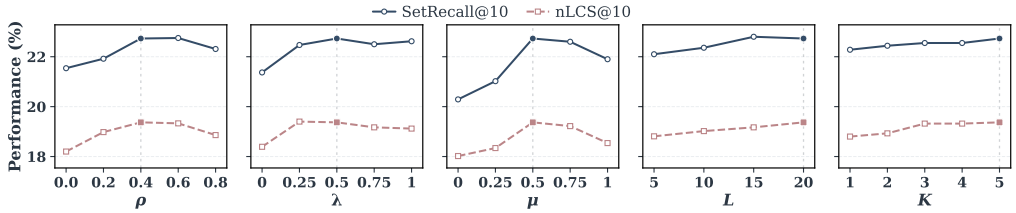


Figure 5: Hyperparameter sensitivity on Twitter for ρ, λ, μ, L, and K.

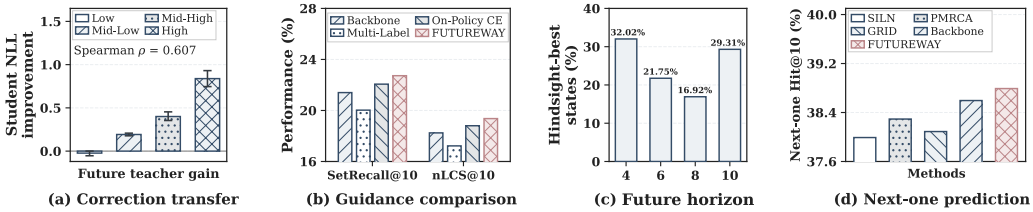


Figure 6: Mechanistic diagnostics on Twitter: (a) teacher-to-student correction transfer; (b) supervision strategies; (c) hindsight-best horizon frequencies; (d) next-participant accuracy.

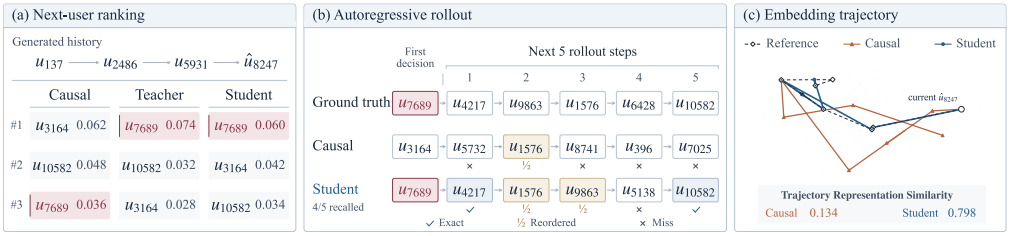


Figure 7: Case study on Twitter. (a) The student promotes u_{7689} from third to first, matching the teacher. (b) Over five subsequent steps, it recovers four users versus the baseline's one. (c) Its embedding trajectory more closely follows the ground-truth reference.

5.4 Mechanistic Analysis

Figure 6 examines four aspects of future-informed training. ***(a)*** *Transfer of corrective guidance.* States are grouped by the teacher's gain over the prefix-only predictor. Student NLL improvement increases with teacher gain (Spearman correlation 0.607), supporting the transfer of stronger future-informed corrections to a student without future access. ***(b)*** *Supervision on generated histories.* Multi-label training on ground-truth histories falls below Backbone, whereas corrective cross-entropy on student-generated histories (On-Policy CE) improves both SR@10 and nLCS@10. FUTUREWAY improves both further. These comparisons support supervising rollout states and enriching corrective labels with teacher distributions. ***(c)*** *Variation in useful horizons.* The hindsight-best horizon assigns the highest teacher probability to the corrective target. All four horizons are selected across states; most choices differ from any fixed horizon. This variation motivates adaptive weighting. ***(d)*** *Preservation of local accuracy.* FUTUREWAY slightly improves next-participant Hit@10 over strong baselines. Thus, long-horizon gains coexist with improved local accuracy.

Case Study. In Figure 7, the student learns how to correct errors with the teacher guidance, and thus improves its subsequent predictions. Specifically, the student still swaps u_{9863} and u_{1576} and misses u_{6428}. These residual errors explain why coverage and order require separate metrics: local correction improves the continuation without fully recovering its participants or their ordering.

6 Conclusion

This paper studies long-horizon information diffusion prediction and introduces the FUTUREWAY framework, which transfers future-informed corrective guidance to model-generated histories encountered during forecasting, while ensuring that the deployed predictor uses only the observed prefix. Experiments on three real-world benchmarks show that learning from future-informed guidance on model-generated histories can consistently improve long-horizon diffusion forecasting. Future work may incorporate richer behavioral and semantic context, for example by using large language models to summarize user interests or interaction patterns from available content, and study how such signals can complement structural and temporal evidence for better long-horizon forecasting.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we use generative AI tools to assist with language editing and manuscript refinement. Generative AI is not used to determine the scientific conclusions of the paper or replace human verification of experimental findings. All AI-assisted writing content has been reviewed and verified by the authors. Related work, technical claims, and mathematical formulations are verified against the relevant literature, implementations, or experimental evidence by authors. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

(This section is **recommended** and does not count toward the page limit.)

This work studies information diffusion prediction using benchmark datasets derived from online social platforms, including Twitter, Weibo, and Quora. User identities are treated only as abstract and anonymous identifiers required for modeling diffusion sequences and graph structure. While information diffusion modeling may support the study of information propagation and collective behavior, this work focuses on methodological advances in sequence prediction and does not develop mechanisms for influencing users or optimizing interventions on real-world populations. We follow the access conditions and usage requirements of the datasets used in our experiments. Potential applications should also consider privacy, fairness, and broader societal consequences.

REPRODUCIBILITY STATEMENT

(This section is **recommended** and does not count toward the page limit.)

We provide sufficient implementation details to facilitate reproducibility. The main paper describes the model architecture, training objective, and inference procedure, while the appendix provides additional implementation details, hyperparameters, ablations, sensitivity analyses, and efficiency measurements. Main experiments are conducted over five random seeds, with mean and variability reported. The core implementation code is provided in an anonymous repository at **https://anonymous.4open.science/r/FutureWay**.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/5be69a584901a26c521c2b51e40a4c20-Abstract-Conference.html.

Joseph B. Bak-Coleman, Ian Kennedy, Morgan Wack, Andrew Beers, Joseph S. Schafer, Emma S. Spiro, Kate Starbird, and Jevin D. West. Combining interventions to reduce the spread of viral misinformation. *Nature Human Behaviour*, 6(10):1372–1380, 2022. doi: 10.1038/s41562-022-01388-6. URL https://doi.org/10.1038/s41562-022-01388-6.

Arindam Banerjee, Xin Guo, and Hui Wang. On the optimality of conditional expectation as a bregman predictor. *IEEE Transactions on Information Theory*, 51(7):2664–2669, 2005. doi: 10.1109/TIT.2005.850145. URL https://doi.org/10.1109/TIT.2005.850145.

Zhangtao Cheng, Wenxue Ye, Leyuan Liu, Wenxin Tai, and Fan Zhou. Enhancing information diffusion prediction with self-supervised disentangled user and cascade representations. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 3808–3812. ACM, 2023. doi: 10.1145/3583780.3615230. URL https://doi.org/10.1145/3583780.3615230.

Zihan Feng, Yajun Yang, Xin Huang, Hong Gao, Liping Jing, and Qinghua Hu. Efficient sphere-effect based information diffusion prediction on large-scale social networks. In *Proceedings of*

the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, pp. 615–625. ACM, 2025. doi: 10.1145/3711896.3736925. URL https://doi.org/10.1145/3711896.3736925.

Zihan Feng, Yajun Yang, Xin Huang, Xin Wang, Hong Gao, and Qinghua Hu. Many hands make light work: Group-based information diffusion prediction over long-context cascades. In *Proceedings of the ACM Web Conference 2026*, pp. 4472–4481. ACM, 2026. doi: 10.1145/3774904.3792082. URL https://doi.org/10.1145/3774904.3792082.

Weikang He, Yunpeng Xiao, Mengyang Huang, Xuemei Mou, Rong Wang, and Qian Li. A pattern-driven information diffusion prediction model based on multisource resonance and cognitive adaptation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, pp. 592–601. ACM, 2025. doi: 10.1145/3726302.3729883. URL https://doi.org/10.1145/3726302.3729883.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. LightGCN: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2020. doi: 10.1145/3397271.3401063. URL https://doi.org/10.1145/3397271.3401063.

Nathan Oken Hodas and Kristina Lerman. The simple rules of social contagion. *CoRR*, abs/1308.5015, 2013. URL http://arxiv.org/abs/1308.5015.

Pengfei Jiao, Hongqian Chen, Qing Bao, Wang Zhang, and Huaming Wu. Enhancing multi-scale diffusion prediction via sequential hypergraphs and adversarial learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8571–8581, 2024. doi: 10.1609/aaai.v38i8.28701. URL https://doi.org/10.1609/aaai.v38i8.28701.

David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. *Theory of Computing*, 11:105–147, 2015. doi: 10.4086/toc.2015.v011a004. URL https://theoryofcomputing.org/articles/v011a004/.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327. Association for Computational Linguistics, 2016. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139/.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. In *4th International Conference on Learning Representations*, 2016. URL https://arxiv.org/abs/1511.03643.

Hongliang Qiao, Shanshan Feng, Xutao Li, Huiwei Lin, Han Hu, Wei Wei, and Yunming Ye. RotDiff: A Hyperbolic Rotation Representation Model for Information Diffusion Prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 2065–2074, 2023. doi: 10.1145/3583780.3615041. URL https://doi.org/10.1145/3583780.3615041.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for Thin Deep Nets. In *3rd International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6550.

Stéphane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635. PMLR, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

Aravind Sankar, Xinyang Zhang, Adit Krishnan, and Jiawei Han. Inf-vae: A variational autoencoder framework to integrate homophily and influence in diffusion prediction. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, 2020*, pp. 510–518. ACM, 2020.

Tianyang Shao, Weixin Zeng, and Xiang Zhao. DSHCL: Dual-state hypergraph contrastive learning for information diffusion prediction. *IEEE Transactions on Knowledge and Data Engineering*, 37(9):5158–5170, 2025. doi: 10.1109/TKDE.2025.3581419. URL https://doi.org/10.1109/TKDE.2025.3581419.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-Distillation Enables Continual Learning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=qA6FgH0nnZ.

Ling Sun, Yuan Rao, Xiangbo Zhang, Yuqian Lan, and Shuanghe Yu. MS-HGAT: Memory-enhanced sequential hypergraph attention network for information diffusion prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 4156–4164, 2022. doi: 10.1609/aaai.v36i4.20334. URL https://doi.org/10.1609/aaai.v36i4.20334.

Vladimir Vapnik and Rauf Izmailov. Learning Using Privileged Information: Similarity Control and Knowledge Transfer. *Journal of Machine Learning Research*, 16(61):2023–2049, 2015. URL https://www.jmlr.org/papers/v16/vapnik15b.html.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Ding Wang, Wei Zhou, and Songiln Hu. Information diffusion prediction with graph neural ordinary differential equation network. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 9699–9708. ACM, 2024. doi: 10.1145/3664647.3681363. URL https://doi.org/10.1145/3664647.3681363.

Cheng Yang, Jian Tang, Maosong Sun, Ganqu Cui, and Zhiyuan Liu. Multi-scale Information Diffusion Prediction with Reinforced Recurrent Networks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4033–4039, 2019. doi: 10.24963/ijcai.2019/560. URL https://www.ijcai.org/proceedings/2019/560.

Shenzhi Yang, Guangcheng Zhu, Bowen Song, Haobo Wang, Mingxuan Xia, Xing Zheng, Yingfan Ma, Zhongqi Chen, Weiqiang Wang, Junbo Zhao, and Gang Chen. OPRD: On-Policy Representation Distillation, 2026. URL https://arxiv.org/abs/2606.06021v4. Preprint; version 4.

Chunyuan Yuan, Jiacheng Li, Wei Zhou, Yijun Lu, Xiaodan Zhang, and Songlin Hu. DyHGCN: A dynamic heterogeneous graph convolutional network to learn users' dynamic preferences for information diffusion prediction. In *Machine Learning and Knowledge Discovery in Databases: ECML PKDD 2020*, volume 12459 of *Lecture Notes in Computer Science*, pp. 347–363. Springer, 2021. doi: 10.1007/978-3-030-67664-3_21. URL https://doi.org/10.1007/978-3-030-67664-3_21.

Jing Zhang, Biao Liu, Jie Tang, Ting Chen, and Juanzi Li. Social influence locality for modeling retweeting behaviors. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pp. 2761–2767, 2013.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2601.18734.

Ting Zhong, Jienan Zhang, Zhangtao Cheng, Fan Zhou, and Xueqin Chen. Information diffusion prediction via cascade-retrieved in-context learning. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2472–2476. ACM, 2024. doi: 10.1145/3626772.3657909. URL https://doi.org/10.1145/3626772.3657909.

A SUPPLEMENTARY EXPERIMENTS

We further examine prediction behavior over rollout steps, hyperparameter sensitivity, future-informed corrective supervision across datasets, the agreement and differences between teacher horizons, and the computational cost of multi-horizon guidance. The teacher diagnostics compare predictors on identical histories, targets, and eligible-user sets within each run. They complement the student rollout results in Table 2. Five-seed results use $\{2025, 2026, 2027, 2028, 2048\}$ and report means with sample standard deviations.

A.1 PREDICTION PERFORMANCE OVER ROLLOUT STEPS

Figure 8 compares stepwise next-participant Hit@10 under two conditioning settings: reference prefixes contain the preceding ground-truth participants, whereas free-running histories incorporate the model's preceding predictions. Both settings begin from the same observed prefix, yielding identical first-step performance. As generation proceeds, free-running Hit@10 generally declines, while performance under reference prefixes fluctuates within a substantially higher range. On Twitter, free-running Hit@10 decreases from approximately 36% at the first step to 7% at step 15, whereas the reference-prefix value remains around 35% at the final step. On Weibo, the corresponding values decrease from approximately 7.8% to 0.4% over 20 steps, compared with approximately 6.3% under the final reference prefix. Quora exhibits the same pattern over five steps, ending near 3.3% in free-running prediction versus 9.8% under the reference prefix.

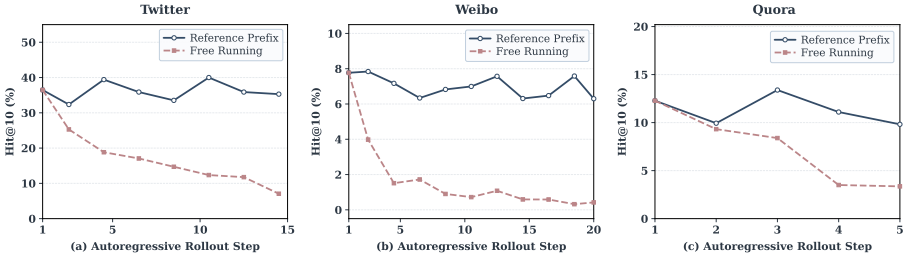


Figure 8: Stepwise next-participant Hit@10 under reference-prefix and free-running conditioning on Twitter, Weibo, and Quora. Reference prefixes use preceding ground-truth participants; free-running histories incorporate preceding model predictions.

The growing separation is consistent with errors in generated histories making subsequent next-participant prediction more difficult, motivating corrective supervision at student-visited states. Reference-prefix conditioning uses the ground-truth history at each step, as commonly done in next-participant prediction evaluation. Here, it serves as a diagnostic rather than the free-running forecasting procedure used for long-horizon evaluation. Moreover, the plotted rollout step is distinct from the training rollout depth L examined below.

A.2 SENSITIVITY ANALYSIS ON WEIBO AND QUORA

Figures 9 and 10 extend the Twitter sensitivity analysis in Section 5.3 to Weibo and Quora. They report SetRecall and nLCS at the respective maximum evaluation horizons, $H = 20$ and $H = 5$, as the adaptive mixture fraction ρ, policy-distillation weight λ, representation-distillation weight μ, training rollout depth L, and number of refinement rounds K vary.

Horizon weighting and distillation strength. In the evaluated sweeps, both datasets achieve higher SetRecall and nLCS with nonzero adaptive mixture weights than with uniform weighting alone. On Weibo, both metrics peak near $\rho = 0.4$ and vary only modestly at larger tested values. On Quora, SetRecall is highest near $\rho = 0.2$, while nLCS peaks near $\rho = 0.4$, indicating that the two metrics need not favor exactly the same mixture. For λ and μ, intermediate values around 0.5 perform well on both datasets; increasing either coefficient further provides no consistent improvement. Quora is more sensitive to removing either distillation term: setting $\lambda = 0$ or $\mu = 0$ visibly reduces both metrics relative to the intermediate settings. Weibo exhibits smaller changes across the same sweeps.

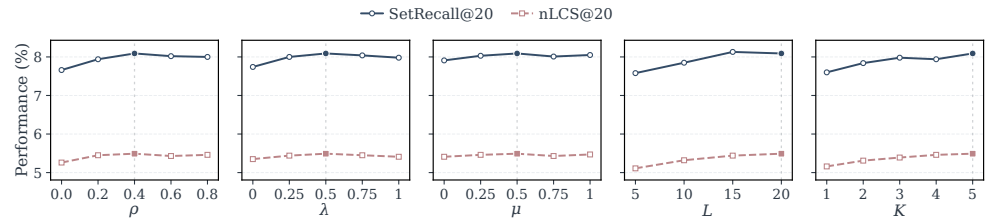


Figure 9: Sensitivity on Weibo: SetRecall@20 and nLCS@20 as ρ, λ, μ, L, and K vary.

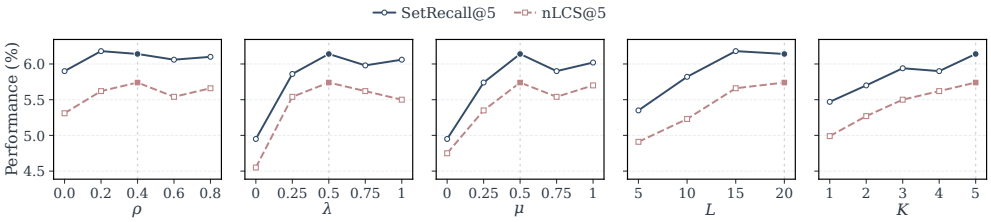


Figure 10: Sensitivity on Quora: SetRecall@5 and nLCS@5 as ρ, λ, μ, L, and K vary.

Rollout depth and refinement rounds. Increasing L from 5 to 15 improves both metrics on Weibo and Quora. Extending it to 20 yields a small further increase in nLCS, while SetRecall is nearly saturated and slightly decreases. These results suggest that, within the tested range, deeper training rollouts can further improve relative-order recovery even as coverage gains level off. Increasing K from one to five yields an overall improvement in both metrics, although SetRecall is not strictly monotonic across rounds. Both metrics show an overall increase as the number of refinement rounds grows under the evaluated training schedule. Together, these sweeps show that the reported settings perform well on both coverage and relative-order metrics within the tested ranges.

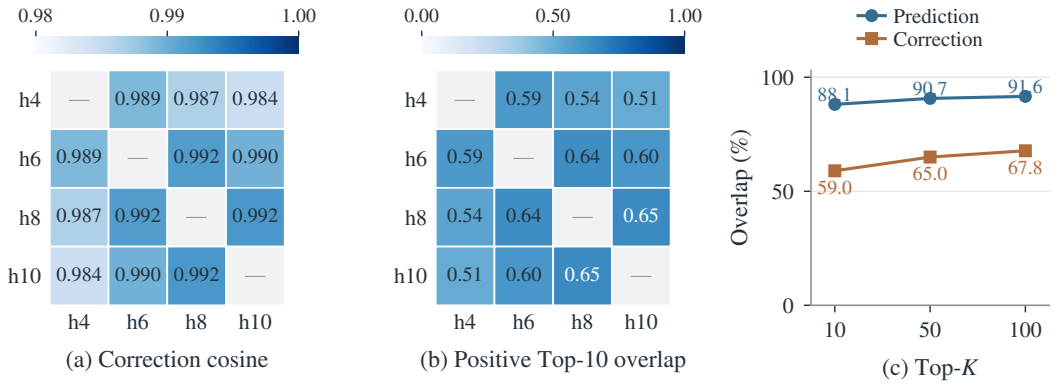
A.3 Future-Informed Corrective Guidance Across Datasets

Table 4 extends Figure 2(b) to Weibo and Quora. We evaluate the prefix-only predictor and four teachers on cached VALID histories generated by the same frozen student within each seed. The corrective target and post-target windows follow Eq. (3). All comparisons use common support where every horizon in $\mathcal{H} = \{4, 6, 8, 10\}$ is available.

Table 4: **Future guidance on Weibo and Quora.** We report target NLL on the common-support VALID states, averaged over five seeds (mean $\pm$ sample standard deviation). Lower is better. Best Fixed selects one horizon using the average VALID NLL across seeds; the target-aware Oracle selects the horizon assigning the highest probability to the observed target at each state and is used only as a target-aware diagnostic.

Method	Weibo NLL ↓	Quora NLL ↓
Prefix-only predictor	11.626 ± 0.095	7.515 ± 0.105
Teacher h=4	10.245 ± 0.039	7.395 ± 0.104
Teacher h=6	10.257 ± 0.045	7.389 ± 0.115
Teacher h=8	10.230 ± 0.026	7.397 ± 0.111
Teacher h=10	10.256 ± 0.045	7.396 ± 0.109
Best Fixed Horizon	10.230 ± 0.026	7.389 ± 0.115
Target-aware Oracle	10.173 ± 0.031	7.356 ± 0.113

Every horizon-specific teacher has lower mean target NLL than the prefix-only predictor in both datasets. The best fixed horizon is $h = 8$ on Weibo and $h = 6$ on Quora, reducing mean NLL



(a) Correction cosine (b) Positive Top-10 overlap (c) Top-K

Figure 11: **Global agreement and local ranking differences on Twitter.** (a) Correction cosine. (b) Positive-correction Top-10 overlap. (c) Prediction and correction overlap at three cutoffs, averaged over six horizon pairs. The plot's Top-K denotes the ranking cutoff m, not refinement rounds. Heatmaps omit diagonals and use different color scales. All use the same 1,655 VALID states.

by 1.396 and 0.126 nats, respectively. Best Fixed is selected and summarized on the same VALID probes, so this is a descriptive comparison.

State-dependent target preferences. The target-aware oracle selects the horizon assigning the largest probability to the observed corrective target at each state. Table 5 reports its assignments and improvement over Best Fixed. The oracle selects a different horizon on 67.13% of Weibo states and 68.79% of Quora states, with mean target-log-probability gains of 0.0572 and 0.0331 nats. Each horizon receives some assignments.

Table 5: **State-dependent horizon preference on VALID.** Disagreement is the fraction of states where the target-aware oracle differs from Best Fixed; gain is the mean increase in target log-probability over Best Fixed. Frequencies indicate the share of states assigned to each horizon. Values are five-seed means ± sample standard deviations; rounded frequencies need not sum to 100%.

Quantity	Weibo	Quora
Best fixed horizon	8	6
Oracle–fixed disagreement (%)	67.13 ± 13.88	68.79 ± 12.82
Oracle gain (nats)	0.0572 ± 0.0228	0.0331 ± 0.0080
$h = 4$ frequency (%)	23.3 ± 5.1	30.2 ± 8.9
$h = 6$ frequency (%)	21.6 ± 5.0	31.2 ± 12.8
$h = 8$ frequency (%)	32.9 ± 13.9	16.5 ± 6.4
$h = 10$ frequency (%)	22.3 ± 5.4	22.0 ± 7.5

A.4 Teacher Agreement and Local Ranking Differences

We compare four frozen teachers on 1,655 Twitter VALID states with all horizons available. For eligible user a, the correction relative to the same cached prefix-only predictor is $r_{i,h}(a) = \log q_{i,h}(a) - \log p_{\text{diag}}(a \mid X_i)$. Figure 11 and Table 6 compare these correction vectors and the teachers' candidate rankings. Summaries weight states equally; they describe one frozen population, rather than independent replications of states from the same cascade.

Correction cosine ranges from 0.9845 to 0.9920. Prediction Top-10 overlap averages 88.12%, compared with 58.98% for positive-correction Top-10 overlap. At cutoffs 50 and 100, prediction overlap is 90.73% and 91.61%, while correction overlap is 65.00% and 67.77%. For cutoff m, overlap is the intersection size divided by m. Correction sets rank signed $r_{i,h}(a)$; every selected entry is positive, and all teachers agree in sign on the unions of these sets. On these states, the selected users

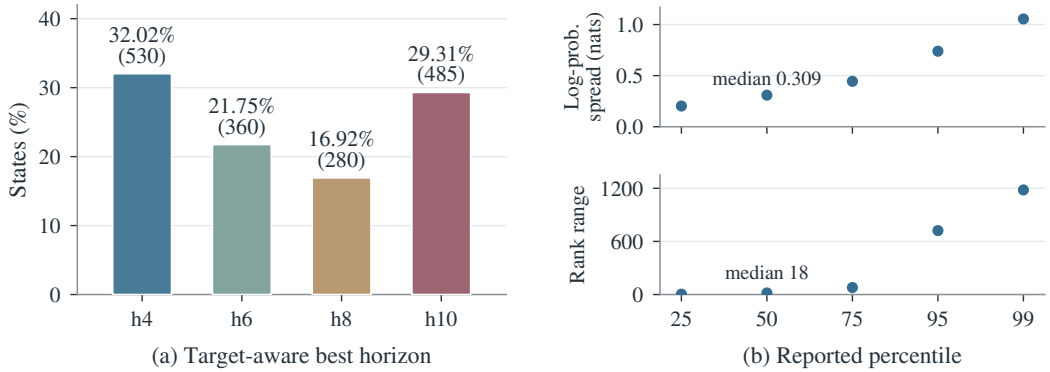
receive positive log-probability changes relative to the cached prefix-only predictor across all teachers, while their rankings differ across horizons. This observation provides empirical motivation for the uniform–adaptive mixture discussed in Appendix B.5.

Table 6: **Global alignment and local ranking differences.** Entries average the same 1,655 states; the last row averages the six pairs. Overlap is $|A \cap B|/K$ in percent. Corr. denotes positive log-policy correction rankings; cosine compares eligible-action correction vectors.

Horizon pair	Corr. cosine	Pred.@10	Corr.@10	Corr.@50
4/6	0.9890	88.19	59.04	64.99
4/8	0.9865	86.72	54.49	61.31
4/10	0.9845	85.82	51.26	58.26
6/8	0.9915	89.70	63.98	69.31
6/10	0.9897	88.37	60.27	66.02
8/10	0.9920	89.93	64.85	70.13
Pair mean	0.9889	88.12	58.98	65.00

A.5 State-Dependent Target Preferences on Twitter

On the same 1,655 states, let $h_i^\star = \min \arg\max_{h \in \mathcal{H}} \log q_{i,h}(y_i)$, so exact ties favor the smaller horizon. Define the within-state target-log-probability spread as $S_i = \max_h \log q_{i,h}(y_i) - \min_h \log q_{i,h}(y_i)$. The target-rank range is the largest minus the smallest target rank across horizons, using the same eligible-user set. These quantities assess target-level differences that may be obscured by high full-vector similarity.



(a) Target-aware best horizon (b) Reported percentile

Figure 12: **Target-aware horizon preferences on Twitter.** (a) Best-horizon frequencies and counts. (b) Quantiles of target-log-probability spread and target-rank range; percentiles are displayed as equally spaced categories. These are hindsight diagnostics on 1,655 VALID states.

The best-horizon counts for $h = 4, 6, 8, 10$ are 530, 360, 280, and 485, respectively. Mean and median log-probability spreads are 0.3485 and 0.3091 nats; the median corresponds to a best-to-worst target-probability ratio of approximately 1.36. The target-rank range has median 18 and 95th percentile 722.1. Thus, similar teachers can retain locally relevant score and rank differences.

A.6 Computational Cost and Inference Latency

Figure 13 measures multi-horizon computation on 128 fixed states per dataset, with batch size 32. Each configuration instantiates teachers only for its active horizons. Runtime covers processing the 128-state workload and is summarized over five timing repetitions after warm-up on NVIDIA H200 NVL GPUs. Memory is the current process's peak CUDA allocation. This is a separate component benchmark from the 188-state guidance measurement in Figure 4(b); its absolute times are not directly comparable with that panel or with complete training time.

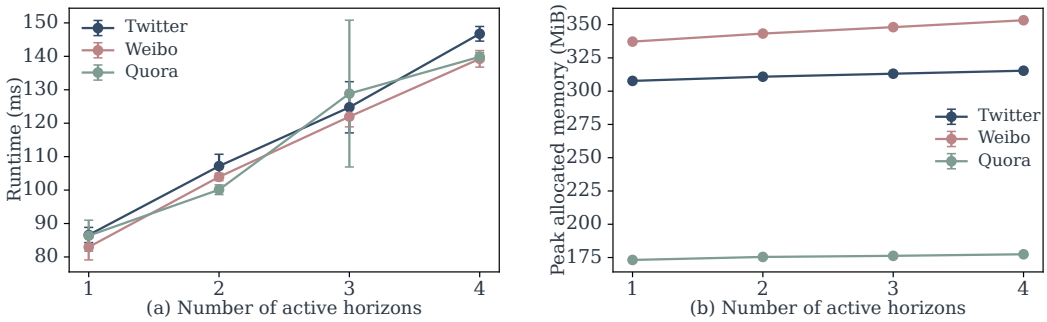


Figure 13: **Component cost of adding teacher horizons.** (a) Runtime for 128 fixed states, with mean $\pm$ sample standard deviation over five timing repetitions. (b) Peak allocated GPU memory for the same configurations.

Increasing the active horizon count from one to four raises mean runtime by $1.69\times$ on Twitter, $1.68\times$ on Weibo, and $1.62\times$ on Quora. Peak memory increases by 2.49%, 4.75%, and 2.47%, respectively. The three-horizon Quora measurement has visibly greater timing variability. These observations characterize the measured workload: adding teachers increases computation, while incremental allocated memory is small and lightweight relative to the shared backbone. Inference retains only the prefix-only student.

Inference latency. Table 7 compares Backbone and the distilled student on matched inputs with batch size 32. Mean student latency is 17.73, 7.43, and 7.84 ms on Twitter, Weibo, and Quora, respectively, compared with 16.03, 6.67, and 7.83 ms for Backbone. Thus, measured latency remains close to Backbone, with small absolute increases on Twitter and Weibo and nearly identical timing on Quora. This is consistent with retaining the same prefix-only inference architecture after distillation; no teacher or horizon-weighting computation is required at deployment. Note that the settings of these per-batch timings are distinct from the full-cohort rollout timings in Figure 4(a).

Table 7: **Matched inference latency per batch of 32.** Inputs are matched between the pretrained backbone and student. Twitter summarizes five timing repetitions for one seed; Weibo and Quora summarize five seed means, each measured over 30 repetitions. Values are means $\pm$ sample standard deviations, in milliseconds.

Dataset	Backbone (ms)	Student (ms)
Twitter	16.03 ± 0.65	17.73 ± 1.15
Weibo	6.67 ± 2.00	7.43 ± 2.69
Quora	7.83 ± 1.81	7.84 ± 1.81

B Theoretical Analysis and Proofs

We prove the projection result, formalize the state-dependent horizon comparison, and connect both to the score and training procedure in Section 4. All results concern a fixed aligned-data law and fixed teacher kernels unless stated otherwise.

B.1 Notation

Table 8 collects the symbols used in the proofs and algorithms. State and round indices are suppressed when fixed. Y denotes the corrective-target random variable and y_t its realization; teacher horizon h, evaluation horizon H, and training rollout depth L have distinct roles.

Table 8: Notation for projection, horizon weighting, and training.

Symbol	Meaning
$G = (V, E)$, $C^\star$, $\widehat{C}_t$	Social graph, ground-truth cascade, and observed/generated history.
X_t, Y, R_t, L_t	State $(\widehat{C}_t, G)$, corrective target, ordered sequence of ground-truth participants absent from the current history, and its length.
h, $Z_{t,h}$, $\mathcal{H}_t$	Teacher horizon, post-target window, and available horizons from $\mathcal{H}$.
H, L, k, K	Forecast horizon, training rollout depth, round index, and number of rounds.
$\mathcal{A}(X)$	Eligible users, excluding padding and all users already in the history.
$\bar{\theta}$, θ_k, $\bar{W}$	Pretrained parameters, round-k student, and frozen pretrained head.
$q_h^{(k)}$, ϕ_h	Horizon-specific teacher and its trainable adapter parameters.
p_0, p_θ, p_{ref}^θ	Pretrained anchor, deployed student distribution, and frozen-head student distribution.
g_h, w_h, Q_k	Student-aware score, horizon weight, and round-k teacher mixture.
ρ, τ_s, λ, μ	Adaptive fraction, score temperature, and policy/representation KD weights.
ν, ν_k, $\nu^\cap$	Fixed aligned law, retained round law, and common-support law.
$p_Q^\star$, $p_h^\star$, Δ_h	Prefix-only projections of Q and q_h, and projected transfer gain.
A, $I_Q(A; Z \mid X)$	Auxiliary teacher action and irreducible conditional future dependence.

B.2 Prediction Setting and Scope

At a generated state $X_t = (\widehat{C}_t, G)$, remove participants already present in the history from $C^\star$, preserving the ground-truth order. The resulting R_t defines $y_t = r_{t,0}$ and $Z_{t,h} = (r_{t,1}, \ldots, r_{t,h})$ whenever $L_t \geq h + 1$, exactly as in Section 4.1. Following common settings in established work, cascades contain distinct users, so the target identity is absent from each post-target window. The student observes X; Y and Z supply training supervision. The prefix-only restriction concerns prediction-time information availability. An aligned generated history is a prediction state used to learn recovery of ground-truth participants, not a physical counterfactual cascade.

On a social network, the vocabulary V is finite. All kernels are measurable, and the variable spaces admit regular conditional distributions. Teachers and predictors use the same nonempty candidate set $\mathcal{A}(X)$, extended by zero outside it, with $Y \in \mathcal{A}(X)$. We use natural logarithms and the usual zero-mass conventions for KL divergence. Theorem 1 applies to any fixed law over (X, Z); when Y is needed, ν also denotes an aligned extension of that law.

The student objective retains states with $\mathcal{H}_t \neq \varnothing$. For clear analysis, horizon comparisons instead condition on every horizon being available: $\nu^\cap = \nu(\cdot \mid \mathcal{H}_t = \mathcal{H})$, assuming this event has positive probability. We write $\mathbb{E}_\cap$ for expectation under this common-support law. The finite, nonempty $\mathcal{H}$ and both prefix-only predictors p_a, p_b are fixed, with finite expected teacher KL for every horizon.

B.3 Proof of Theorem 1: Prefix-Only Projection

Proof of Theorem 1. Under the joint law $\widetilde{\nu}_Q(dx, dz, a) = \nu(dx, dz)Q(a \mid x, z)$, let A denote the auxiliary action. The tower property gives

$$\widetilde{\nu}_Q(A = a \mid X) = \mathbb{E}_\nu[Q(a \mid X, Z) \mid X] = p_Q^\star(a \mid X).$$

Each coordinate is nonnegative and measurable, and their sum is $\mathbb{E}_\nu[\sum_a Q(a \mid X, Z) \mid X] = 1$. Since eligibility depends only on X, the projection also respects $\mathcal{A}(X)$. Thus $p_Q^\star$ is an admissible prefix-only distribution.

Write H for entropy under $\widetilde{\nu}_Q$. For any prefix-only p with finite risk, conditioning its cross-entropy on X yields

$$\begin{aligned}\mathcal{R}_Q(p) &= \mathbb{E}_{\widetilde{\nu}_Q}[-\log p(A \mid X)] - \mathsf{H}(A \mid X, Z) \\ &= \mathsf{H}(A \mid X) - \mathsf{H}(A \mid X, Z) + \mathbb{E}_{\nu_X} D_{\mathrm{KL}}(p_Q^\star \| p) \\ &= I_Q(A; Z \mid X) + \mathbb{E}_{\nu_X} D_{\mathrm{KL}}(p_Q^\star \| p). \end{aligned} \tag{11}$$

Both entropies lie in $[0, \log |V|]$, so there is no subtraction of infinite quantities. Taking $p = p_Q^\star$ gives finite risk $I_Q(A; Z \mid X)$. Nonnegativity of KL proves optimality. Equality holds exactly when $p(\cdot \mid X) = p_Q^\star(\cdot \mid X)$ for ν_X-almost every X, proving uniqueness up to null sets. □

The theorem characterizes the prefix-only component of a given future-informed guidance signal. Its conditional mutual information measures teacher variation that cannot be reproduced from X alone; closer fitting cannot remove that residual. In the ideal-teacher case, under the *same* aligned law P,

$$Q(a \mid X, Z) = P(Y = a \mid X, Z) \quad \Longrightarrow \quad p^\star_Q(a \mid X) = P(Y = a \mid X).$$

This is a statistical consistency statement, not a claim that access to future information during training changes the Bayes-optimal prefix-only predictor. For a learned teacher, the projection inherits the teacher's quality. Attainment by a finite neural student and improvements in SetRecall or nLCS additionally depend on estimation, optimization, and the rollout task.

B.4 State-Dependent Horizon Advantage

Under $\nu^\cap$, define $p^\star_h = \mathbb{E}_\cap[q_h \mid X]$ and $\Delta_h(X) = D_{\mathrm{KL}}(p^\star_h \| p_a) - D_{\mathrm{KL}}(p^\star_h \| p_b)$. The conditional version of Theorem 1 cancels the same future-dependent residual from the two risks, giving

$$\Delta_h(X) = \mathbb{E}_\cap[D_{\mathrm{KL}}(q_h \| p_a) - D_{\mathrm{KL}}(q_h \| p_b) \mid X]\,. \tag{12}$$

Finite risks imply integrability of each gain and of their finite maximum.

Corollary 1 (State-dependent horizon advantage). *Under the common-support law,*

$$\mathbb{E}_\cap\left[\max_{h \in \mathcal{H}} \Delta_h(X)\right] \geq \max_{h \in \mathcal{H}} \mathbb{E}_\cap[\Delta_h(X)]. \tag{13}$$

The inequality is strict if and only if no single horizon maximizes $\Delta_h(X)$ almost surely.

Proof. Let $h^\star$ maximize $\mathbb{E}_\cap \Delta_h$ and write $M(X) = \max_h \Delta_h(X)$. The difference in Eq. (13) is $\mathbb{E}_\cap[M(X) - \Delta_{h^\star}(X)] \geq 0$. It vanishes exactly when $h^\star$ attains $M(X)$ almost surely. Conversely, any fixed horizon attaining $M(X)$ almost surely also attains its expectation, giving equality. □

This formalizes the oracle comparison in Section 4.2: a fixed horizon can lose relative projected agreement when the preferred horizon changes with the state. It does not assert that the implemented weights attain this oracle. Nor does it require heterogeneous gains: if every teacher is ideal under one common aligned law, all $p^\star_h$ equal $P(Y \mid X)$ and the gains coincide. For learned teachers, horizon-dependent approximation errors can produce the heterogeneity measured by Δ_h.

B.5 From Projected Transfer to Student-Aware Weighting

With $p_a = p_0$ and $p_b = p^{(k)}_{\mathrm{ref}}$, the score in Eq. (8) has the exact conditional interpretation

$$\mathbb{E}_\cap[g^{(k)}_h \mid X] = D_{\mathrm{KL}}(p^\star_h \| p_0) - D_{\mathrm{KL}}(p^\star_h \| p^{(k)}_{\mathrm{ref}}). \tag{14}$$

The first divergence measures departure from the pretrained anchor; the second discounts disagreement with the current representation decoded through the frozen head. Teacher entropy cancels, so the realized score is $\mathbb{E}_{q^{(k)}_h}[\log p^{(k)}_{\mathrm{ref}} - \log p_0]$. It measures relative fit to the fixed pretrained anchor and student comparison distribution. At initialization $p^{(1)}_{\mathrm{ref}} = p_0$, all scores vanish and the first round uses uniform guidance.

Consensus and horizon-specific residuals. At a fixed aligned tuple, let $\bar{q} = |\mathcal{H}_t|^{-1} \sum_h q_h$, $\delta_h = q_h - \bar{q}$, and $\alpha_h = \mathrm{softmax}_h(g/\tau_s)$. Equation (9) becomes

$$Q_k = \bar{q} + \rho \sum_{h \in \mathcal{H}_t} \alpha_h \delta_h, \qquad \|Q_k - \bar{q}\| \leq \rho \max_{h \in \mathcal{H}_t} \|q_h - \bar{q}\|. \tag{15}$$

The bound follows from the triangle inequality and convex weights, for any norm. The uniform component preserves the shared distribution, while ρ controls the contribution of horizon-specific residuals. This shrinkage is intended to help when the realized scores depend on a single realization of the future windows and differ at locally important rankings, as in Appendix A.4. Conditional unbiasedness of each score does not pass through the nonlinear softmax. Moreover, the weights depend on the same future evidence as the teachers, so the projection of the mixture is $\mathbb{E}[\sum_h w_h q_h \mid X]$, with the expectation over their product. The target-NLL oracle in Figure 2(b) instead selects using the realized corrective label; it is distinct from the projected-gain oracle in Corollary 1.

B.6 Roundwise On-Policy Interpretation and Scope

At round k, the frozen snapshot θ_{k-1} induces the retained rollout law ν_k. Adapter updates and horizon weighting then define a fixed Q_k for student optimization. Theorem 1 applies to that complete mixture under ν_k, including states with only some horizons available. Common support is needed for the per-horizon comparison, not for the projection of Q_k.

The three terms of Eq. (10) serve complementary roles. Corrective cross-entropy anchors the predictor to the aligned target. Policy KD transfers Q_k through the deployed distribution p_θ. Representation KD transfers the same signal through p^θ_{ref}, shaping the representation evaluated by the fixed pretrained head and used in the next round's comparison. These two outputs share student parameters. Refreshing rollouts changes *where* the projection is learned, while future-informed teachers change *what* supervision is supplied. The fixed-round identity does not establish convergence of $\nu_1 \to \theta_1 \to \nu_2 \to \theta_2 \to \cdots$. The empirical evaluation therefore measures the resulting free-running coverage and order preservation directly. Overall, these above analyses aim to provide a theoretical perspective, rather than a guarantee of performance improvement; experiments aim to validate the effectiveness of the practical design.

C Algorithm Details

Algorithms 1–4 implement the main method. $\mathrm{Eligible}(C)$ excludes padding and every user already in the full history; $\mathrm{MaskedSoftmax}$ normalizes over that set. Bars identify frozen pretrained features and head parameters. $\Omega = (\rho, \tau_s, \lambda, \mu)$ contains the weighting/loss settings, and $\mathcal{S} = (\mathcal{S}_0, \mathcal{S}_{\mathrm{T}}, \mathcal{S}_{\mathrm{S}})$ the optimization schedules. Sequence indexing is one-based. $\theta_{\mathrm{b}} = \theta_{k-1}$ is frozen throughout collection and guidance construction; Q_b denotes Q_k for sample b.

Algorithm 1 presents the prefix-only forward pass shared by pretraining, rollout collection, student refinement, and inference. Lines 1–3 encode the social graph into user embeddings, process the current cascade prefix with a causal Transformer, and extract the prediction representation s. Lines 4–6 then identify eligible users, apply the prediction head and masked softmax, and return the next-participant distribution together with the intermediate features. Every prediction depends only on the current history C and graph G.

Algorithm 2 presents the future-informed teacher forward pass in Eq. (4). Line 1 extracts prefix representations and user embeddings from the frozen pretrained backbone. Lines 2–3 construct a state-conditioned query and a memory combining prefix representations with embeddings of the post-target window Z_h, which excludes the corrective target Y. Finally, Lines 4–7 attend to this joint memory, transform the residual query into the teacher representation, and decode it through the frozen pretrained head over eligible users. During teacher training, only the adapter parameters are updated; gradients pass through the frozen head to the adapter output.

Algorithm 3 presents one round of representation-aware on-policy self-distillation. Lines 1–2 initialize a trainable student from the frozen round-start snapshot and return it unchanged if no aligned samples are available. For each retained sample, Lines 4–7 construct the pretrained anchor and the student comparison distribution using the same eligible-user set. The teachers are then evaluated, and Lines 9–12 compute the KL-difference scores in Eq. (8), combine uniform and adaptive horizon weights, and cache the stop-gradient guidance Q_b. Finally, Lines 14–17 evaluate both student distributions and update θ using the minibatch-averaged corrective cross-entropy, policy KD, and representation KD in Eq. (10). The teacher guidance remains fixed, while the frozen pretrained head preserves gradients into the student representation.

Algorithm 4 summarizes the complete training and inference workflow. Lines 1–4 pretrain and freeze the backbone, initialize the student, and initialize horizon-specific adapters on common-support ground-truth prefixes, with adapter selection based on VALID target NLL. In each round, Lines 6–17 freeze the current student and greedily extend sampled prefixes for at most L steps. Lines 10–14 preserve the order of unseen ground-truth participants to obtain the corrective target and post-target windows; Line 15 retains states with at least one available teacher horizon. Lines 20–23 continue adapter training by corrective cross-entropy on ground-truth prefixes and the newly collected histories, then freeze the updated teacher bank. Line 24 distills its guidance into the student using Algorithm 3. Finally, Line 25 selects the earliest student checkpoint maximizing the valida-

Algorithm 1: PREFIXONLYFORWARD: Prefix-Only Backbone Execution

Input: State $X = (C, G)$; backbone parameters θ.
Output: Next-participant distribution p and features $(\mathbf{H}, s, \mathbf{E})$.

// 1. Encode the current prefix state
$\mathbf{E} \leftarrow \mathrm{GNN}_\theta(G)$
$\mathbf{H} \leftarrow \mathrm{AutoregressiveTransformer}_\theta(\mathrm{Lookup}(\mathbf{E}, C))$
$s \leftarrow \mathrm{Readout}_\theta(\mathbf{H})$
// 2. Decode over eligible users
$\mathcal{U} \leftarrow \mathrm{Eligible}(C)$
$p \leftarrow \mathrm{MaskedSoftmax}(W_\theta s; \mathcal{U})$
return $(p, \mathbf{H}, s, \mathbf{E})$

Algorithm 2: TEACHERFORWARD: Future-Informed Teacher Execution

Input: $X = (C, G)$; post-target window Z_h; frozen $\bar{\theta}$; adapter ϕ_h.
Output: Teacher distribution q_h.

// 1. Construct the state-conditioned query and memory
$(_, \bar{\mathbf{H}}, \bar{s}, \bar{\mathbf{E}}) \leftarrow \mathrm{PrefixOnlyForward}(X; \bar{\theta})$
$u_h \leftarrow u_h^{(0)} + U_h \bar{s}$
$M_h \leftarrow [\bar{\mathbf{H}} \parallel \mathrm{Lookup}(\bar{\mathbf{E}}, Z_h)]$
// 2. Attend to the joint memory and predict
$a_h \leftarrow \mathrm{Attn}(W_h^Q u_h, W_h^K M_h, W_h^V M_h)$
$s_h^{\mathrm{T}} \leftarrow F_h(u_h + a_h)$
$q_h \leftarrow \mathrm{MaskedSoftmax}(\bar{W} s_h^{\mathrm{T}}; \mathrm{Eligible}(C))$
return q_h

Algorithm 3: OPSDROUND: Representation-Aware On-Policy Self-Distillation

Input: $\mathcal{B}_k$; frozen $(\theta_{\mathrm{b}}, \bar{\theta}, \Phi_k)$; settings Ω; schedule $\mathcal{S}_{\mathrm{S}}$.
Output: Updated student parameters θ_k.

$\theta \leftarrow \mathrm{TrainableCopy}(\theta_{\mathrm{b}})$; $\mathcal{T}_k \leftarrow [\,]$
if $\mathcal{B}_k = \emptyset$ **then return** θ
// 1. Construct fixed guidance for each aligned sample
foreach $b = (X, Y, Z) \in \mathcal{B}_k$ **do**
  $(C, G) \leftarrow X$; $\mathcal{H}_b \leftarrow \mathrm{Keys}(Z)$; $\mathcal{U} \leftarrow \mathrm{Eligible}(C)$
  $(p_0, _, _, _) \leftarrow \mathrm{PrefixOnlyForward}(X; \bar{\theta})$
  $(_, _, s_{\mathrm{b}}, _) \leftarrow \mathrm{PrefixOnlyForward}(X; \theta_{\mathrm{b}})$
  $p_{\mathrm{ref}}^{(k)} \leftarrow \mathrm{MaskedSoftmax}(\bar{W} s_{\mathrm{b}}; \mathcal{U})$
  $q_h \leftarrow \mathrm{TeacherForward}(X, Z_h; \bar{\theta}, \Phi_k[h]), \quad h \in \mathcal{H}_b$
  $g \leftarrow \mathrm{StudentAwareScores}(\{q_h\}, p_0, p_{\mathrm{ref}}^{(k)})$
  $w \leftarrow (1 - \rho)\,\mathrm{Uniform}(\mathcal{H}_b) + \rho\,\mathrm{Softmax}(g/\tau_s)$
  $Q_b \leftarrow \mathrm{StopGrad}(\mathrm{WeightedMean}(\{q_h\}, w))$
  $\mathrm{Append}(\mathcal{T}_k, (X, Y, Q_b))$
// 2. Optimize the student on the stored supervision
foreach *minibatch* $(X, Y, Q) \in \mathrm{Batches}(\mathcal{T}_k; \mathcal{S}_{\mathrm{S}})$ **do**
  $(C, G) \leftarrow X$; $(p_\theta, _, s_\theta, _) \leftarrow \mathrm{PrefixOnlyForward}(X; \theta)$
  $p_{\mathrm{ref}}^{\theta} \leftarrow \mathrm{MaskedSoftmax}(\bar{W} s_\theta; \mathrm{Eligible}(C))$
  $\ell \leftarrow \mathrm{OPSDLoss}(Y, Q, p_\theta, p_{\mathrm{ref}}^{\theta}; \lambda, \mu)$
  $\theta \leftarrow \mathrm{Update}(\theta; \ell)$
return θ

tion criterion, and Lines 26–32 generate the forecast by repeatedly appending the highest-scoring eligible user. At inference, only the selected prefix-only student is used. Ground-truth alignment and future-window construction are used for training and teacher diagnostics, and are not part of the deployed forecasting procedure.

Algorithm 4: FUTUREWAY: Overall Training and Inference Workflow

Input: TRAIN $\mathcal{D}_{\mathrm{tr}}$; VALID $\mathcal{D}_{\mathrm{val}}$; $G, \mathcal{H}, K, L, \pi, \Omega, \mathcal{S}$; observed prefix C_{obs}; forecast length H.
Output: Selected prefix-only student $p_{\theta_{k^\star}}$ and forecast $\widehat{C}_{\mathrm{future}}$.

// 1. Pretrain the backbone and initialize adapters
$\bar{\theta} \leftarrow \mathrm{FrozenCopy}(\mathrm{Pretrain}(\mathcal{D}_{\mathrm{tr}}, \mathcal{D}_{\mathrm{val}}, G; \mathcal{S}_0))$
$\theta_0 \leftarrow \mathrm{TrainableCopy}(\bar{\theta})$
$\mathcal{G}_h \leftarrow \mathrm{GroundTruthSamples}(\mathcal{D}_{\mathrm{tr}}, G, h), \quad h \in \mathcal{H}$
$\Phi_0 \leftarrow \mathrm{InitializeTeachers}(\{\mathcal{G}_h\}, \mathcal{D}_{\mathrm{val}}; \bar{\theta}, \mathcal{S}_{\mathrm{T}})$
for $k = 1, \ldots, K$ **do**
  // 2. Collect and align snapshot-generated histories
  $\theta_{\mathrm{b}} \leftarrow \mathrm{FrozenCopy}(\theta_{k-1}); \quad \mathcal{B}_k \leftarrow [\,]$
  foreach $C^\star \in \mathcal{D}_{\mathrm{tr}}$ **do**
    $o \sim \pi(\cdot \mid C^\star); \quad C \leftarrow C^\star_{1:o}$
    for $t = 1, \ldots, L$ **do**
      $R \leftarrow \mathrm{FilterUnseen}(C^\star, C)$
      if $R = [\,]$ **then break**
      $X \leftarrow (\mathrm{Copy}(C), G); \quad Y \leftarrow R_1$
      $\mathcal{H}_t \leftarrow \{h \in \mathcal{H} : |R| \geq h + 1\}$
      $Z \leftarrow \{h \mapsto R_{2:h+1} : h \in \mathcal{H}_t\}$
      if $\mathcal{H}_t \neq \emptyset$ **then** $\mathrm{Append}(\mathcal{B}_k, (X, Y, Z))$
      $(p, _, _, _) \leftarrow \mathrm{PrefixOnlyForward}(X; \theta_{\mathrm{b}})$
      $C \leftarrow C \oplus \arg\max_{v \in \mathrm{Eligible}(C)} p(v)$
  // 3. Update each adapter, then freeze the teacher bank
  $\Phi_k \leftarrow \Phi_{k-1}$
  foreach $h \in \mathcal{H}$ **do**
    $\phi_h \leftarrow \mathrm{TrainableCopy}(\Phi_{k-1}[h])$
    $\mathcal{B}_{k,h} \leftarrow \mathrm{SelectHorizon}(\mathcal{B}_k, h)$
    $\phi_h \leftarrow \mathrm{TrainAdapter}(\phi_h, \mathcal{G}_h \uplus \mathcal{B}_{k,h}; \bar{\theta}, \mathcal{S}_{\mathrm{T}})$
    $\Phi_k[h] \leftarrow \mathrm{FrozenCopy}(\phi_h)$
  // 4. Distill the teacher guidance into the student
  $\theta_k \leftarrow \mathrm{OPSDRound}(\mathcal{B}_k, \theta_{\mathrm{b}}, \bar{\theta}, \Phi_k; \Omega, \mathcal{S}_{\mathrm{S}})$
// 5. Select on VALID, then forecast with the prefix-only student
$k^\star \leftarrow \min \arg\max_{1 \leq j \leq K} B_{\mathrm{val}}(\theta_j; H_{\max})$
$C \leftarrow C_{\mathrm{obs}}; \quad \widehat{C}_{\mathrm{future}} \leftarrow [\,]$
for $t = 1, \ldots, H$ **do**
  if $\mathrm{Eligible}(C) = \emptyset$ **then break**
  $(p, _, _, _) \leftarrow \mathrm{PrefixOnlyForward}((C, G); \theta_{k^\star})$
  $\hat{v} \leftarrow \arg\max_{v \in \mathrm{Eligible}(C)} p(v)$
  $C \leftarrow C \oplus \hat{v}; \quad \widehat{C}_{\mathrm{future}} \leftarrow \widehat{C}_{\mathrm{future}} \oplus \hat{v}$
return $(p_{\theta_{k^\star}}, \widehat{C}_{\mathrm{future}})$

D EXTENDED RELATED WORK

D.1 INFORMATION DIFFUSION PREDICTION

Microscopic diffusion predictors typically improve the representation used to rank the next participant. MS-HGAT models sequential higher-order interactions (Sun et al., 2022), and DisenIDP separates user and cascade factors with self-supervision (Cheng et al., 2023). RotDiff uses hyperbolic rotations to represent diffusion relations (Qiao et al., 2023). CARE retrieves related historical cascades (Zhong et al., 2024); SILN and PMRCA model sphere effects and propagation patterns, respectively (Feng et al., 2025; He et al., 2025). These approaches enrich structural or contextual evidence for participant prediction. FOREST and MINDS additionally connect microscopic prediction to cascade-level outcomes (Yang et al., 2019; Jiao et al., 2024). Most recently, GRID extends supervision to groups of future participants through a multi-label ranking formulation (Feng et al., 2026). The distinction highlighted in the Introduction is between future users as *prediction targets* and future evolution as *future-informed guidance for the current decision*. In autoregressive diffusion forecasting, each predicted participant changes the history conditioning the next decision. FUTUREWAY addresses this learning problem by constructing corrective supervision on generated histories, varying the teacher's future horizon, and transferring that guidance into a predictor that uses only

the current history and graph at inference. It complements advances in cascade representation with a training procedure designed for rollout-induced states.

D.2 On-Policy Learning and Knowledge Distillation

The source of training histories is central to sequential learning. Sequence-level distillation uses teacher-generated sequences (Kim & Rush, 2016). DAgger queries expert actions at learner-visited states (Ross et al., 2011), while GKD supplies distributional teacher feedback on student-generated sequences (Agarwal et al., 2024). These methods motivate learning on the contexts encountered by the learner. FutureWay applies this principle to cascade histories whose participant composition changes after every prediction, using corrective targets obtained by alignment with the ground-truth cascade at deviated states. LUPI and generalized distillation formalize the use of information available only during training (privileged information) (Vapnik & Izmailov, 2015; Lopez-Paz et al., 2016). OPSD conditions a self-teacher on reasoning traces available only during training, and SDFT uses a demonstration-conditioned self-teacher for on-policy learning (Zhao et al., 2026; Shenfeld et al., 2026). FutureWay shares this training/inference information asymmetry. Its diffusion-specific construction couples post-target participant windows with learner-generated histories, and weights their horizon-specific guidance against both a fixed pretrained anchor and the evolving student representation. Representation distillation provides a complementary transfer channel. FitNets supervises intermediate features (Romero et al., 2015), and OPRD aligns teacher and student hidden states along on-policy rollouts (Yang et al., 2026). FutureWay instead supervises the student's distribution under a frozen pretrained head, without hidden-state matching. This directly connects representation refinement to the distribution used for subsequent horizon comparisons. The contribution is the combination of corrective alignment, multi-horizon future-informed guidance, roundwise rollout collection, and representation-aware transfer for long-horizon diffusion prediction.

E Experimental Setup

E.1 Datasets

Twitter Hodas & Lerman (2013), Quora Sankar et al. (2020), and Weibo Zhang et al. (2013) contain timestamped cascades of distinct participants and associated social graphs. Cascades are split chronologically in an 8:1:1 ratio; splits and user vocabularies are fixed across methods and runs. TRAIN supplies supervision, VALID supports model selection, and TEST is used for final evaluation. The supplied graphs provide fixed transductive structural information. Table 9 records the dataset sizes and evaluation support.

Table 9: Dataset statistics and fixed-cohort evaluation settings. Lengths precede context truncation; coverage is the fraction of TEST included at every reported horizon.

Property	Twitter	Quora	Weibo
Users	12,627	4,578	39,043
Social links	619,262	55,698	369,920
Cascades	3,435	1,267	7,963
Mean cascade length	32.64	34.60	147.74
Training cascades	2,748	1,013	6,370
Validation cascades	343	126	796
Test cascades	344	128	797
Reported horizons	$\{3, 5, 10\}$	$\{3, 5\}$	$\{10, 15, 20\}$
Maximum horizon $H_{\max}$	10	5	20
Validation cohort size	146	114	708
Test cohort size	123	101	350
Test cohort coverage	35.76%	78.91%	43.91%

For length T_i, the observed prefix has $t_i = \max(1, \lfloor 0.3T_i \rfloor)$ participants. In each partition $\mathcal{D}$, the fixed cohort is $\mathcal{C}(\mathcal{D}, H_{\max}) = \{i \in \mathcal{D} : T_i - t_i \geq H_{\max}\}$. All reported horizons use that same cohort, controlling sample composition across horizons. The benchmark therefore measures fixed-length continuation of eligible cascades under a retrospectively defined observation budget. Quora's

maximum of five steps retains 101 TEST cascades; only 28 support ten steps. The corresponding limits are ten steps on Twitter and twenty on Weibo.

E.2 Autoregressive Evaluation and Metrics

Rollout protocol. Starting from the observed prefix, each model greedily generates one continuation of length $H_{\max}$. Scores are recomputed after appending each prediction. Padding and all previously observed/generated users are excluded, including users outside the encoder's context window. The first H predictions are evaluated against the first H users in the original ground-truth continuation; corrective training alignment does not alter this evaluation target sequence.

Coverage and relative order. For ground-truth continuation $Y_i^{(H)} = (v_{i,t_i+1}, \ldots, v_{i,t_i+H})$ and predictions $\widehat{Y}_i^{(H)}$, define

$$\mathrm{SetRecall}\,@H = \frac{1}{|\mathcal{C}|} \sum_{i \in \mathcal{C}} \frac{|\,\mathrm{set}(Y_i^{(H)}) \cap \mathrm{set}(\widehat{Y}_i^{(H)})|}{H}, \tag{16}$$

$$\mathrm{nLCS}\,@H = \frac{1}{|\mathcal{C}|} \sum_{i \in \mathcal{C}} \frac{\mathrm{LCS}(Y_i^{(H)}, \widehat{Y}_i^{(H)})}{H}, \tag{17}$$

where LCS is the length of the longest common subsequence. Cascades receive equal weight and both metrics are reported as percentages. Distinct users imply $0 \le \mathrm{nLCS}\,@H \le \mathrm{SetRecall}\,@H \le 1$ before percentage scaling. For ground-truth sequence (a, b, c, d, e) and prediction (b, c, x, d, a), with x absent from the ground-truth sequence, SetRecall is $4/5$ and nLCS is $3/5$: (b, c, d) preserves relative order without requiring positional equality.

Local diagnostics. For a distribution p_e and target y_e, NLL is $-\log p_e(y_e)$ and Hit@10 is the indicator that y_e is among the ten highest-scoring eligible users. These quantities are averaged over states, rather than cascades. Figure 2(a) compares conditioning on ground-truth prefixes and free-running histories using chronological next-user targets; its Twitter diagnostic uses 85 TEST cascades over 15 steps. Figure 2(b) instead compares corrective next-participant NLL on 1,738 common-support Twitter VALID states from 101 cascades, including the rollout starts. A common-support state has at least eleven ground-truth users not yet present in the current history, so all four post-target windows exclude the same corrective target. The independent teacher-complementarity population is specified in Appendix A.4.

Aggregation. Table 2 reports means and sample standard deviations over five independent runs, with seeds $\{2025, 2026, 2027, 2028, 2048\}$. The 10.56% average improvement gives equal weight to the 16 result columns: each contributes the relative increase over its highest non-FutureWay mean, including Backbone.

E.3 Compared Methods and Baseline Adaptation

All methods use the same splits, observed prefixes, fixed TEST cohorts, forecast horizons, and exclusion of previously observed/generated users. Table 10 summarizes their use in this evaluation. Recomputing scores after every prediction places all methods on the same free-running task; the GRID comparison concerns this adaptation of its native set-ranking formulation. Backbone is trained by next-participant cross-entropy. Teacher adapters and horizon weighting are discarded at deployment.

Table 10: Adaptation to the common autoregressive evaluation.

Method	Candidate scoring and rollout
MS-HGAT, RotDiff, SILN, PMRCA, CARE	Recompute next-user scores from the current history, select the highest-scoring eligible user, and greedily append it.
GRID	Retain multi-label ranking training; recompute dot-product user scores from each updated history and greedily append the highest-scoring eligible user.
Backbone / FUTUREWAY	Same deployed prediction architecture and per-seed pretrained initialization; FUTUREWAY additionally receives on-policy future-informed guidance.

E.4 IMPLEMENTATION AND MODEL SELECTION

Architecture. The backbone uses 128-dimensional user representations and one LightGCN propagation layer (He et al., 2020). Its structural and stage-context sequence branches each contain one causal attention block with four heads (128-dimensional keys and values per head) and a two-layer $128 \to 128 \to 128$ feed-forward network. Here, causal attention means that each position attends only to itself and earlier positions in the cascade sequence. A learned gate combines the branches before the user-prediction head; the stage-context branch aggregates three local positions. Sequence/attention dropout is 0.2 and graph-message dropout is 0.5. Pretraining retains at most 199 next-participant targets per cascade. The encoder retains the latest 200 participants during rollout; full histories remain available for masking and alignment with the ground-truth cascade. To balance efficiency and effectiveness in practice, we design four horizon-specific adapters, $\mathcal{H} = \{4, 6, 8, 10\}$, which share the frozen pretrained backbone, user embeddings, and head and follow Eq. (4). The core implementation code is provided in an anonymous repository at `https://anonymous.4open.science/r/FutureWay`.

Training. AdamW uses weight decay 10^{-2} for pretraining and teacher initialization, and zero for student refinement. Backbone pretraining and teacher initialization use learning rates up to 10^{-3}, batches of 32 cascades, and budgets of 30 epochs, with 500 warm-up updates on Quora and 1,000 on Twitter and Weibo, followed by cosine decay. Teacher initialization uses states formed from ground-truth prefixes with a complete ten-user post-target window. Each subsequent round updates the adapters on ground-truth prefixes and the current snapshot's generated histories, then freezes them for student optimization. In each round, each teacher adapter is updated for one epoch with a learning rate of 3×10^{-5}, using a 1:1 mixture of ground-truth-prefix and rollout states.

For each TRAIN cascade, one nonempty proper anchor prefix is sampled uniformly and extended greedily for at most $L = 20$ steps. States with at least one available horizon are retained. Each of $K = 5$ rounds makes one student-training pass over these states in batches of 64, with backbone learning rates $(3, 2, 1, 0.7, 0.5) \times 10^{-5}$ across rounds; the prediction head uses 0.3 times that rate. The corrective CE coefficient is one, $\lambda = \mu = 0.5$, and the weighting parameters are $\rho = 0.4$, $\tau_s = 1$. KL terms use untempered distributions with a shared candidate mask. Figure 5 varies ρ, λ, μ, L, K around these settings.

Selection. All checkpoint selection uses VALID before TEST evaluation. Backbone pretraining selects the checkpoint with the largest sum of long-horizon prediction metrics on validation. Teacher initialization selects each adapter by validation target NLL. Early-stopping patience is ten epochs for pretraining and five for teacher initialization. After all refinement rounds, select the earliest student checkpoint maximizing

$$B(H_{\max}) = \tfrac{1}{2}\big(\mathrm{SetRecall}\,@H_{\max} + \mathrm{nLCS}\,@H_{\max}\big) \tag{18}$$

on the fixed VALID cohort. The unrefined Backbone is not a candidate in this student selection. The two component metrics are reported separately on TEST.

Efficiency. Figure 4 uses NVIDIA H200 NVL GPUs and FP32 arithmetic. Panel (a) times autoregressive rollout on the fixed Twitter TEST cohort with batch size 64; panel (b) benchmarks guidance on 188 training states. Timings follow warm-up and GPU synchronization. Rollout timing includes prediction and history updates; guidance timing includes encoding, teacher prediction, weighting, and guidance transfer, excluding rollout collection, loading, and data preprocessing construction.