When Do Propagation Models Beat Cascade Summaries? A Released-Signal Audit of Text-Free Misinformation Detection

Xuan-Bach Le*
Ho Chi Minh City University of Technology (HCMUT), VNU-HCM
Ho Chi Minh City, Vietnam
lexuanbach@hcmut.edu.vn

Phat T. Tran-Truong†
Ho Chi Minh City University of Technology (HCMUT), VNU-HCM
Ho Chi Minh City, Vietnam
phatttt@hcmut.edu.vn

Son Ha Xuan
The Business School
RMIT University
Ho Chi Minh City, Vietnam
ha.son@rmit.edu.vn

Duy Nguyen†
University of Economics Ho Chi Minh City
Ho Chi Minh City, Vietnam
duynguyen.726202321440@st.ueh.edu.vn

Abstract

We audit the predictive value of propagation graphs for text-free misinformation detection. Across nine public data releases, we compare graph models with summaries of size, timing, and user profiles when both receive the same source fields. On four releases containing event lists, no evaluated graph neural network establishes an accuracy advantage over a neural network using seven summary features. The User Preference-aware Fake News Detection (UPFD) datasets provide retweet trees that record how stories spread, together with user-profile features. On UPFD-GossipCop, our BiGCN graph model selected using validation data beats the HistGBT tree-based summary model by 1.3 accuracy points (95% confidence interval: 0.5 to 2.0). This gain falls within the three-point equivalence margin used to measure benchmark resolution. Shuffling profiles reduces performance. Rewiring links while preserving each node's connection count has no resolved effect, although nearly star-shaped trees allow few such changes. UPFD-PolitiFact remains unresolved. Twitter15/16 and Weibo results are exploratory because model selection used test scores. Testing on later stories lowers macro-F1, a classification score averaged across classes, by 0.18 to 0.27 on FakeNewsNet (FNN)-PolitiFact and 0.014 to 0.083 on FNN-GossipCop. Cross-dataset tests also expose limited transfer. Our audit separates summary strength, profile dependence, and generalization.

CCS Concepts

• **Information systems** → **Web mining**; • **Computing methodologies** → *Machine learning approaches.*

Keywords

misinformation detection, propagation cascades, strong baselines, benchmark audit, temporal generalization, negative results

ACM Reference Format:
Xuan-Bach Le, Phat T. Tran-Truong, Son Ha Xuan, and Duy Nguyen. 2027. When Do Propagation Models Beat Cascade Summaries? A Released-Signal Audit of Text-Free Misinformation Detection. In *Proceedings of the Twentieth ACM International Conference on Web Search and Data Mining (WSDM '27), February 15–19, 2027, Hong Kong, Hong Kong*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3862322.3863898

1 Introduction

A story's sharing trace records events, their timing, and sometimes who shared from whom. These linked events form a *propagation cascade*, which remains observable when the text is unavailable. True and false news can spread differently [38], motivating recurrent and graph-based detectors [2, 7, 17, 34] when text-dependent approaches [30] cannot be used. MuMiN combines text with a broader social graph [22]. The central question is whether these connections add predictive value beyond summaries of size, timing, and profiles.

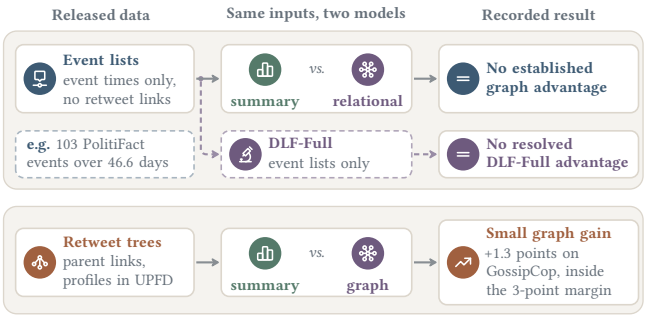

Figure 1: Released signals determine the comparison: event-list summaries versus constructed relations, and retweet-tree summaries versus graph models. DLF-Full applies only to event lists.

*Corresponding author.
†Both authors contributed equally to this research.

This work is licensed under a Creative Commons Attribution 4.0 International License.
WSDM '27, Hong Kong, Hong Kong
© 2027 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-3117-4/2027/02
https://doi.org/10.1145/3862322.3863898

Figure 1 separates two kinds of data. The upper row covers *event lists*: sharing records without observed retweet links. Relational models construct connections from the released fields, including event timing, while summary models compress the same records. Neither the graph baselines nor DocLessFake (DLF-Full), the additional diagnostic shown by the dashed branch, establishes an advantage over the summary model.

The lower row covers *retweet trees*: each link connects a retweet to its parent, and UPFD also supplies user-profile attributes. Summaries compress these fields, while graph models retain individual links and attributes. Profile shuffling and link rewiring test which inputs the models use. Near-star trees allow few degree-preserving rewirings, limiting that test.

Across nine releases, our *same-information* comparisons give both model families the same source fields. Strong summary models establish what these fields predict before we model their connections. We use input interventions to identify what graph models use, then test transfer to later stories, other datasets, and unseen publishers. We distinguish models selected on validation data from exploratory comparisons selected using test scores.

What this study adds. We test graph value beyond matched summaries (**RQ1**), the contribution of additional event-list components (**RQ2**), and transfer across evaluation populations (**RQ3**). We report macro-F1, a class-averaged score combining precision and recall, and paired accuracy tests for selected comparisons. Our CPU-runnable artifact provides code and saved outputs.[1]

Sections 2 to 4 define the comparison. Section 5 tests performance, input dependence, and transfer. Sections 6 and 7 develop the implications and position our findings in prior work.

2 Task and Available Signals

Task. We predict each story's corpus-defined label using only its released text-free signals. The event-list corpora, UPFD, and Weibo use binary labels. Twitter15/16 retain four labels: non-rumor, false rumor, true rumor, and unverified rumor, encoded as 0, 1, 2, and 3. We preserve these categories: unverified denotes annotation status, not falsehood. Macro-F1 averages over the corresponding two or four classes. Available signals include sharing events, timestamps, URLs, media, and profile attributes. Matched comparisons exclude article and post text. Separate checks use node text to audit the official implementation and titles as a reference point (Section 5.7).

Available signals. CoAID, ReCOVery, and FakeNewsNet (FNN) provide tweet or retweet identifiers; we do not fetch the original posts (hydrate the IDs). Twitter identifiers encode creation time, which we recover. Each identifier represents an event rather than a stable user, so distinct-proxy count $u \approx n$, the event count. These *event-list releases* cannot link users across cascades. User Preference-aware Fake News Detection (UPFD) [7] provides retweet trees and 10-dimensional profile attributes for each node, with official data splits. Repeated profile vectors do not establish stable user identity. The BiGCN-preprocessed Twitter15/16 [18] and Weibo [17] files provide propagation trees with bag-of-words node vectors but no event times or user identifiers. We use their structural inputs only. Together these are the *retweet-tree releases*, with profile and structure-only subgroups. In both groups, RQ1 compares a learned graph representation with per-cascade summaries derived from the same released information.

Running example. An FNN-PolitiFact story has 103 events over 46.6 days, with its tenth event arriving after 19.6 hours. These observations describe size and timing even though the release has no retweet links. Each identifier denotes an event proxy, so event count and distinct-proxy count coincide. The supplement retains its identifier, feature vector, and loader-cap details.

Summary features. For story s, we use seven features to summarize how many events occur, what they contain, and how quickly they arrive. The transform $g(x) = \log(1 + x)$ compresses large counts while allowing zeros:

$$\mathbf{c}_s = (g(n), g(u), g(r), g(m), g(P), E_{\text{hash}}, g(\text{spread})).$$

Counts n, u, r, and m denote events, distinct event proxies, URLs, and media objects. URL and media counts are constant in these releases, and proxy count duplicates event count. The remaining variation describes size, early activity, identifier concentration, and duration. Let t_0 be the first-event time. The early-popularity feature P counts distinct proxies observed before the earlier of the tenth event and t_0+1 h. A fixed hash function assigns each proxy to one of four buckets $b \in \{1, 2, 3, 4\}$. If q_b is the share of early proxies in bucket b, we define $E_{\text{hash}} = (\max_b q_b - 1/4)/(3/4)$. It is zero when early proxies are evenly spread across buckets and one when all occupy one bucket. These are identifier groups, not detected communities. Spread is the time in seconds from first to last event.

PropMLP+ augments this seven-dimensional vector with the median and interquartile range of inter-arrival times, fractions observed within 1 h and 24 h, times to the 50th and 75th event percentiles, the early-to-late distinct-proxy ratio, and a five-minute burst count. In the raw-feature ablation of supplement Table S3, the tree-based tabular models remain within ±0.01. PropMLP changes by −0.06 on CoAID and by +0.01, +0.02, and +0.02 on ReCOVery, PolitiFact, and GossipCop. The CoAID change is unresolved, and the other changes preserve the conclusions. Supplement Table S10 lists the notation.

Unit of analysis. We analyze one cascade at a time. The releases do not link translated, paraphrased, or reposted versions of a story, so they cannot establish whether separate cascades trace back to the same post. We split by *story*. For event-list releases, we also build each hypergraph inside its split, which prevents message passing across training, validation, and test stories.

3 Models Under Audit

Summary family. *PropMLP* turns seven summary features into a 64-unit hidden representation and two output scores. This two-layer multilayer perceptron (MLP, 7→64→2) uses ReLU and dropout 0.2. We standardize features using training means and standard deviations, then train with Adam (10^{-2}) for 25 full-batch epochs. *PropMLP+* uses the same architecture with 15 input features. Our tabular models use the identical features. They include logistic regression, RandomForest with 200 trees and depth 8, sklearn GradientBoosting, and a histogram gradient-boosted tree (*HistGBT*). For HistGBT, we select the learning rate, number of leaves, and ℓ_2 regularization using validation data. In the retweet-tree regime, the summary contains log size, maximum depth, log maximum width,

[1] https://github.com/lexuanbach/wsdm2027-propagation-audit

mean branching, leaf fraction, log root fan-out, mean depth, structural virality (mean pairwise tree distance), and depth per node. When profiles are available, we also include the mean, standard deviation, and maximum of each profile attribute.

Graph family. We evaluate GCNFN [20], BiGCN [2], UPFD-prop [7], RGCN [28], GATv2 [3], and an EBGCN bidirectional model [40]. In the event-list regime, each graph model uses the same connections constructed within each split from event proxies, URLs, media, and time slices. These constructed relations replace unobserved retweet edges. In the retweet-tree regime, each model receives the same released tree and node inputs. Tree GCN and GATv2 average learned node representations and append the root representation. BiGCN instead distributes raw root features within each branch before averaging. We standardize UPFD profiles using training nodes. Our GCN reaches the published UPFD result obtained with profile features. Model names denote our PyTorch implementations of the corresponding aggregation patterns under matched text-free inputs. We document verified architecture differences and unrecorded historical training settings in the supplement and artifact. RGCN and GATv2 use standard library layers and do not adapt published misinformation detectors. Our released-code comparison executes the BiGCN implementation (§5.8).

Full diagnostic configuration. DocLessFake (DLF-Full) tests four additions to cascade summaries. A hypergraph encoder learns from relations that can join several nodes. Next-slice contrastive pretraining learns to distinguish matching from nonmatching time slices. Adjustment averages predictions over proxy groups. An abstention rule withholds a prediction when the output heads disagree. We measure this disagreement by total-variation distance, half the sum of absolute differences between their predicted class probabilities. The encoder represents event proxies, URLs, media, and five-minute slices, with a residual summary branch. The current sampler chooses nonmatching examples without labels. Historical records omit this setting for some runs. When we rerun with an explicitly specified sampler, only label-free nonmatching examples reproduce all 40 historical per-seed scores (supplement). The supplement separates these pretraining tests from the five-seed results. We use hidden size 64, four heads, dropout 0.2, four pretraining epochs, and 25 fine-tuning epochs with Adam at 5×10^{-4}.

4 Experimental Setup

Table 1 records the released signals and split protocols. Counts are source stories (labeled IDs for Twitter15/16 and Weibo) followed by analyzed stories per seed. Prevalence is the raw misinformation share, and 'balanced' marks a balanced analysis. Supplement §8 provides dataset cards and the full audited signal matrix.

Event-list datasets. After balancing, CoAID [6], ReCOVery [47], and FakeNewsNet PolitiFact and GossipCop [31] contain 214, 906, 628, and 3,000 analyzed stories per seed. We use class-stratified 80/10/10 splits. Each seed determines the split and model initialization. For GossipCop, it also redraws 1,500 stories per class. The resulting test folds contain 24, 92, 64, and 300 stories per seed. Five more GossipCop draws change macro-F1 by at most 0.011.

Retweet-tree datasets. UPFD-PolitiFact and UPFD-GossipCop contain 314 and 5,464 trees under the fixed official 20/10/70 split. Their five seeds vary model initialization. Twitter15 and Twitter16 contain 1,470 and 808 four-class trees. Weibo contains 4,527 binary trees. The BiGCN snapshot `5779f2c` and audited loader determine exclusions. Twitter15 excludes 20 of 1,490 labeled IDs whose maximum parsed child index is below two, leaving 1,470. Twitter16 similarly excludes 10 of 818, leaving 808. Weibo excludes 137 of 4,664 IDs without a parsed tree, leaving 4,527. Retained class counts in order 0/1/2/3 are 373/362/366/369 for Twitter15 and 205/200/206/197 for Twitter16. Weibo has 2,278 label-0 and 2,249 label-1 trees. Exclusions precede evaluation, with source and retained-ID hashes in the artifact. For these three corpora, each seed redraws a stratified 80/10/10 split and varies initialization. Their node inputs are a depth one-hot vector, log out-degree, a root indicator, and log size.

Scores and uncertainty. We report mean macro-F1 because class balance differs across settings. For paired comparisons, we record whether each model misclassifies each story (0/1 loss) and define Δ as the model's accuracy minus its comparator's accuracy. Comparing both models on the same stories isolates their performance gap. Its confidence intervals, p-values, minimum detectable effects (MDEs), and equivalence checks all concern *accuracy points*. The drift-correction figure separately reports seed-level macro-F1 intervals over the saved runs, not story-level intervals. The archived event-list family has 23 contrasts: 20 against PropMLP, two DLF-Full+ versus PropMLP+, and one hidden-size check. Holm and Benjamini-Hochberg (BH) corrections account for testing multiple comparisons. The uncertainty calculations resample different units. Eighteen contrasts keep all predictions for the same story together when resampling, the four GATv2 contrasts resample pooled story-seed rows, and the hidden-size check resamples five seed-level differences. All use 10,000 draws. We re-estimated the six DLF-Full and DLF-Full+ contrasts with story clustering after verifying that stored per-story predictions reproduce every historical per-seed score. The GATv2 comparator's predictions were not saved, so those contrasts keep archived row-level values, which can understate uncertainty. The re-estimation leaves every Holm and BH conclusion at the 0.05 level unchanged. The retweet-tree intervals and TOST tests cluster by story. Supplement §9 maps each contrast to its source and distinguishes three event-list HistGBT configurations.

An 80%-power MDE is an effect size the test could detect with 80% probability under its assumptions. Equivalence asks a different question: is the gap small enough to fall within a chosen margin? We use two one-sided tests (TOST) with margins 0.05 and 0.03. The latter describes benchmark resolution, not deployment tolerance. It remains below the accuracy effects from 0.056 to 0.077 that the mid-sized event-list folds can resolve (§5.5). A comparison is *TOST-equivalent* when its clustered 90% interval lies inside the stated margin. It is *unresolved* when the analysis establishes neither a difference nor equivalence. A small effect can differ detectably from zero while remaining inside the chosen equivalence margin.

Our initial retweet-tree analysis selects the graph and summary family members by mean *test* macro-F1 before paired inference. We report its five intervals without selection adjustment and add a post-hoc Holm correction across corpora. Section 5.2 then repeats selection on validation data for both UPFD corpora. Twitter15, Twitter16, and Weibo retain the exploratory test-selected protocol. We describe ranking and calibration at natural prevalence separately because their displayed values come from different runs.

Table 1: Released propagation signals, corpus sizes, and analyzed splits.

Corpus	Stories: raw/analyzed (split)	Retweet tree	Node and extra features	Stable user ID	Event times	Prevalence
Event-list releases: identifiers are event proxies, not stable users						
CoAID	2,275/214 (80/10/10, 5 seeds)	no	structural, 1 URL, 0 media	no: IDs per event	yes: from IDs	5%, balanced
ReCOVery	1,603/906 (80/10/10, 5 seeds)	no	structural, 1 URL, 0 media	no: IDs per event	yes: from IDs	28%, balanced
PolitiFact (FNN)	677/628 (80/10/10, 5 seeds)	no	structural, 1 URL, 0 media	no: IDs per event	yes: from IDs	54%, balanced
GossipCop (FNN)	17,977/3,000 (80/10/10, 5 seeds)	no	structural, 1 URL, 0 media	no: IDs per event	yes: from IDs	23%, balanced
Retweet-tree releases: no event times in the files we use (Twitter15/16, Weibo: BiGCN snapshot, bag-of-words unused)						
UPFD-PolitiFact	314/314 (official 20/10/70)	yes	10-d profile attributes	no stable ID used	no	50%
UPFD-GossipCop	5,464/5,464 (official 20/10/70)	yes	10-d profiles, repeats occur	no stable ID used	no	50%
Twitter15	1,490/1,470 (80/10/10, 5 seeds)	yes	structural only	no: within-tree indices	no	4 classes, ≈25%
Twitter16	818/808 (80/10/10, 5 seeds)	yes	structural only	no: within-tree indices	no	4 classes, ≈25%
Weibo	4,664/4,527 (80/10/10, 5 seeds)	yes	structural only	no: within-tree indices	no	50%

Comparison scope and budget. We compare models that consume the released text-free information and run on six CPU cores within our recorded budget. BiGCN supplies the additional released-code audit (§5.8). The slowest DLF-Full run takes about 840 seconds per seed. Recursive tree models and large pretrained graph or language models exceed this budget.

Tuning parity. In Table 2, RGCN, GATv2, and the EBGCN model select epochs and learning rate on the validation fold, while the other graph rows use the shared fixed budget. We give learned graph baselines and PropMLP the same validation grid, retaining PropMLP's default as a fifth candidate (supplement). PropMLP then moves to 0.673 on ReCOVery, 0.671 on PolitiFact, and 0.840 on GossipCop. PolitiFact's +0.003 EBGCN point difference becomes −0.013. Tuning therefore changes the model ordering. Our later frozen-protocol study completes 2,340 trials, with eight validation-scored configurations per included model, corpus, and seed. No tuned graph baseline establishes an accuracy advantage over tuned PropMLP on these three corpora; several CoAID graph models have higher macro-F1 point estimates (Supp. §31). Of 34 orderings, 11 reverse, all unresolved. Equal trial counts do not imply equal computation time. GossipCop graph baselines and test-selected corpora are excluded (Supp. §31).

5 Results

We first compare graphs with summaries on event lists and retweet trees, then test which inputs explain the tree results. Transfer, calibration, and leakage checks establish when these findings apply. A text reference point and implementation audit place them in context.

5.1 Event-list releases

No evaluated graph model establishes an accuracy advantage over PropMLP. Table 2 groups graph baselines, summary models, and diagnostics. Rankings vary with corpus and tuning. GossipCop uses 1,500 stories per class. Uncertainty follows the resampling units in Section 4. EBGCN has higher mean macro-F1 than PropMLP on ReCOVery (0.675 vs. 0.659) and PolitiFact (0.660 vs. 0.657), but the seed-paired ReCOVery macro-F1 comparison is unresolved (p=0.50) and the PolitiFact mean ordering reverses under the partial tuning-parity grid. EBGCN has the highest graph mean on CoAID, ReCOVery, and PolitiFact. GATv2 has the highest graph mean on GossipCop.

The GCNFN and BiGCN baselines have lower means than PropMLP in all eight model-corpus comparisons under the fixed budget. PropMLP's macro-F1 differences range from +0.107 to +0.269 against GCNFN and from +0.049 to +0.321 against BiGCN. Five of eight secondary seed-level tests have $p<0.05$. Supplement §9 identifies the per-comparison output file. These tests and the EBGCN check are secondary analyses outside the 23-test Holm/BH family.

Table 2: Event-list macro-F1 for our implementations over five seeds. Subscripts: population standard deviations (ddof=0, NA unavailable). Bold: descriptive maximum. HistGBT-A: validation-tuned 15-feature run.

Method	CoAID	ReCOVery	PolitiFact	GossipCop
majority class	$0.333_{0.000}$	$0.333_{0.000}$	$0.333_{0.000}$	$0.333_{0.000}$
Graph baselines				
GCNFN [20]	$0.442_{0.109}$	$0.390_{0.086}$	$0.550_{0.028}$	$0.692_{0.180}$
BiGCN [2]	$0.539_{0.146}$	$0.338_{0.009}$	$0.536_{0.029}$	$0.787_{0.014}$
UPFD-prop [7]	$0.333_{0.000}$	$0.333_{0.000}$	$0.333_{0.000}$	$0.333_{0.000}$
RGCN [28]	$0.596_{0.110}$	$0.545_{0.064}$	$0.559_{0.022}$	$0.780_{0.015}$
GATv2 [3]	$0.598_{0.136}$	$0.597_{0.076}$	$0.627_{0.043}$	$0.797_{0.019}$
EBGCN [40]	$0.611_{0.138}$	$0.675_{0.020}$	$0.660_{0.028}$	$0.789_{0.018}$
Summary baselines				
PropMLP (7 features)	$0.618_{0.085}$	$0.659_{0.024}$	$0.657_{0.032}$	$0.836_{0.018}$
PropMLP+ (15 features)	$0.610_{0.059}$	$\mathbf{0.699}_{0.067}$	$0.717_{0.039}$	0.851_{NA}
RandomForest (7 features)	$0.595_{0.085}$	$0.651_{0.057}$	$0.667_{0.040}$	$0.865_{0.022}$
HistGBT-A (15 features)	$0.629_{0.036}$	$0.646_{0.054}$	$0.706_{0.032}$	$\mathbf{0.877}_{0.019}$
Diagnostic configurations				
DLF-Full	$\mathbf{0.654}_{0.106}$	$0.663_{0.034}$	$0.678_{0.041}$	$0.842_{0.012}$
DLF-Full+ (15 features)	$0.621_{0.067}$	$0.689_{0.047}$	$\mathbf{0.724}_{0.036}$	$0.851_{0.008}$

UPFD-prop stays at the majority floor across the tested configurations, including increased width and training duration and 120 tuning trials. EBGCN trails PropMLP on GossipCop (0.789 versus 0.836, secondary seed-paired p=0.05). DLF-Full's accuracy differences range from +0.006 to +0.033, with story-clustered p from 0.21 to 0.61. Expanding GossipCop from 500 to 1,500 stories per class changes its resolved result (p=0.017) to an unresolved comparison.

Summary and tabular results. Eight temporal-dynamics features change PropMLP accuracy by +0.064 on PolitiFact (raw p=0.009, Holm p=0.18) and +0.044 on ReCOVery (raw p=0.027, Holm p=0.52). Feeding the same vector to DLF-Full does not resolve a difference from PropMLP+ (story-clustered p=0.57 and 0.65). On GossipCop, RandomForest on seven features beats PropMLP in the clustered family ($p<0.001$). Its mean also exceeds DLF-Full's, although the saved tests do not compare this pair directly. HistGBT-A holds the highest descriptive mean (0.877). Against the separate fixed 15-feature HistGBT-C, PropMLP+ comparisons are unresolved on CoAID, ReCOVery, and PolitiFact, whereas PropMLP+ is 0.027 accuracy below HistGBT-C on GossipCop

(story-clustered p=0.0002, the bootstrap floor). HistGBT-C uses fixed settings, while Table 2's HistGBT-A is tuned. Under equal eight-trial tuning, both GossipCop gaps fall inside the 0.03 margin (0.003 and 0.008). RandomForest ranks temporal spread first on ReCOVery and PolitiFact and early popularity first on GossipCop. The constant URL and media fields receive no feature importance. Thus the same release format can preserve different predictive signals.

One feature. Logistic regression on $g(n)$ alone reaches macro-F1 0.514, 0.623, 0.496, and 0.707 across the four corpora, against PropMLP's 0.618, 0.659, 0.657, 0.836 and a 0.333 floor. Size alone approaches the full summary on ReCOVery but is near chance on PolitiFact, making all seven features a stronger cross-corpus control.

Dropping size features. Removing $g(n)$ changes PropMLP accuracy by at most 0.028 and RandomForest by at most 0.022 across the four corpora. No difference has $p<0.05$, consistent with $g(n)$ and $g(u)$ carrying nearly identical information in the event-proxy representation. Removing both changes ReCOVery and PolitiFact by at most 0.013 (all $p\geq 0.67$). GossipCop RandomForest falls by 0.034 (0.865→0.831, $p<0.001$), while PropMLP falls by 0.011 (p=0.058). CoAID PropMLP falls by 0.117 on its 24-story fold (p=0.021). Thus, the remaining features largely recover size information on the mid-sized corpora, while size adds information on GossipCop. Supplement §21 identifies the full feature-drop output.

Point-process and growth features. Six descriptors augment the 15-dimensional vector. A fitted Hawkes process models how each event can trigger later events. Its descriptors capture the background event rate, the expected number of follow-on events (branching ratio), and how quickly an event's influence decays. The other three are the time to peak hourly rate, peak-hour share, and log-log growth slope over the first 24 h. HistGBT's accuracy changes by −0.008, +0.028, +0.034, and +0.005 across the four corpora and no interval excludes zero (Table S4). These additions change some within-family rankings without changing the graph-versus-summary conclusion.

Structure versus summary. Under the identical budget (supplement Table S8), summary-only exceeds structure-only by 0.069 on CoAID, 0.107 on ReCOVery, and 0.089 on PolitiFact. Adding the hypergraph to the summary branch changes macro-F1 by +0.047 (unresolved), +0.000, and +0.007. Hidden size 128 with 8+35 epochs changes ReCOVery by +0.002 (p=0.87). In a separate exploratory three-seed CoAID ablation, label-free pretraining raises macro-F1 from 0.531 to 0.563 (0.587 when nonmatching examples have opposite labels). We next examine whether the adjustment and abstention components change predictions.

Adjustment and abstention. We first test whether changing proxy-group membership changes the adjusted prediction. In this diagnostic, adjusted and unadjusted probability vectors are identical. Their disagreement therefore supplies no signal for abstention. The behavior persists when we permute the two proxy variables or substitute a Louvain polarization diagnostic. A separate run produces disagreement between the heads (supplement). On CoAID, withholding low-confidence predictions reaches macro-F1 0.638 at coverage 0.875, compared with 0.555 at 0.792 when withholding predictions based on head disagreement. Coverage is the fraction of

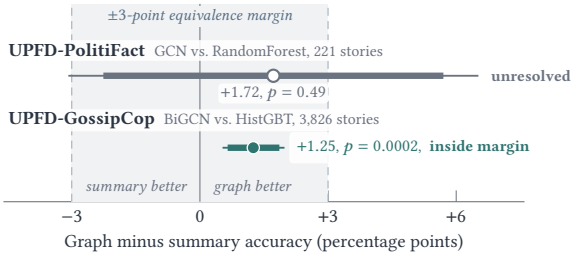

Figure 2: Validation-selected UPFD accuracy, graph minus best summary (percentage points). Thick/thin segments: story-clustered 90%/95% intervals. Shading: ±3-point equivalence margin.

stories receiving a prediction. A matched-budget ablation changes accuracy by at most 0.050 after removing pretraining, adjustment, or the hypergraph, but none of 12 contrasts survives Holm correction. At matched 90% coverage, disagreement abstention never beats confidence abstention (Supp. §31).

Validation-only selection. Primary paired tests compare against fixed PropMLP. Bold table entries identify test-score leaders. We examine selection effects retrospectively by choosing the highest five-seed validation mean from a candidate pool declared after test inspection: eight candidates on CoAID, ReCOVery, and PolitiFact, four on GossipCop. Blanking every test field reproduces the choices over 140 outputs. The rule selects PropMLP, DLF-Full+, PropMLP+, and HistGBT, respectively, never a standalone graph baseline. The hybrid's graph contribution remains unresolved. Gaps to the regenerated pool's test leader are 0.086, 0.010, 0.007, and 0.000. On CoAID the reference is inner-fold tuned HistGBT (0.703), not Table 2's DLF-Full (0.654); the latter gives a 0.036 gap (Supp. §26).

Key finding. On these event lists, graph models establish no accuracy gain over the seven-feature summary network. Better summaries and tuning can change which model leads. Test whether graph connections add value after accounting for both.

5.2 Released retweet trees

Table 3 separates validation-selected UPFD results from exploratory test-selected comparisons. Model scores are macro-F1, while paired gaps and intervals measure accuracy.

On UPFD-GossipCop, validation selects BiGCN and HistGBT. Refitting HistGBT through an inner-training-fold search gives a 1.3-percentage-point accuracy gain (95% CI 0.5 to 2.0, clustered p = 0.0002, the bootstrap floor, over 3,826 stories). Figure 2 makes the distinction visible: the gain excludes zero, but its 90% interval lies inside the shaded three-point equivalence margin. Holm over the two UPFD comparisons gives p = 0.0004. Even correction over all five corpora would give at most 0.001. On UPFD-PolitiFact, validation selects GCN and RandomForest. The +0.017 accuracy difference remains unresolved on 221 stories.

Our later eight-trial study again selects BiGCN and HistGBT on UPFD-GossipCop, with gain 0.017 (95% CI 0.010 to 0.024), inside the 0.03 margin (Supp. §31). The earlier test-selected gain is 0.009 with

Table 3: Our graph and summary implementations on retweet trees. Model scores are mean macro-F1, while gaps and clustered intervals are accuracy. Selection uses validation for UPFD and test scores for the exploratory rows. The last column gives TOST equivalence at 0.03.

Dataset	Summary	F1	Graph	F1	Accuracy gap [95% CI]	p	Within 0.03
Validation-selected							
UPFD-PolitiFact	RandomForest	0.765	GCN	0.783	+0.017 [−0.031, +0.065]	0.4864	unresolved
UPFD-GossipCop	HistGBT	0.928	BiGCN	0.940	+0.013 [+0.005, +0.020]	0.0002	yes
Exploratory: test-selected							
Twitter15	MLP	0.351	BiGCN	0.342	−0.008 [−0.044, +0.028]	0.6876	unresolved
Twitter16	MLP	0.336	GATv2	0.327	−0.015 [−0.059, +0.030]	0.5596	unresolved
Weibo	LogReg	0.889	GCN	0.894	+0.005 [−0.004, +0.015]	0.2956	yes

Table 4: BiGCN graph-input controls over five seeds (macro-F1, global profile shuffle). Asterisks mark seed-paired t-test $p < 0.05$.

Corpus	intact graph	rewired edges Δ	shuffled profiles Δ
UPFD-GossipCop	0.9404	−0.0038	−0.0658*
UPFD-PolitiFact	0.7516	+0.0038	−0.0315*
Twitter15	0.3418	−0.0175	n/a
Twitter16	0.3114	−0.0368	n/a

a different HistGBT fit. Its matrix remains separate in Supp. §33. Twitter15/16 and Weibo remain exploratory because model selection used test scores. Structure-only Twitter15/16 models reach 0.28 to 0.35 macro-F1, with unresolved selected-pair accuracy gaps. Weibo's +0.005 gap is also unresolved as a difference, though descriptively TOST-equivalent at 0.03. These selected-pair intervals do not account for test-based model selection.

Key finding. BiGCN improves UPFD-GossipCop accuracy by 1.3 points after validation selection. The gain is measurable but remains within the three-point margin. Its size matters alongside its statistical significance.

5.3 Which graph inputs matter?

We investigate the UPFD-GossipCop gain by testing profile vectors and attachment patterns separately. Changing one input at a time identifies model dependence. The graph-versus-summary comparison separately measures predictive advantage.

What summaries omit. Tree statistics omit exact attachment order, parent assignments, some subtree shapes, and individual profiles. The models also differ in representation, optimization, and tuning sensitivity.

Profile attributes versus topology. For Table 4, we retrain models after changing one input throughout the training, validation, and test graphs. Root profile vectors, structural node features, and summary inputs remain unchanged. The first control swaps edge endpoints while keeping the graph connected and preserving every node's degree, as well as node and edge counts. The cache contains 7,891 distinct tree IDs: Twitter15/16 share 165 identical trees. Of these, 7,780 pass rewiring and 111 are skipped. No corpus shows a resolved five-seed change. The second control permutes the released 10-dimensional profile vectors across all non-root nodes of the corpus, preserving every edge and the overall distribution of profile values. Summary inputs remain unchanged because recomputing them after global permutation would change each tree's profile statistics.

The two interventions produce different responses. On UPFD-GossipCop, rewiring costs the BiGCN model 0.004 (p=0.23) while permuting profiles costs 0.066 (p<0.001). Profile permutation drops macro-F1 from 0.940 to 0.875, which is below every intact summary baseline. This macro-F1 drop measures a different contrast from the 0.013 accuracy gain. Profile permutation also costs GCN 0.025 and GATv2 0.044 on UPFD-GossipCop. On UPFD-PolitiFact, a within-tree permutation moves profile vectors between positions without changing which profiles the tree contains. Its effect is unresolved for all three architectures, whereas the global permutation costs GCN, GATv2, and BiGCN 0.038, 0.053, and 0.032 (all p<0.05).

Repeated profile vectors remain attributes rather than verified identities. Tree shape limits the rewiring test: in a near-star tree, most nodes connect directly to the root, leaving few alternative connections that preserve every degree. In UPFD-GossipCop, 75.2% of nodes attach to the root, mean depth is 1.20, and rewiring changes structural virality on only 31.6% of trees.

Key finding. The graph model uses profile information: shuffling profiles lowers its score. Most nodes connect directly to the root, leaving few links that can be rewired while preserving connection counts. The value of tree shape remains unresolved under this constrained intervention.

5.4 Generalization splits

We next test whether the within-corpus ordering persists under time and corpus shift. Figure 3(a) measures feature shift with the Kolmogorov-Smirnov (KS) statistic, the largest difference between the fractions of training and future values below a given threshold. Panels (b,c) plot correction effects: points to the right of zero improve macro-F1, and intervals summarize variation across five seeds. Importance weighting (IW) gives more weight to training examples that resemble the future test distribution. We apply it only to future-test splits, so its random-split table entries are not applicable. In Table 5, Panel A uses the latest 10% of stories for testing, while Panel B places training corpora on rows and test corpora on columns. A constant predictor on a balanced binary test set has macro-F1 0.333.

Temporal. Testing on the latest 10% of stories (Table 5, Panel A) reduces PolitiFact macro-F1 by 0.18 to 0.27 but GossipCop macro-F1 by only 0.014 to 0.083. HistGBT, for example, changes from 0.706 to 0.472 on PolitiFact and from 0.877 to 0.856 on GossipCop. We exclude ReCOVery because its future test period is 99% misinformation. The size of the temporal loss therefore depends strongly on the corpus.

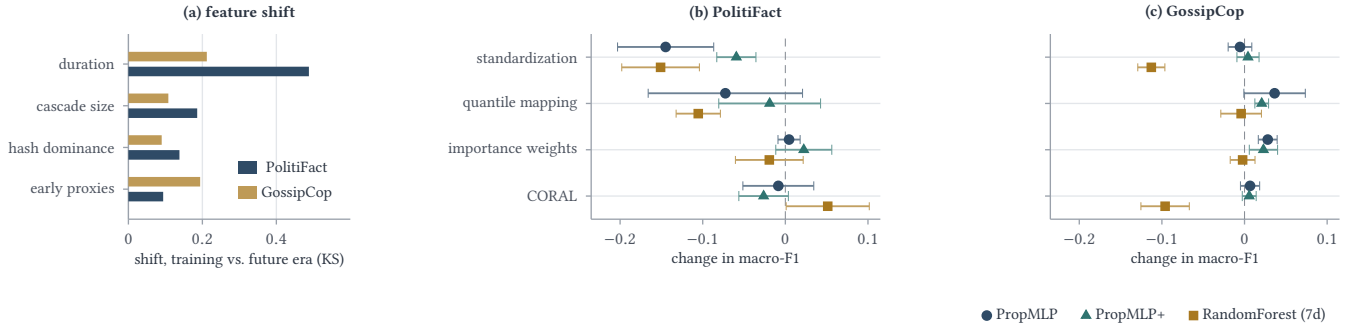

Figure 3: Future-test feature shift and paired effects of label-free corrections. KS is the largest gap between the training and future feature distributions. Positive changes mean improvement. Panels (b,c) show macro-F1 changes with paired five-seed *t* intervals (95%) over the saved runs.

Table 5: Random, future-test, and cross-corpus macro-F1 over five seeds.

(A) random vs. temporal (future-test) split

	PolitiFact (64 test)		GossipCop (300 test)	
Method	**random**	**future**	**random**	**future**
PropMLP (7 features)	0.657	0.458	0.836	0.753
PropMLP+ (15 features)	0.717	0.483	0.851	0.786
RandomForest (7 features)	0.667	0.450	0.865	0.851
RandomForest (15 features)	0.721	0.455	0.877	0.859
HistGBT (7 features)	0.651	0.416	0.862	0.828
HistGBT (15 features)	0.706	0.472	0.877	0.856
DLF-Full	0.678	0.500	0.842	0.775
HistGBT (15 features) + reweighting	n/a	0.475	n/a	0.853
RandomForest (15 features) + reweighting	n/a	0.473	n/a	0.856

(B) cross-corpus transfer (macro-F1, 5 seeds)

	train ↓ / test →	**CoAID**	**ReCOV.**	**PolitiF.**	**GossipC.**
HistGBT	CoAID	**0.629**	0.472	0.521	0.361
	ReCOVery	0.522	**0.646**	0.516	0.574
	PolitiFact	0.489	0.445	**0.706**	0.645
	GossipCop	0.387	0.494	0.414	**0.877**
PropMLP+	CoAID	**0.610**	0.449	0.522	0.349
	ReCOVery	0.548	**0.699**	0.513	0.510
	PolitiFact	0.450	0.472	**0.717**	0.595
	GossipCop	0.362	0.525	0.523	**0.851**

The loss persists when capping observation duration and moving the boundary between training and future testing through time. Some corrections improve these runs, with positive five-seed intervals before adjustment for multiple comparisons. They do not consistently recover performance across settings. Validation drawn from the training era can favor a correction that performs poorly in the future. We report the diagnostics and horizon controls in the supplement. A selector validated on a later training-era slice remains future work.

Cross-corpus. Off-diagonal macro-F1 in Table 5, Panel B, ranges from 0.35 to 0.65. Many transfers score below the 0.5 expectation of uniform random binary prediction, and GossipCop-trained models transfer at only 0.36 to 0.53. PolitiFact-to-GossipCop is the strongest direction (0.645 HistGBT, 0.613 RandomForest, 0.595 PropMLP+). Both corpora use FakeNewsNet's collection pipeline. This shared pipeline may help explain the stronger transfer, but the comparison does not isolate its effect. Overall, summary-feature relationships transfer unreliably.

Publisher and size splits. Holding out unseen publishers gives PropMLP/PropMLP+/RandomForest macro-F1 of 0.689/0.726/0.704 on PolitiFact and 0.839/0.863/0.863 on GossipCop. The point estimates show no large degradation on either FakeNewsNet corpus. Paired uncertainty intervals are needed to establish robustness to unseen publishers. Size-stratified splits are harder to interpret because size also changes prevalence. GossipCop's largest and smallest test deciles are 76% and 94% misinformation. On PolitiFact, where the corresponding misinformation shares are 52% and 34%, training on the smallest 80% and testing on the largest 10% changes PropMLP/RandomForest/HistGBT by −0.047/−0.045/−0.036 and PropMLP+ by +0.026. Supplement Table S6 reports both directions and the larger, prevalence-confounded GossipCop losses.

Key finding. Time-based evaluation exposes losses that random splits miss. Testing on future stories lowers macro-F1 by 0.18 to 0.27 on PolitiFact and 0.014 to 0.083 on GossipCop. How well propagation features transfer depends on the corpus.

5.5 Calibration and statistical resolution

At natural class prevalence, we ask both whether models rank stories correctly and whether their confidence matches accuracy. Table 6 reports area under the receiver operating characteristic curve (ROC-AUC) for ranking and expected calibration error (ECE) for confidence mismatch. Supplement Table S5 adds area under the precision-recall curve (AUPRC) and Brier score, the mean squared probability error.

Natural misinformation prevalence is 5% on CoAID, 28% on ReCOVery, 54% on PolitiFact, and 23% on GossipCop. Table 6 preserves two descriptive blocks from separate runs, namely ROC-AUC and seed-pooled ECE. ECE differs between the archived runs. HistGBT on PolitiFact changes from 0.169 to 0.118, and RandomForest on CoAID changes from 0.031 to 0.027. Within their respective runs, HistGBT has the largest GossipCop ROC-AUC (0.936) and smallest ECE (0.007). PropMLP+ has the largest ReCOVery/PolitiFact

Table 6: Descriptive ranking and calibration at natural prevalence over five seeds.

Method	CoAID	ReCOVery	PolitiFact	GossipCop
ROC-AUC (discrimination)				
PropMLP	0.504	0.655	0.732	0.900
PropMLP+	0.539	0.698	0.804	0.917
RandomForest	0.686	0.681	0.772	0.933
HistGBT	0.611	0.668	0.785	0.936
ECE (calibration-error point estimate)				
PropMLP	0.026	0.048	0.061	0.022
PropMLP+	0.027	0.058	0.054	0.025
RandomForest	0.027	0.068	0.082	0.014
HistGBT	0.033	0.094	0.118	0.007
HistGBT + isotonic	0.010	0.055	0.098	0.016

ROC-AUC (0.698/0.804), while HistGBT has ECE 0.094/0.118 there. Isotonic recalibration changes these ECE values to 0.055/0.098. To compare ranking and calibration jointly, both must be calculated from the same saved predictions.

Practical-margin sensitivity. The 0.03 accuracy margin corresponds to three additional errors per hundred stories. We use it to describe benchmark resolution. Deployment tolerance requires a separate utility study. As a retrospective sensitivity check, the validation-selected UPFD-GossipCop story-clustered 90% interval $[0.0065, 0.0186]$ lies inside margins 0.02, 0.03, and 0.05, but not 0.01. The 95% interval excludes zero, so the gain is detectable yet equivalent under these larger margins. UPFD-PolitiFact's 90% interval $[-0.0226, 0.0570]$ establishes equivalence at none of these margins. These checks use the saved paired predictions and change no selected model.

Power and equivalence. Event-list test folds contain 24 CoAID, 92 ReCOVery, 64 PolitiFact, and 300 GossipCop stories. CoAID accuracy MDEs are 0.12 to 0.13, while the mid-sized folds resolve effects of approximately 0.056 to 0.077. RandomForest and PropMLP are TOST-equivalent at 0.05 on ReCOVery and PolitiFact, while DLF-Full's accuracy differences from PropMLP remain below the MDE. In the test-selected retweet-tree analysis, UPFD-GossipCop and Weibo are TOST-equivalent at 0.03. The other three comparisons establish neither difference nor equivalence. Their MDEs are 0.080, 0.051, and 0.063, compared with 0.010 for UPFD-GossipCop and 0.014 for Weibo. The archived mixed-unit Holm/BH family retains GossipCop RandomForest and fixed 7-feature HistGBT-B improvements over PropMLP, and GATv2's deficit. The latter uses row resampling and is not a cluster-validated finding. PropMLP+'s PolitiFact difference has Holm p=0.18. Across five retweet-tree comparisons, the smallest nominal value (p=0.011 on UPFD-GossipCop) becomes 0.053 under post-hoc Holm correction.

Key finding. Small test sets leave more uncertainty about a performance gap. Larger tree test sets can establish equivalence within three points. Similar scores alone do not establish similar performance.

5.6 Early detection and leakage

To identify the first k% of sharing events, we must know how large the cascade eventually becomes. This provides a retrospective view of partial cascades, but a detector cannot use that future information when a story arrives. Under this diagnostic, CoAID reaches its maximum by 75% of events, whereas ReCOVery requires the full cascade. Training and testing on the same truncated representation changes ReCOVery PropMLP from 0.659 to 0.657 and removes PropMLP+'s temporal-feature advantage on PolitiFact. Operational early detection instead requires a fixed observation window.

Fixed wall-clock windows avoid conditioning on eventual size. At 1, 2, and 7 days, PolitiFact PropMLP obtains 0.692, 0.685, and 0.661. RandomForest obtains 0.720, 0.698, and 0.666. The corresponding GossipCop values are 0.828, 0.821, and 0.821 for PropMLP and 0.862, 0.850, and 0.853 for RandomForest. These five-seed estimates describe folds of 64 and 300 test stories. Confidence intervals are needed to resolve the effect of observation window.

Of 580 CoAID/ReCOVery test stories, 3 share an event proxy with training, and none shares a URL or media node. Table 7 reports a label-permutation negative control whose protocol and interpretation rule we fixed before execution. We shuffle training labels within each fold while keeping splits, seeds, features, and *test* labels fixed. This breaks the association between training features and labels. All eight permuted means lie between 0.467 and 0.529, and each five-seed interval contains 0.50. Compared with the same script using real training labels, scores fall by 0.12 to 0.36. We report three exploratory diagnostics with two to three seeds separately. Training-label noise reduces ReCOVery macro-F1 by 0.18 at 10% and 0.07 at 20%. The three-seed curve is non-monotone. DLF-Full is 20 to 44× slower than GCNFN and uses approximately 1.5 GB of resident memory.

Table 7: Label-permutation negative control over five seeds (accuracy). HistGBT-C uses 15 features, while RandomForest uses 7. Real and shuffled labels belong to this control's matched runs, not Table 2.

Corpus	Model	Test n	Real labels	Shuffled labels [95% CI]
CoAID	HistGBT-C	24	0.704	0.467 [0.392, 0.541]
CoAID	RForest	24	0.613	0.469 [0.272, 0.666]
ReCOVery	HistGBT-C	92	0.670	0.506 [0.450, 0.561]
ReCOVery	RForest	92	0.657	0.483 [0.383, 0.584]
PolitiFact	HistGBT-C	64	0.698	0.498 [0.410, 0.587]
PolitiFact	RForest	64	0.645	0.529 [0.392, 0.665]
GossipCop	HistGBT-C	300	0.878	0.518 [0.453, 0.583]
GossipCop	RForest	300	0.862	0.518 [0.423, 0.614]

Key finding. Early detection must use information available at the time, such as a fixed observation window. Shuffling training labels moves performance toward chance. This control tests label dependence, alongside the separate checks for shared events and nodes across splits.

5.7 A text reference point

We test the contribution of story titles in a separate FakeNewsNet run. Logistic regression uses term-frequency/inverse-document-frequency (TF-IDF) features, which weight words by their frequency in a title and rarity across titles. It reaches 0.790 macro-F1 on PolitiFact and 0.765 on GossipCop. In the same run, the 15-feature summaries (logistic regression, HistGBT) score 0.743/0.706 on PolitiFact and 0.848/0.872 on GossipCop. Fusing titles with summaries reaches 0.811 and 0.875.

Key finding. Titles work better on PolitiFact, propagation summaries on GossipCop. Combining them helps in both measured splits. The two sources provide complementary information, with their value depending on the collection setting.

5.8 Implementation scope

Original BiGCN and EBGCN use TF-IDF node vectors and root text. Our structural adaptations reach 0.28 to 0.35 macro-F1 on four-class Twitter15/16 and 0.88 to 0.89 on binary Weibo. Isolating the contribution of text requires fixed implementation and training. Original UPFD uses user-history preferences and news text. Our GCN is consistent with its reported profile-feature result.

Implementation differences. We implement the tabulated graph models locally, adapting published detectors and including generic RGCN/GATv2 baselines. The official BiGCN run audits inputs and selection. We document source-level BiGCN differences in root readout, dropout, edge dropping, and optimization, and GCNFN/UPFD and EBGCN differences in operators and inputs (Supp. §29).

Running the released implementation. We execute the BiGCN release at commit `5779f2c`. External compatibility shims forward one scatter call, restore a NumPy array alias, and set the process start method. The authors' source and preprocessor stay unchanged. The five-fold release provides training and test data but no validation fold, so early stopping selects epochs on test data. It also reads 5,000-dimensional TF-IDF node vectors, so an as-released run differs from our text-free adaptation in inputs and implementation.

The releases provide no verified recurring user identities for global hypergraph methods such as HG-SL. Repeated UPFD profiles do not establish identity. Likewise, P and E_{hash} describe proxies rather than established confounders. Their GossipCop RandomForest importances are 0.318 and 0.097, yet changing the proxy groups leaves the adjustment head unchanged. That diagnostic produces no disagreement, whereas separate runs do. Supplement Table S9 summarizes the input and mechanism differences.

Excluded baselines. FANG [21] and HG-SL [34] need cross-story social links or recurring user identities absent here. Forest-level methods require multiple cascades per story. Recursive tree models and GPU-scale pretrained graph models exceed our CPU budget.

6 Discussion

We establish matched summaries as a strong starting point for evaluating graph models. Our UPFD-GossipCop interventions identify profile dependence: shuffling profiles hurts performance. Near-star trees constrain rewiring, leaving the contribution of tree shape unresolved. The 0.03 margin describes benchmark resolution, not a universal threshold for useful improvement.

Our transfer tests make future-story evaluation essential. Calibration uses separate runs, and publisher-held-out estimates need paired intervals. ReCOVery's 99.3% misinformation future slice and CoAID's 24-story fold, with MDEs of 0.12 to 0.13, constrain temporal conclusions.

Section 5.8 details the release and implementation boundaries. Released-code validation covers BiGCN, not official UPFD, GCNFN, or EBGCN. The next comparisons need equal-compute tuning across nine corpora, matched-input implementations, independent retraining, and fresh tests for the three test-selected corpora. Data that preserve identity, profiles, and timing together would let us vary these signals separately. Validation under shift would then test whether the resulting advantage survives transfer.

7 Related Work

Text-centric and multimodal detectors. Detectors use post streams [17], comment-based explanations [30], dual emotion [46], text-response modeling [27], or LLM-generated reactions and explanations [39]. Their text inputs distinguish them from our matched comparisons.

Propagation and graph detectors. Propagation methods classify cascade structure through tree kernels [18], campaign augmentation [19], interaction attention [14], geometric learning [20], bidirectional convolution [2], co-attention [16], social-context graphs [21], user preferences [7], edge uncertainty [40], or global-local spreading [34]. These methods use standard graph encoders [3, 11, 15, 28, 36] with different input requirements. Our adaptations compare representations of the same released fields.

Diagnostic components. DLF-Full combines a hypergraph encoder [9, 43] with a next-slice pretext from contrastive learning [4, 12, 35, 37, 45]. We test their incremental predictive value. The data-dependent negative sampler has no mutual-information guarantee, and its comparison remains unresolved.

Simple controls. Tuning budgets, split protocols, and weak baselines can inflate graph gains [8, 29]. Removing nonlinearity can cost little [41], while label propagation with an MLP can rival GNNs [13]. We test these concerns under the input constraints of public releases.

Benchmarks and releases. News-plus-social-context repositories [31], COVID-19 collections [6, 47], and multilingual graphs [22] preserve different signals. Redistribution can leave only identifiers from richer platform records. Real diffusion asymmetries [38] therefore need not remain measurable in released data.

Diagnostics. We draw on backdoor adjustment [23–25], unobserved-confounding sensitivity [26, 44], selective classification [5, 10], importance weighting [32], CORAL [33], and invariance objectives [1, 42]. We test their predictive value with matched controls and input interventions.

8 Conclusion

Our matched comparisons establish no graph accuracy advantage on event lists and a validation-selected 1.3-point gain on UPFD-GossipCop, within the three-point margin. Our interventions identify profile dependence, while the value of tree shape remains unresolved. Transfer tests expose further losses. Together, our results define how to assess graph value: match inputs and tuning, intervene on model inputs, and test transfer under declared selection and uncertainty protocols.

Acknowledgments

We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, for supporting this study.

Ethical Considerations

We use public releases under their original terms without hydrating tweets, collecting user data, or attempting re-identification. Identifiers serve only as event proxies, and UPFD provides profile vectors rather than raw profiles. Before platform use, these models require in-domain validation, recalibration, human review, and monitoring of false positives on fast-spreading legitimate stories. Protected-group performance requires subgroup evaluation beyond these corpus-level results. We retain negative results in the artifact.

References

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893* (2019). doi:10.48550/arXiv.1907.02893

[2] Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 549–556. doi:10.1609/aaai.v34i01.5393

[3] Shaked Brody, Uri Alon, and Eran Yahav. 2022. How attentive are graph attention networks?. In *International Conference on Learning Representations*. https://openreview.net/forum?id=F72ximsx7C1

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 1597–1607. https://proceedings.mlr.press/v119/chen20j.html

[5] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. 2016. Learning with rejection. In *Algorithmic Learning Theory*. Springer, 67–82. doi:10.1007/978-3-319-46379-7_5

[6] Limeng Cui and Dongwon Lee. 2020. CoAID: COVID-19 healthcare misinformation dataset. *arXiv preprint arXiv:2006.00885* (2020). doi:10.48550/arXiv.2006.00885

[7] Yingtong Dou, Kai Shu, Congying Xia, Philip S Yu, and Lichao Sun. 2021. User preference-aware fake news detection. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2051–2055. doi:10.1145/3404835.3462990

[8] Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli. 2020. A Fair Comparison of Graph Neural Networks for Graph Classification. In *International Conference on Learning Representations*. https://openreview.net/forum?id=HygDF6NFPB

[9] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. 2019. Hypergraph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3558–3565. doi:10.1609/aaai.v33i01.33013558

[10] Yonatan Geifman and Ran El-Yaniv. 2017. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems 30*. 4878–4887. https://proceedings.neurips.cc/paper/2017/hash/4a8423d5e91fda00bb7e46540e2b0cf1-Abstract.html

[11] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30*. 1024–1034. https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html

[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9726–9735. doi:10.1109/CVPR42600.2020.00975

[13] Qian Huang, Horace He, Abhay Singh, Ser-Nam Lim, and Austin R. Benson. 2021. Combining Label Propagation and Simple Models Out-performs Graph Neural Networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=8E1-f3VhX1o

[14] Ling Min Serena Khoo, Hai Leong Chieu, Zhong Qian, and Jing Jiang. 2020. Interpretable rumor detection in microblogs by attending to user interactions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 8783–8790. doi:10.1609/aaai.v34i05.6405

[15] Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SJU4ayYgl

[16] Yi-Ju Lu and Cheng-Te Li. 2020. GCAN: Graph-aware co-attention networks for explainable fake news detection on social media. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 505–514. doi:10.18653/v1/2020.acl-main.48

[17] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. 3818–3824. https://www.ijcai.org/Proceedings/16/Papers/537.pdf

[18] Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 708–717. doi:10.18653/v1/P17-1066

[19] Jing Ma, Wei Gao, and Kam-Fai Wong. 2019. Detect rumors on Twitter by promoting information campaigns with generative adversarial learning. In *The World Wide Web Conference*. ACM, 3049–3055. doi:10.1145/3308558.3313741

[20] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M Bronstein. 2019. Fake news detection on social media using geometric deep learning. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*. https://rlgm.github.io/papers/34.pdf

[21] Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. 2020. FANG: Leveraging social context for fake news detection using graph representation. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. ACM, 1165–1174. doi:10.1145/3340531.3412046

[22] Dan Saattrup Nielsen and Ryan McConville. 2022. MuMiN: A large-scale multilingual multimodal fact-checked misinformation social network dataset. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 3141–3153. doi:10.1145/3477495.3531744

[23] Judea Pearl. 2009. *Causality: Models, Reasoning, and Inference* (2 ed.). Cambridge University Press. doi:10.1017/CBO9780511803161

[24] Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. 2016. *Causal Inference in Statistics: A Primer*. Wiley.

[25] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press. https://mitpress.mit.edu/9780262037310/elements-of-causal-inference/

[26] Paul R Rosenbaum. 2002. *Observational Studies* (2 ed.). Springer. doi:10.1007/978-1-4757-3692-2

[27] Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. CSI: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM International Conference on Information and Knowledge Management*. ACM, 797–806. doi:10.1145/3132847.3132877

[28] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web (Lecture Notes in Computer Science)*. Springer, 593–607. doi:10.1007/978-3-319-93417-4_38

[29] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. 2018. Pitfalls of Graph Neural Network Evaluation. In *Relational Representation Learning Workshop, NeurIPS*. https://arxiv.org/abs/1811.05868

[30] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. dEFEND: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 395–405. doi:10.1145/3292500.3330935

[31] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2020. FakeNewsNet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big Data* 8, 3 (2020), 171–188. doi:10.1089/big.2020.0062

[32] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. 2007. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research* 8 (2007), 985–1005. https://www.jmlr.org/papers/v8/sugiyama07a.html

[33] Baochen Sun, Jiashi Feng, and Kate Saenko. 2016. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30. 2058–2065. doi:10.1609/aaai.v30i1.10306

[34] Ling Sun, Yuan Rao, Yuqian Lan, Bingcan Xia, and Yangyang Li. 2023. HG-SL: Jointly learning of global and local user spreading behavior for fake news early detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 5248–5256. doi:10.1609/aaai.v37i4.25655

[35] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *arXiv preprint arXiv:1807.03748* (2018). doi:10.48550/arXiv.1807.03748

[36] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=rJXMpikCZ

[37] Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2019. Deep graph infomax. In *International Conference on Learning Representations*. https://openreview.net/forum?id=rklz9iAcKQ

[38] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science* 359, 6380 (2018), 1146–1151. doi:10.1126/science.aap9559

[39] Herun Wan, Shangbin Feng, Zhaoxuan Tan, Heng Wang, Yulia Tsvetkov, and Minnan Luo. 2024. DELL: Generating reactions and explanations for LLM-based misinformation detection. In *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics, 2637–2667. doi:10.18653/v1/2024.findings-acl.155

[40] Lingwei Wei, Dou Hu, Wei Zhou, Zhaojuan Yue, and Songlin Hu. 2021. Towards propagation uncertainty: Edge-enhanced Bayesian graph convolutional networks for rumor detection. In *Proceedings of the 59th Annual Meeting of the Association*

for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. Association for Computational Linguistics, 3845–3854. doi:10.18653/v1/2021.acl-long.297

[41] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying Graph Convolutional Networks. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*. PMLR, 6861–6871. https://proceedings.mlr.press/v97/wu19e.html

[42] Ying-Xin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. 2022. Discovering invariant rationales for graph neural networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=hGXij5rfiHw

[43] Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha Talukdar. 2019. HyperGCN: A new method for training graph convolutional networks on hypergraphs. In *Advances in Neural Information Processing Systems 32*. https://proceedings.neurips.cc/paper/2019/hash/1efa39bcaec6f3900149160693694536-Abstract.html

[44] Steve Yadlowsky, Hongseok Namkoong, Sanjay Basu, John Duchi, and Lu Tian. 2022. Bounds on the conditional and average treatment effect with unobserved confounding factors. *The Annals of Statistics* 50, 5 (2022), 2587–2615. doi:10.1214/22-AOS2195

[45] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. In *Advances in Neural Information Processing Systems 33*. 5812–5823. https://proceedings.neurips.cc/paper/2020/hash/3fe230348e9a12c13120749e3f9fa4cd-Abstract.html

[46] Xueyao Zhang, Juan Cao, Xirong Li, Qiang Sheng, Lei Zhong, and Kai Shu. 2021. Mining dual emotion for fake news detection. In *Proceedings of the Web Conference 2021*. ACM, 3465–3476. doi:10.1145/3442381.3450004

[47] Xinyi Zhou, Apurva Mulay, Emilio Ferrara, and Reza Zafarani. 2020. ReCOVery: A multimodal repository for COVID-19 news credibility research. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. ACM, 3205–3212. doi:10.1145/3340531.3412880