Temporal Trust & Sentiment Dynamics in Truth Social

Deep Learning for Social Analytics

Vincent Ridder [1] Johann Strunck [2] Sargunpreet Kaur [3] Yunus Aras [4]

vincent.ridder@tuhh.de,
johann.strunck@tuhh.de,sargunpreet.kaur@tuhh.de,yunus.aras@tuhh.de

Contents

Abstract

This paper studies the temporal and structural dynamics of untruthful behavior on Truth Social. Moving beyond post-level classification, we introduce a rolling user-level truthfulness measure that captures short-term behavioral persistence across time. Empirical analysis reveals wave-like growth patterns, pronounced local clustering of untruthful users, and strong exposure-dependent transition risks: users following untruthful accounts exhibit substantially higher probabilities of becoming untruthful themselves, particularly during early expansion phases.

To model these dynamics, we propose a Temporal Graph Neural Network (TGNN) that jointly integrates semantic text representations, behavioral features, recurrent temporal updates, and graph-based message passing. Without relying on prior truthfulness labels, the model achieves a Macro-F1 score of 0.855, demonstrating that structural and content-based signals alone provide substantial predictive power. Incorporating past user-level truthfulness further improves performance to a Macro-F1 of 0.910, highlighting strong temporal persistence in behavioral patterns.

Our findings emphasize that untruthful behavior in polarized online environments is not purely individual but emerges from the interaction of semantic content, temporal reinforcement, and network exposure. The proposed framework offers both descriptive insight into behavioral diffusion and a scalable foundation for network-aware moderation strategies. All code and implementation details are publicly available in our GitHub repository: `truegraphdynamics2026`.

1 Introduction

Social media can be conceptualized as a digitally mediated social space in which individuals and groups interact, exchange information, and construct identities. Unlike traditional mass media, these platforms enable many-to-many communication, allowing users to simultaneously act as content creators, distributors, and consumers within a decentralized and participatory ecosystem. In order to maintain a trustworthy and non-discriminatory environment, content moderation has become a central and ongoing concern for platform operators. However, the rapid dissemination and algorithmic amplification of content also create conditions in which untruthful postings can spread widely before they are detected or corrected. Such content can distort public discourse, undermine trust, and pose significant challenges for effective moderation and automated classification systems.

The social media platform Truth Social was launched in February 2022 by former U.S. president Donald Trump and his Trump Media & Technology Group as an alternative to mainstream networks following his bans from those platforms after the January 6 Capitol attack. It was marketed as a "free speech" space meant to welcome users who felt censored by larger platforms, yet its content moderation policies have been inconsistent and at times more permissive of extreme or misleading content than those of comparable services.

In this paper, we investigate methods to restore informational integrity within highly polarized and hostility-prone online networks. Our initial baseline approach relied on post-level flagging to classify untruthful behavior, yielding limited success. Given access to the structural properties of the underlying social network, we subsequently examined the expressive power of network topology in identifying users who disseminate false statements. This led to the development of a rolling, user-level truthfulness measure, shifting the focus from individual posts to longitudinal user behavior.

Evaluation of this network-based approach demonstrated that interaction patterns between users contain significant signals related to the emergence and reinforcement of untruthful behavior. Building on these findings, we explored a temporal graph neural network framework to model the dynamic evolution of user interactions. To enrich the model's contextual understanding, we incorporated comment-level feature engineering, extracting categorical, sentiment, engagement, and clustering attributes, which were then used to predict user-level truthfulness at each time step.

2 Data

The dataset used in this paper is derived from the publicly released corpus described in Gerard et al. [2023]. Because Truth Social does not provide a public API, data were collected via automated web scraping of publicly accessible user profiles.

The crawl began with the account `@realDonaldTrump` and expanded through follower relationships in a breadth-first manner. For each user, the dataset includes:

- User metadata (e.g., follower and following counts),
- Directed follower relationships,
- Authored posts ("Truths") and associated interactions.

The collection was conducted between September 4 and October 14, 2022, resulting in a network of 65,536 users along with their historical posts available at crawl time.

2.1 Data Model and Graph Construction

The data were stored in a relational schema linking users, posts, and interaction metadata. From this structure, we construct a directed temporal graph $\mathcal{G} = (V, E, \mathcal{T})$, where:

- V denotes the set of users,
- E denotes directed follower relationships,
- $\mathcal{T}$ denotes discrete weekly time intervals.

Because the full platform network is not publicly accessible, $\mathcal{G}$ represents an induced subgraph over the observed users. The follower structure E is treated as static over the observation period, while each node $v \in V$ is associated with time-indexed activity features derived from posting behavior.

This formulation enables joint analysis of structural position and temporal user activity.

Network Structure. The follower network exhibits strong heterogeneity in connectivity. While most users maintain relatively few connections, a small fraction accumulates disproportionately large follower counts. This centralization implies unequal exposure and influence potential across the network.

User Activity. Posting behavior is similarly skewed. A minority of accounts generates a substantial share of total content, whereas many users post only sporadically. Consequently, aggregate behavioral patterns may be driven by a limited subset of highly active users.

Temporal Activity. To characterize aggregate engagement dynamics, we compute the total number of posts per week (1). Weekly activity fluctuates over time, indicating non-stationary behavior and motivating temporally aware modeling approaches.

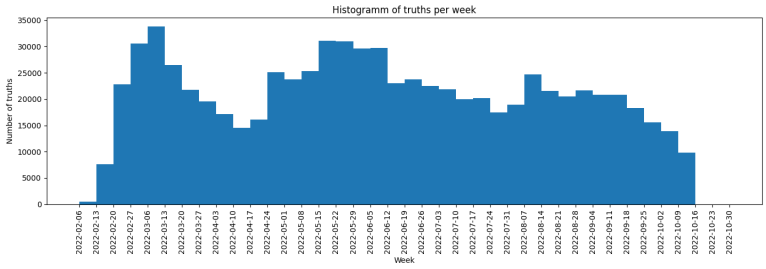


Figure 1: Number of posts per week over the observation period.

Preprocessing. To ensure structural and temporal consistency, we apply the following filters:

- Removal of users without follower or following relationships,
- Removal of users without recorded posts,
- Removal of posts with invalid or missing timestamps.

After filtering, all retained users are embedded in a structurally connected subgraph and associated with temporally valid activity records. The resulting dataset forms the basis for subsequent analysis.

Before modeling user-level temporal dynamics, we first require reliable post-level truthfulness signals from which higher-level behavioral measures can be derived. We therefore begin with a baseline approach that operates directly at the post level, aiming to identify false statements within individual pieces of content. This initial step establishes the labeling foundation necessary for subsequent user-level aggregation and temporal modeling.

3 Baseline Models for False Statement Detection at the Post Level

Detecting false statements in social media posts is inherently challenging. Determining factual accuracy often requires deep contextual understanding and background knowledge. Statements can be partially true, misleading, or dependent on specific temporal, cultural, or scientific contexts.

Social media posts frequently include sarcasm, exaggeration, or ambiguous phrasing, further complicating automated detection. Even human annotators face difficulties, as verifying claims may require expert knowledge or reliable external sources, making the process time-consuming and resource-intensive.

To address these challenges, we propose a two-stage baseline pipeline. First, a lightweight statement classifier preselects posts likely to contain factual content. Second, a truth classifier estimates the probability that each post is true or false. Based on predefined probability thresholds, posts are either automatically classified or forwarded to a more powerful LLM or human experts for final verification.

3.1 Obtaining Labels

Since our dataset lacks labels, we used ChatGPT-5 to automatically generate them. ChatGPT-5 was chosen for its improved reasoning, contextual understanding, and factual consistency compared to earlier versions, making it well-suited for generating high-quality labels. Labels (`TRUE`, `FALSE`, `NO_STATEMENT`) were created in three steps:

1. Posts from the first three timestamps were labeled using ChatGPT-5 and used to train the statement classifier.
2. The classifier was applied to prefilter posts, afterwards ChatGPT-5 generated soft probability distributions over the three classes for each post. These distributions were used to train the truth classifier.
3. A streamlined API call with ChatGPT-5 produced labels for the remaining prefiltered posts.

The cost of API calls depends on the number of tokens sent and returned. Returning full probability distributions is more expensive; the total cost of the labeling process was €70.

3.2 Statement Classifier

The initial labeled set contained 75,129 posts, of which only 6% were factual statements. To focus on these, we fine-tuned a binary BERT-based uncased model with a LoRA adapter to separate statements from non-statements. The classifier achieved a precision of 0.3 and a recall of 0.89 for the statement class. While only one in three predicted statements was correct, this reduced the dataset size by roughly 82%, producing a higher proportion of factual statements for training the truth classifier.

3.3 Truth Classifier

To train the truth classifier, posts prefiltered by the statement classifier were labeled with soft probability distributions over `FALSE`, `TRUE`, and `NO_STATEMENT` using ChatGPT-5. These soft labels were used to fine-tune a transformer-based sequence classification model (`DeBERTa-v3-base`).

Each training instance was represented as a normalized label distribution rather than a hard target. The model was trained using Soft Focal Loss to mitigate class imbalance, with class weights derived from the average label distribution in the training set, over 10 epochs.

Evaluation on a validation set of $5,004$ samples yielded an overall accuracy of 79.1% and a macro-F1 score of 0.653. Performance on the dominant class (`NO_STATEMENT`) was strong (F1 $= 0.869$), while the minority classes showed moderate performance (`TRUE`: $F1 = 0.528$; `FALSE`: $F1 = 0.561$). This demonstrates robust detection of non-statements and weak discrimination between true and false statements despite substantial class imbalance.

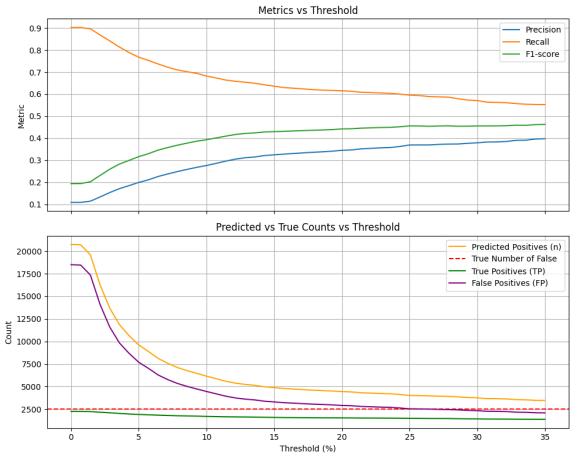


Figure 2: Precision-recall trade-off based on thresholding the predicted false probability.

To flag a post as false, we applied a threshold on the predicted false probability. Figure 2 illustrates precision, recall, and F1-score as a function of this threshold. Higher thresholds increase precision at the cost of recall, allowing the pipeline to flexibly balance false positive reduction and the retention of true false statements. Selecting an intermediate threshold efficiently filters posts, reducing the volume forwarded to computationally expensive final verification while maintaining detection quality. To achieve

a meaningful recall of 70%, this pipeline step would have a precision of approximately 25%, meaning that for each correctly flagged false statement, there would be roughly three false positives. Combining both steps of the pipeline further reduces both recall and precision.

While the model could likely be improved with more training data, we focus on investigating how network information, as well as other text-based and cluster-based features, could enhance detection. Since our main interest is identifying users who spread false statements, we shift our approach to flagging users, as described in the next section.

4 Empirical Characterization of Truthfulness Dynamics

We first define a user-level temporal truthfulness measure and then analyze its structural and temporal behavior within the follower network.

4.1 Temporal User-Level Truthfulness

Let $c_{u,i}$ denote a post authored by user u in time interval i, and let $\ell(c_{u,i}) \in \{0, 1\}$ be its binary truthfulness label, where 0 represents a truthful post and 1 otherwise.

To capture short-term behavioral persistence rather than isolated posts, we define a rolling aggregation over three consecutive intervals. For each user u and time t, we compute:

$$y_u(t) = \min\left(\frac{1}{3}\sum_{i=t-2}^{t}\sum_{c\in C_u(i)}\ell(c),\, 1.0\right), \tag{1}$$

where $C_u(i)$ denotes the set of posts authored by user u in interval i.

The resulting score $y_u(t) \in [0, 1]$ represents the recent proportion of untruthful content, smoothed over time. This rolling formulation reduces noise from single posts and reflects short-term behavioral tendencies.

For graph-level analysis, we derive a binary state variable:

$$d_u(t) = \begin{cases} 1 & \text{if } y_u(t) = 1.0 \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Users with $d_u(t) = 1$ are considered fully untruthful within the recent window.

The temporal sequence $\{d_u(t)\}$ enables analysis of prevalence, clustering, and transition dynamics.

4.2 Temporal Evolution of Untruthful Behavior

Untruthful prevalence exhibits a two-stage growth pattern, with an initial rapid increase followed by a second, higher peak and subsequent gradual decline. The sharp expansion phase suggests temporally correlated transitions rather than independent behavioral shifts, which can be seen in Figure 3.

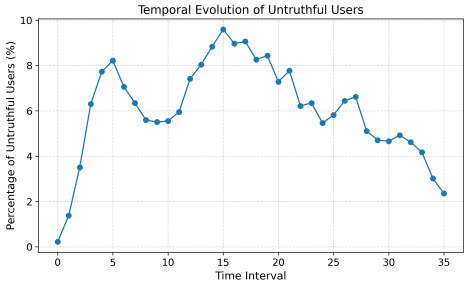


Figure 3: Percentage of untruthful users over time.

4.3 Structural Concentration

To assess whether untruthful users cluster in the network, we compute (i) assortativity with respect to $d_u(t)$ and (ii) the fraction of edges connecting two untruthful users, which can be seen in Figure 4.

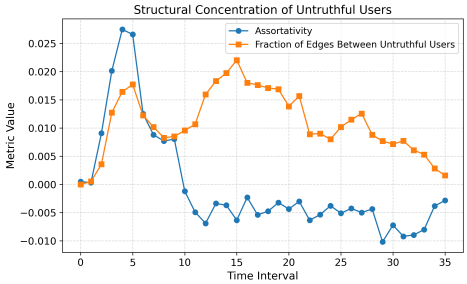


Figure 4: Assortativity and fraction of edges between untruthful users over time.

During the expansion phase, the fraction of within-group edges increases

substantially, and the induced subgraph of untruthful users forms a dominant connected component. This indicates localized structural concentration.

At later intervals, global assortativity becomes slightly negative while the within-group edge fraction remains elevated. This suggests that untruthful users form cohesive regions without complete global segregation of the network.

4.4 Exposure and Transition Risk

To quantify exposure effects, we compute the risk ratio of the probability of transitioning from truthful to untruthful in interval $t+1$, conditional on following at least one untruthful user at time t to the probability of transitioning from truthful to untruthful in interval $t+1$, conditional on following no untruthful user at time t.

Figure 5 shows the resulting risk ratio over time.

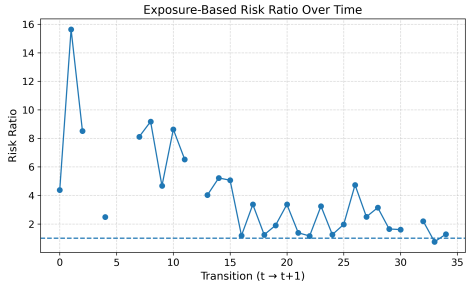


Figure 5: Risk ratio of becoming untruthful given exposure to untruthful neighbours.

In early intervals, the risk ratio exceeds 8 and in some cases surpasses 15, indicating a strong association between exposure and subsequent behavioural change. As overall prevalence increases, the risk ratio declines but remains above 1 for most transitions, consistent with saturation effects in diffusion-like processes.

4.5 Summary of Empirical Findings

The empirical analysis reveals three consistent patterns:

- **Temporal persistence:** Untruthful behaviour evolves in waves rather than independently across users.

- **Localized clustering:** Untruthful users form cohesive structural regions without complete polarization.
- **Exposure dependence:** Following untruthful users is strongly associated with increased transition risk, particularly during early growth phases.

Together, these findings indicate that user-level truthfulness exhibits both temporal dependence and network-structured correlation. These observations motivate predictive models that jointly incorporate sequential history and structural neighbourhood information, to support this graph based approach it is important to enhance it with post based as well as cluster based features.

5 Features

In this section, we describe the feature representations used for modelling. We distinguish between *post-level features* and *behavioural features*. Post-level features are extracted directly from individual posts. In contrast, behavioural features operate at the user level and summarize longitudinal activity patterns. These behavioural dimensions are derived through clustering procedures that group users into structurally similar profiles.

5.1 Post-Level Features

In this section, we describe the post-level features used in our model. These features are extracted directly from individual posts and capture lexical, semantic, and structural characteristics. We leverage the predictions of the statement classifier together with the output distribution of the baseline false-statement model. Instead of relying on hard class labels, we incorporate the full softmax probability distribution over all classes as features, thereby preserving uncertainty information.

5.1.1 Post-Level Emotion Classification

To generate an additional post-level feature, we apply a pretrained emotion classifier Hartmann [2022] to each post. The model assigns one of seven discrete emotion labels: neutral, anger, fear, joy, surprise, sadness, and disgust.

As shown in the left panel of Figure 6, the corpus is heavily skewed toward neutral expressions ($\sim 37\%$). The right panel presents the conditional false-statement rate $P(\text{FALSE} \mid \text{Emotion})$. Emotionally charged categories exhibit

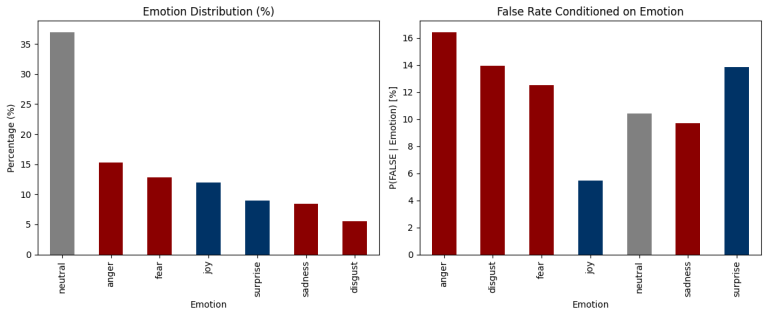


Figure 6: Distribution of Emotions Among Posts Classified as Statements

substantially higher misinformation rates compared to neutral content. Anger shows the highest false rate ($\sim 16.5\%$), followed by surprise ($\sim 14\%$) and disgust ($\sim 14\%$). In contrast, joy and neutral posts display considerably lower rates ($\sim 5.5\%$ and $\sim 10.5\%$, respectively).

These findings indicate a systematic association between affective intensity and factual inaccuracy. Consequently, we include both the predicted emotion label and the corresponding softmax confidence scores across all seven emotion classes as features for downstream false-statement classification.

5.1.2 Text-to-Vector Transformation

Raw post text is converted into a fixed-dimensional numerical representation using a pre-trained transformer model.

For each post, the text is tokenized and fed into the transformer encoder. We extract the contextualized embedding of the `[CLS]` token, which serves as a holistic representation of the entire post.

To reduce dimensionality and adapt the representation to the downstream classification task, the transformer output is passed through a feed-forward projection network. This projection layer produces a compact, fixed-size text vector that is used as input to the subsequent model components.

The transformer parameters remain frozen during training. This stabilizes the semantic representations and mitigates overfitting, while the learnable projection layer enables task-specific adaptation of the embedding space.

5.2 Behavioral Features

Behavioral modeling was decomposed into three dimensions: writing style, temporal rhythm, and activity intensity.

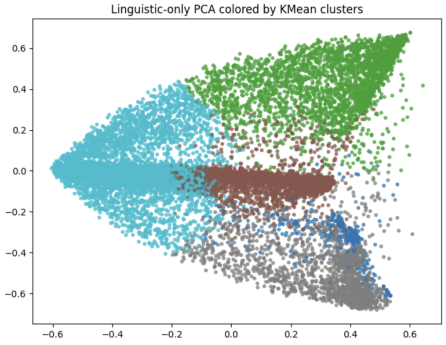
5.2.1 Stylometric Clustering

To capture writing style independent of topic, we used character-level TF-IDF representations (3–5 grams), preserving casing to retain stylistic signals such as punctuation, capitalization, and emoji usage.

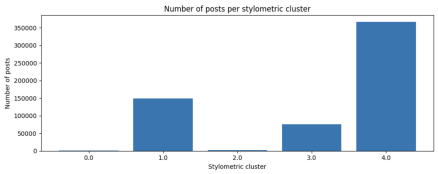
The sparse TF-IDF matrix was reduced using Truncated SVD (50 components) and L2-normalized to operate in cosine space. Silhouette evaluation over $k \in \{4, \ldots, 9\}$ selected $k = 5$ (0.182) as the optimal number of clusters. K-means was applied to obtain the final `style_cluster`.

Visualization and Cluster Distribution The stylometric embedding was projected into two dimensions using PCA (Figure 7a), revealing partially separable stylistic regions with expected overlap in high-dimensional linguistic data.

The cluster distribution (Figure 7b) shows one dominant writing style, two moderately represented groups, and two smaller clusters capturing rare or outlier stylistic patterns. Overall, while stylistic variation exists, a prevailing communication norm characterizes the majority of users.



(a) PCA projection of stylometric embeddings colored by K-means clusters ($k = 5$).



(b) Distribution of posts across stylometric clusters ($k = 5$).

Figure 7: Stylometric clustering analysis results.

5.3 Temporal Posting Patterns

We evaluated temporal posting behavior using monthly and day-of-week distribution features. Silhouette analysis suggested $k = 2$ as the optimal clustering configuration (0.264). However, cluster distribution revealed that

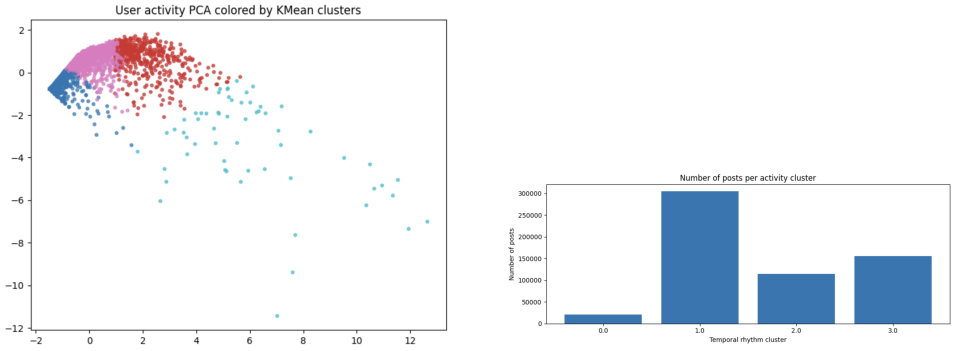
the vast majority of users fall into a single dominant temporal regime, with only a small subgroup exhibiting distinct temporal behavior. This indicates that posting rhythm is largely homogeneous across the platform, with limited evidence of strong temporal segmentation. Consequently, temporal features were treated descriptively rather than used as a primary clustering dimension in downstream analysis.

5.3.1 Activity-Based Clustering

User engagement intensity was captured using:

- Total posts
- Active days
- Lifespan (days)
- Posts per active day

After standardization, silhouette analysis showed strong separability (0.519 at $k = 2$). We selected $k = 4$ (0.508) to model engagement tiers while maintaining high cluster quality.



(a) PCA projection of activity features ($k = 4$).

(b) Distribution of users across activity clusters.

Figure 8: Activity clustering analysis results.

Clear separation indicates strong stratification of users into low, moderate, high, and highly intensive engagement tiers.

5.4 Topic Modeling and Labeling

To identify thematic structure in Truth Social posts, we implemented a semantic topic modeling pipeline combining sentence embeddings, matrix factorization, and LLM-based labeling.

Feature Representation. Each post was encoded using the Sentence-Transformers model `all-MiniLM-L6-v2`, producing 384-dimensional sentence embeddings. The embeddings were L2-normalized to operate in cosine space, allowing semantically similar posts to be close in vector space. This approach captures contextual meaning beyond simple keyword frequency.

Topic Extraction. We applied Non-negative Matrix Factorization (NMF) to extract latent topics from the embedding matrix. Since NMF requires non-negative inputs, negative embedding values were clamped to zero before factorization. We initially experimented with $K = 30$ topics; however, qualitative inspection revealed overlapping and semantically redundant themes. Reducing the model to $K = 20$ produced more coherent and distinct topic groupings. The resulting post-topic matrix was normalized so each post forms a probabilistic topic distribution.

Representative Posts and Labeling. To interpret each topic, we selected the top 15 posts with the highest topic weight as representative examples. These exemplars were provided to a locally hosted LLM (LLaMA3 via Ollama), which was prompted to generate a concise 3–6 word topic label without explanation or punctuation. This ensured consistent and human-readable labels across topics.

Dominant Topic Assignment. Each post was assigned a dominant topic via $\arg\max$ over its topic mixture. Figure 9 shows the distribution of posts across topics. To examine user-level engagement, we computed the average topic weight per author and considered a topic present if its mean weight exceeded 0.05. The number of users associated with each topic is shown in Figure 10.

Observations. Both the post-level and user-level distributions exhibit a highly similar and strongly right-skewed structure. The same set of dominant political topics account for the majority of both posts and participating users. The close alignment between the two distributions suggests that high-volume

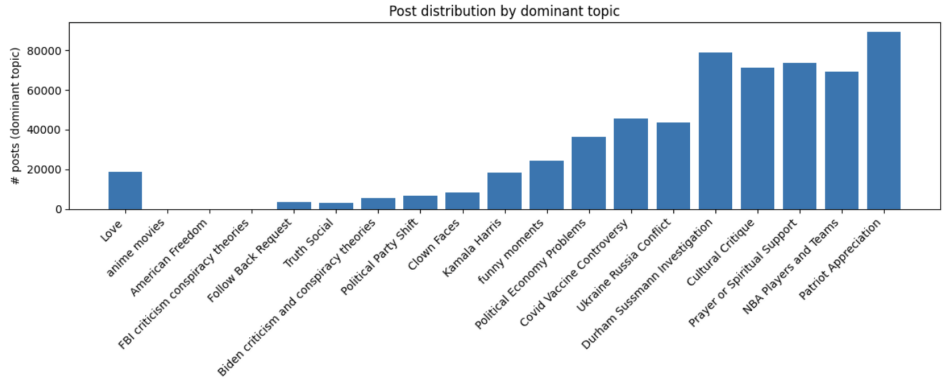


Figure 9: Distribution of posts by dominant topic.

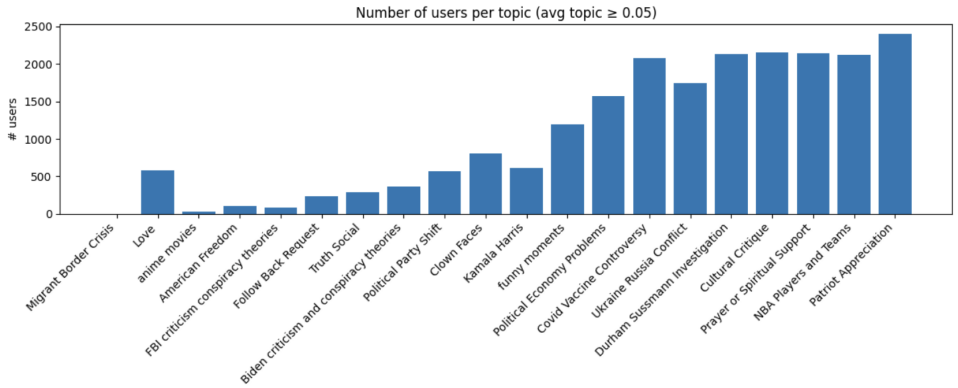


Figure 10: Number of users associated with each topic (average weight ≥ 0.05).

topics are not driven by a small number of highly active users, but rather reflect broad engagement across the user base. Conversely, niche topics remain marginal both in post volume and in user participation. Overall, the results indicate that thematic dominance is consistent at both the content and user levels.

5.5 Feature Representation

The preceding subsections introduced the different post-level and behavioral feature types. We now describe how these components are transformed into numerical representations and combined for integration into the Temporal

Graph Neural Network.

5.5.1 Numerical Encoding of Structured Features

All non-textual features are converted into numerical form prior to model integration. Categorical variables are represented using one-hot encoding, while continuous variables are standardized via z-score normalization. Count-based engagement features are log-transformed before normalization to reduce skewness. Probabilistic outputs, such as truth or sentiment scores, are retained as continuous values, and multi-cluster assignments are encoded as normalized frequency vectors.

5.5.2 Feature Concatenation

After conversion, all components are concatenated into a single high-dimensional feature vector:

$$x = [x_{\text{text}}, x_{\text{label}}, x_{\text{categorical}}, x_{\text{sentiment}}, x_{\text{engagement}}, x_{\text{cluster}}].$$

This unified vector preserves semantic content, probabilistic predictions, behavioral attributes, and interaction statistics within a single representation space.

The resulting post-level vectors serve as input to the post encoder of the Temporal Graph Neural Network described in the following section.

6 Temporal Graph Neural Network Architecture

To capture the joint temporal and structural dynamics of user-level truthfulness, we propose a Temporal Graph Neural Network (TGNN) that integrates post-level encoding, temporal state evolution, and graph-based message passing within a unified framework.

6.1 Temporal User Representation

Within each time interval, a user may produce multiple posts. These post feature vectors are aggregated using a GRU-based encoder to produce a fixed-size representation summarizing the user's activity during that interval. If no posts are present, a zero vector is used.

Each user maintains a hidden state that evolves sequentially across time intervals. The state update is implemented using a GRUCell that integrates the current interval's post embedding, the previous hidden state,

and optionally the prior truthfulness signal. This recurrent formulation captures behavioral persistence and temporal dependencies in user activity.

6.2 Graph-Based Propagation

After the temporal update, user representations are propagated through the follower network. For each user, hidden states of followed accounts are aggregated via mean pooling and combined with the user's own state through learnable linear transformations followed by a non-linear activation.

This step enables structural exposure effects to influence user representations while preserving individual temporal dynamics.

6.3 Prediction and Modeling Perspective

At each time interval, the updated user representation is mapped to a scalar logit and transformed via a sigmoid function to obtain a probability estimate of untruthfulness.

Overall, the proposed TGNN jointly integrates:

- Semantic modeling (text features),
- Temporal modeling (recurrent state evolution),
- Structural modeling (graph-based message passing).

By combining these components, the model captures non-linear behavioral evolution driven by both individual activity and network exposure.

7 Training and Experimental Setup

7.1 Optimization Procedure

The model is trained using the binary cross-entropy loss with logits. This formulation allows direct optimization of probabilistic predictions without requiring an explicit sigmoid layer during training.

Parameters are optimized using the Adam optimizer with weight decay. Gradient clipping is applied to stabilize training and prevent exploding gradients in the recurrent components.

Training is performed for a maximum of E epochs with early stopping based on validation macro-F1. If validation performance does not improve for P consecutive epochs, training is terminated and the best-performing model is retained.

7.2 Temporal Data Split

To preserve chronological consistency, we employ a temporal split. Let T denote the number of time intervals.

- The first 80% of intervals are used for training.
- The remaining 20% are used for validation.

This setup ensures that future information is not used to predict past behavior and reflects a realistic forecasting scenario.

7.3 Decision Threshold Selection

The model outputs a probability estimate of user-level untruthfulness. To obtain binary predictions, a classification threshold τ is applied:

$$\hat{y} = \begin{cases} 1 & \text{if } p \geq \tau \\ 0 & \text{otherwise.} \end{cases}$$

We evaluate multiple thresholds and select the one that maximizes validation macro-F1.

7.4 Evaluation Metrics

Model performance is evaluated using class-wise precision, recall, and F1-score, as well as macro-F1 and overall accuracy. Macro-F1 serves as the primary evaluation metric, as it accounts for class imbalance by weighting both truthful and untruthful classes equally.

8 Results

We first report the performance of the temporal graph neural network without incorporating the previous truthfulness state of a user (`use_past_y=False`). This ensures that predictions are based exclusively on textual content and structural information from the network, avoiding autoregressive dependence on prior labels.

The best-performing configuration under this constraint (`threshold=0.35`, `use_graph=True`, `use_posts=True`) achieved a Macro-F1 score of 0.855 and an overall accuracy of 0.967. The F1 score for the untruthful class was 0.728, while the truthful class achieved an F1 score of 0.982. Precision for untruthful users reached 0.75 with a recall of approximately 0.71, indicating

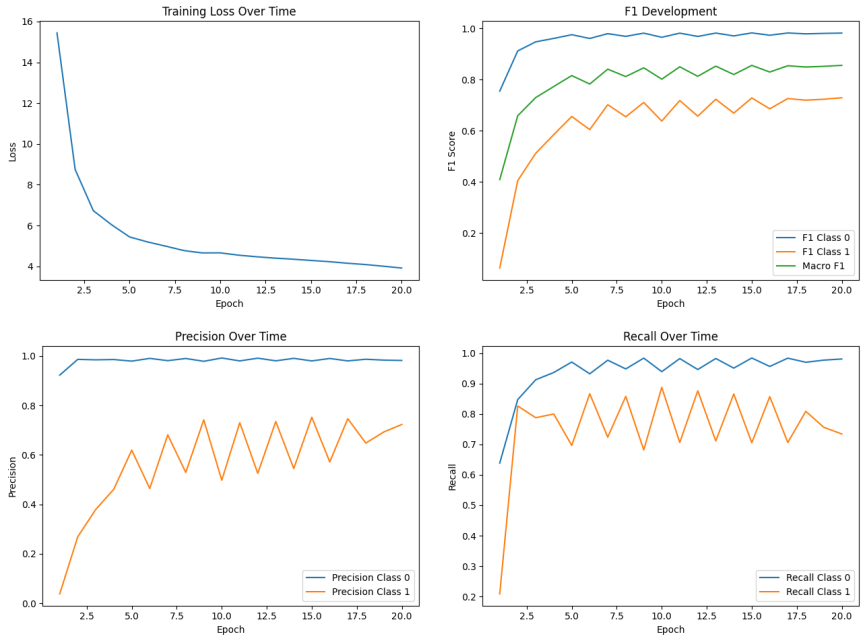


Figure 11: Training dynamics with the following paramters: `threshold=0.35`, `use_graph=True`, `use_posts=True`.

that the model maintains a balanced trade-off between false positives and false negatives. At the same time, the extremely high precision and recall for the truthful class demonstrate that the model does not collapse into trivial majority class prediction despite the underlying imbalance.

When comparing the graph-augmented model to the version without explicit graph features, we observe only a marginal improvement in macro F1 score (0.851). This limited gain may be attributed to the post-based features already encoding substantial relational information, thereby reducing the additional signal provided by explicit graph structure. Furthermore, the use of mean pooling for neighborhood aggregation may have contributed to the modest performance increase. Mean aggregation assigns equal weight to all neighbors and can dilute highly informative signals, potentially leading to over-smoothing and reduced discriminative power of node representations.

When incorporating the previous truthfulness state of a user (`use_past_y=True`), the model achieves a Macro-F1 score of 0.910 and an overall accuracy of 0.976, substantially outperforming the non-autoregressive configuration. The improvement highlights the strong temporal persistence of user-level truth-

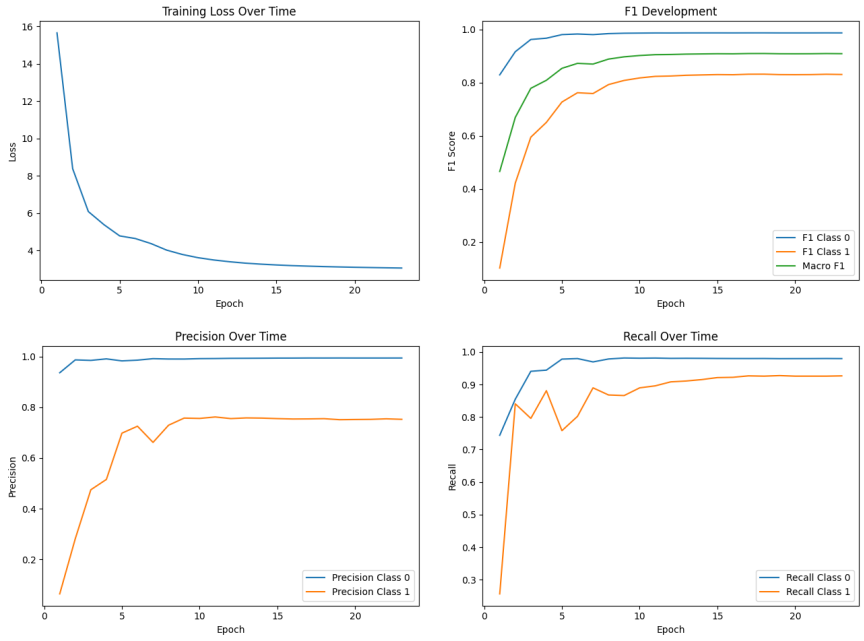


Figure 12: Training dynamics of the temporal graph neural network without temporal self-dependence.

fulness behavior.

However, incorporating previous truthfulness values requires access to ground-truth labels from earlier time steps. If these labels are already available and reliable, the prediction problem becomes largely autoregressive. In such a scenario,one could directly calculate the labels instead of relying on the structural modeling capacity of the graph neural network. In the extreme case where truthfulness is continuously labeled over time, the need for graph-based inference diminishes substantially, as the historical labels themselves carry strong predictive power.

Nevertheless, the high recall of 0.926 for the untruthful class suggests a practically meaningful compromise. Instead of exhaustively labeling all users at every time step, one could apply selective labeling: initial ground-truth annotations could be used to bootstrap the model, which then identifies high-risk users for further verification. Such a semi-supervised strategy preserves the value of structural and content-based modeling while limiting manual annotation effort.

9 Conclusion and Outlook

In this work, we investigated temporal truthfulness dynamics within a large-scale Truth Social follower network and proposed a Temporal Graph Neural Network (TGNN) to model user-level behavioral evolution. Our empirical analysis revealed three consistent structural patterns. First, untruthful behavior evolves in temporally correlated waves rather than through isolated individual shifts. Second, exposure to untruthful neighbors is associated with elevated transition risk, particularly during early growth phases, suggesting that network context plays a substantial role in behavioral evolution.

From a predictive perspective, feature-based user flagging achieved high performance even without incorporating prior truthfulness states. The strong Macro-F1 score demonstrates that semantic content, behavioral attributes, and structural information jointly provide substantial predictive power. Interestingly, explicit graph message passing only marginally improved performance compared to feature-only modeling. This limited gain may indicate that important relational signals are already encoded within post-level and engagement features. Additionally, the use of mean aggregation in the GraphSAGE framework may have diluted highly informative neighbor signals, potentially reducing the discriminative capacity of structural propagation.

Incorporating prior truthfulness states significantly increased predictive performance, highlighting strong temporal persistence. This suggests that users who are flagged as untruthful are substantially more likely to be flagged again in subsequent intervals. While this autoregressive component improves accuracy, it also raises conceptual questions. If historical labels are continuously available, prediction becomes largely persistence-driven, reducing the marginal contribution of structural modeling. In realistic moderation settings, however, exhaustive labeling is infeasible. A selective labeling strategy—where the model identifies high-risk users for targeted review—may offer a scalable compromise between predictive performance and annotation effort.

Several limitations must be acknowledged. First, the follower network represents an induced subgraph of publicly accessible accounts and is treated as static over the observation period. Structural changes in the network could not be captured. Second, the observed wave-like dynamics may partially reflect exogenous political events rather than purely endogenous diffusion processes. Third, the binary truthfulness labels used in this paper were generated by a large language model rather than human annotators. While automated labeling enables large-scale and internally consistent annotation,

it may introduce systematic biases or misclassifications compared to expert human judgment. Consequently, the reported performance should be interpreted relative to the consistency of the automated labeling scheme rather than as an absolute measure of real-world truthfulness detection accuracy. Incorporating human-validated annotations or hybrid human–AI labeling approaches would strengthen future evaluations.

Future research can extend this work in several directions. More expressive graph neural network architectures—such as attention-based aggregation or influence-weighted message passing—may better capture asymmetric exposure effects and prevent information dilution. Dynamic graph formulations could model evolving follower relationships more realistically.

It would also be valuable to examine how the proposed approach performs on less politically polarized platforms. Truth Social represents a highly ideologically concentrated environment, and it remains unclear whether similar temporal persistence and structural clustering patterns emerge in more heterogeneous or less politically heated networks.

Finally, the strong performance gain from incorporating past truthfulness labels warrants deeper investigation. Understanding whether this persistence is driven by stable individual traits, reinforcement through network feedback loops, or platform-level dynamics would provide important insight into the mechanisms underlying behavioral escalation.

Overall, this paper demonstrates that semantic, behavioral, temporal, and structural signals jointly shape the evolution of untruthful behavior in online networks. By integrating these components into a unified modeling framework, we provide both descriptive insight and a foundation for scalable, network-aware moderation strategies in polarized digital environments.

References

Patrick Gerard, Nicholas Botzer, and Tim Weninger. Truth social dataset, 2023.

Jochen Hartmann. Emotion english distilroberta-base. https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/, 2022.