Forecasting "Neg Storms": Time-Aware Modeling of Toxic Situations in Social Media

Irien Akter
University at Albany, SUNY
Albany, NY, USA
Email: iakter@albany.edu

Vivek K. Singh
Rutgers University
New Brunswick, NJ, USA
Email: v.singh@rutgers.edu

Pradeep K. Atrey
University at Albany, SUNY
Albany, NY, USA
Email: patrey@albany.edu

Abstract*—Social media platforms face escalating harm from concentrated waves of toxic interactions, which we term *Neg Storms*. Unlike isolated abusive remarks, these storms emerge through rapid, correlated actions that amplify negativity and create severe risks for targets and communities. Existing moderation approaches largely focus on piecemeal detection of individual comments, missing the situational dynamics that drive escalation. This paper introduces a proactive framework for forecasting *neg storms* using early conversational signals. We formalize *Comment Storm Severity (CSS)* as a time-aware metric of thread-level toxicity, propose models that predict CSS from only the first k comments, and evaluate feature sets combining timing and content cues. Experiments on Reddit and Instagram show that timing features alone outperform content-only features, and that combining both yields the best performance (ROC-AUC $\approx$ 79.7%; $R^2 \approx 0.24$). While predictive scores are modest, these results validate the feasibility of anticipating harmful situations before they fully unfold. We discuss practical implications for platforms, including early checkpoints to prioritize high-risk threads, apply reversible friction, and route uncertain cases for human review. This work establishes an important starting point for research on situation-level modeling of toxicity and proactive moderation in online communities. *Note: This paper deals with a sensitive topic and includes examples of negative online comments.

***Index Terms*—Early Prediction, Comment Storm Severity (CSS), Toxicity Forecasting, Social Media (Reddit, Instagram), Neg Storm.**

I. Introduction

In today's digital age, social media platforms such as Reddit and Instagram connect people, surface new ideas, and sustain vibrant communities [1]. Yet these same spaces are vulnerable to concentrated waves of harm that unfold at the situation level, not the sentence level. We refer to these episodes as *Neg Storms*: periods where rapid, mutually reinforcing responses produce a surge of toxicity that is qualitatively different from isolated abusive remarks [2], [3]. Prior work on coordinated harassment and "pile-ons" shows that collective actions like dogpiling and brigading can be organized or emergent, and their impact is felt through volume, synchronicity, and visibility rather than any single comment [4]. Related trust-and-safety analyses describe brigading as coordinated mass engagement that overwhelms targets and distorts discourse, again highlighting the collective mechanism of harm rather than individual posts [5].

Despite this, most automated moderation still treats toxicity as a piecewise classification problem, labeling comments one at a time [6]. This design choice misses two critical realities. First, toxicity is deeply contextual and socially situated; per-comment models trained on decontextualized text can encode biases and over-flag language from marginalized dialect communities, which undermines both fairness and trust [7], [8]. Second, escalation is a process. Harm often arises from the temporal evolution of a thread, where sequences of replies accumulate into a burst. Large-scale studies show that longer online discussions become more toxic in systematic ways across platforms and over time, indicating that interaction dynamics, and not just content, are key to prediction [9]. In diffusion research, hateful or abusive content spreads faster, further, and wider than benign content, again pointing to collective and temporal mechanisms that per-comment classifiers do not capture [10].

To move from "looking at trees" to "seeing forests," we argue for modeling situations, not isolated events [11]. Thread-level structure and timing carry predictive signals. Early work that incorporates broader conversational context already shows sizable gains in forecasting hateful discussions relative to limited-context baselines, and suggests that community norms shape these dynamics in platform-specific ways [12], [13]. Moreover, self-exciting point-process models such as Hawkes processes provide a principled foundation for representing burstiness, contagion, and escalation in social systems, underscoring the value of time-aware indicators for early warning [14].

This paper operationalizes that shift. We present an early-warning framework for forecasting *neg storms* using only the first few comments in a thread. The core element of our approach is the *Comment Storm Severity (CSS)*, a compact, interpretable [15], time-aware metric that quantifies how intensely toxicity concentrates within a short span, normalized against early baseline behavior. We formally define a *Neg Storm* as a thread segment in which the CSS exceeds a predefined threshold, indicating a concentrated episode of harmful interaction.

We evaluate our framework on Reddit and Instagram, two platforms with contrasting interaction patterns, ranging from threaded discussions to rapid, reactive commenting, to assess generalization across community styles and examine whether early trajectory signals transfer across ecosystems [9].

Concretely, our contributions are threefold:

1) We formalize CSS as a dynamic, time-aware indicator of escalation and introduce the problem of early prediction

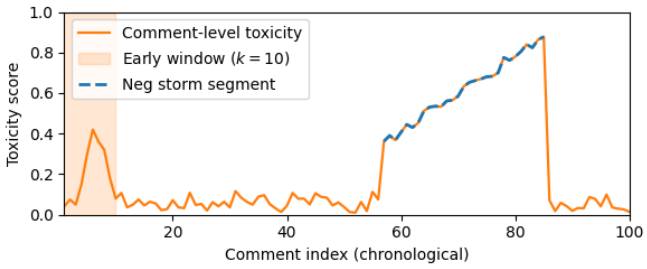

Fig. 1. Schematic example of a "neg storm". The thin solid curve shows comment-level toxicity scores over the thread (higher is more toxic). The hatched region on the left marks the early window (first k=10 comments) used for forecasting. The thicker dashed segment highlights the *neg storm* region: a later block of comments with persistently elevated toxicity that yields a high Comment Storm Severity (CSS) value.

of *Neg Storms*.

2) We propose early-prediction models that rely only on the first k comments to estimate future CSS, enabling proactive intervention.
3) We introduce a joint text–time feature set that captures lexical toxicity, reply tempo, inter-arrival burstiness, and early concentration patterns to improve early CSS forecasting.

Figure 1 illustrates the CSS and *Neg Storm* constructs on a hypothetical conversation: toxicity is modest in the early window (first k comments) and later concentrates into a sharp "neg storm" segment, which yields a high CSS score.

By forecasting severity trajectories from the earliest comments, this work advances a proactive moderation paradigm, where moderators and automated systems can anticipate harmful escalation before it fully unfolds, rather than react after damage has accumulated [8], [12].

The rest of the paper is organized as follows. Section II reviews related work and positions our study in context. Section III describes the dataset used in this study, while Section IV presents the proposed methodology. Implementation details are provided in Section V, followed by results and analysis in Section VI. Finally, Section VII concludes the paper.

II. Related Work

A. Event-level Toxicity Detection and Its Limits

Much of the existing literature treats toxicity as a property of individual comments, using supervised classifiers trained on crowd-labeled corpora or platform-scale datasets. These approaches have enabled scalable detection but remain primarily reactive and sentence-focused [16]–[18]. Scholars have highlighted fairness concerns when models ignore conversational context, such as over-flagging language from marginalized communities [7], [8]. Beyond model bias, platform-scale analyses show that harmful behavior is shaped by interaction dynamics [19]. Longer discussions tend to become more toxic, suggesting that situation-level modeling is necessary to anticipate escalation rather than only detect isolated events [9], [10].

B. Thread-level Modeling

Recent work by Ranjith et al. [20] explores thread-level toxicity prediction by estimating the average toxicity of a discussion. While this is a step toward contextual modeling, our approach goes further by capturing the temporal and structural aggregation of toxicity into a concentrated episode, which we term a *neg storm*. We evaluate this using a CSS score that reflects how intensely toxicity clusters within a short span. The difference is analogous to predicting average rainfall versus forecasting a storm. Average toxicity may remain low even when a thread contains a brief but severe toxic burst. Our goal is to detect and forecast these bursts early, enabling proactive moderation before escalation occurs.

C. Individual-targeted Constructs vs Thread-centric Neg Storms

Several situation-level constructs in online harm focus on targeted abuse. Dog-piling refers to groups converging to attack an individual, often with explicit coordination [21], [22]. Brigading involves cross-community mobilization to flood or manipulate a thread, frequently targeting a person or group [23]. Cyberbullying is defined as intentional, repeated aggression against a victim who cannot easily defend themselves [24]–[27]. These constructs are valuable for victim-centric protections, but our formulation of a neg storm is thread-centric. It is evaluated by CSS and does not require identifying a specific target. This distinction is important for content moderators, who often triage threads rather than individuals and need tools that can flag harmful situations even when no single victim is evident [28], [29].

D. Context-aware Modeling

Hebert et al. [12] propose graph transformer models to predict hateful discussions on Reddit by incorporating conversational structure and community context. Their work focuses on detecting whether a thread will become hateful, using community-specific features and full-thread context. In contrast, we forecast a neg storm using only the first k comments and a continuous CSS metric. Our approach emphasizes early intervention and generalization across platforms with distinct interaction styles, such as Reddit and Instagram. While Hebert et al. advance context-aware detection within a single platform, our work introduces time-aware forecasting of thread-level escalation that is platform-agnostic and operationally aligned with moderation workflows.

E. Temporal Modeling and Operational Relevance

Studies on burst modeling and diffusion dynamics support the need for time-aware indicators. Self-exciting point processes such as Hawkes models have been used to represent burstiness and mutual excitation in social media [14]. Toxic content tends to spread faster and deeper than benign content, and toxicity often increases with thread depth [9], [10]. Our CSS metric captures these dynamics by aggregating early timing and content signals to quantify escalation. This enables practical interventions such as prioritizing high-risk threads, applying

reversible friction like slow-mode or warning nudges, and routing uncertain cases for human review. These actions are compatible with existing trust-and-safety pipelines and do not depend on identifying a specific target, making them well-suited for thread-level moderation.

Position of this work: In summary, prior research provides powerful detection models, insights into temporal dynamics and participant roles, and practical pipelines for media-centric platforms. What remains missing, however, is a simple and deployable approach to forecast the eventual, thread-level severity of a conversation based solely on its earliest remarks—while also characterizing whether escalation is likely to burst, build gradually, or be mitigated—and to achieve this across platforms with explicit consideration of timeliness. This is the direction our work takes.

III. Dataset

A suitable dataset is an essential requirement for task validation. We evaluate our approach on two widely used social media platforms that exhibit distinct conversation dynamics: Instagram (media–comment threads) and Reddit (post–comment threads). Using both allows us to study early escalation cues across different interaction platforms while keeping a consistent forecasting setup.

A. Instagram Dataset

We use a dataset shared by the authors of Hosseinmardi et al. [30]. The data were collected between August 2011 and June 2014 and link media posts to their complete comment streams. It includes records for 14,063 Instagram users, about 1.74M media sessions, and 6.82M comments. Each session is a time-ordered thread under a post. We retain post/thread IDs, timestamps, and model-based toxicity scores for timing-aware features.

B. Reddit Dataset

We use a thread-organized Reddit dataset released alongside prior work on forecasting conversation toxicity by Ranjith et al. [20]. Each instance is a conversation thread composed of a post and its comments in chronological order. Comments are annotated with continuous toxicity scores in $[0, 1]$ using the Perspective API, yielding a representation that aligns naturally with our thread-level escalation target, i.e., CSS.

Across the dataset, Reddit threads contain on average 22.1 comments and Instagram threads 71.3 comments. Therefore, the early window of k=10 comments used for the prediction models uses only a small portion of the conversation, i.e., roughly half and one third of a conversation for Reddit and Instagram threads, respectively.

IV. Methodology

Our goal is to *forecast* whether a conversation will escalate into a high-severity neg storm using only the earliest conversation patterns in that conversation. We cast this as early thread-level prediction: given the first k comments in a thread, we compute timing- and content-aware features and map them to a single severity target, which can be a scalar *Comment Storm Severity (CSS)* or its binarized form above a threshold. This section details the problem setup, notation, target construction, feature design, models, and evaluation.

A. Problem Setup

Consider a thread with comments $\{(t_i, x_i)\}_{i=1}^{n}$, where t_i is the wall-clock timestamp *in seconds* and x_i the text of the i-th comment. We consume only the first k=10 comments (sorted by time, ties broken by ID) to compute features $\phi(\cdot)$, and we predict a scalar thread-level target $s \in \mathbb{R}_{\geq 0}$ that summarizes future escalation dynamics. The value of $k = 10$ was chosen based on preliminary parameter analysis. Threads with fewer than k comments are excluded to maintain a consistent early window.

B. Notations and Symbols

We summarize the main symbols used in the methodology; each symbol is also defined again near its first appearance in an equation.

- n: total number of comments in a thread.
- k: number of early comments used for forecasting (operationalized at k=10 in current experiments).
- t_i: wall-clock timestamp (in seconds) of the i-th comment.
- x_i: text of the i-th comment.
- $\hat{p}_i \in [0, 1]$: toxicity probability of comment i from the RoBERTa-based classifier.
- $y_i \in \{0, 1\}$: binary toxicity label of comment i (1 = toxic, 0 = non-toxic) obtained by thresholding $\hat{p}_i$.
- θ_{tox}: fixed probability threshold used to convert $\hat{p}_i$ into y_i (we use $\theta_{\text{tox}} = 0.5$).
- $\phi(\cdot)$: feature mapping from the first k comments in a thread to a fixed-dimensional feature vector.
- s: scalar thread-level target; in our case $s = \text{CSS}$.
- λ_0: early baseline toxic rate computed over the first k comments.
- $\lambda(\tau, \Delta)$: toxic rate within a candidate future window that starts at time τ and has duration Δ.
- τ: start time (in seconds) of a candidate future window, with $\tau \geq t_k$.
- Δ: duration (in seconds) of a candidate future window.
- Δ_{ref}: reference duration (in seconds) used in the compactness weight; we set Δ_{ref} to a one-day reference timescale.
- $\alpha > 0$: compactness exponent controlling how strongly we favor short, sharp bursts over long, diffused ones.
- $\varepsilon > 0$: small constant added to denominators to avoid division by zero.

C. Target Construction: Comment Storm Severity (CSS)

CSS measures how intensely toxicity concentrates after the early window, *relative to* the early baseline. Let the early baseline toxic rate be

$$\lambda_0 \;=\; \frac{\sum_{i=1}^{k} y_i}{t_k - t_1 + \varepsilon},$$

where y_i is the binary toxicity label of comment i, t_1 and t_k are the timestamps of the first and k-th comments, and $\varepsilon > 0$ is a constant, as defined earlier.

For any future window $[\tau, \tau + \Delta]$ with $\tau \geq t_k$ and $\Delta > 0$, we define the toxic rate in that window as

$$\lambda(\tau, \Delta) \;=\; \frac{\sum_{i:\, t_i \in [\tau, \tau+\Delta]} y_i}{\Delta}.$$

Here τ is the start time of the candidate window, $\tau + \Delta$ is its end time, and $\lambda(\tau, \Delta)$ measures the average number of toxic comments per second within that window.

Excess intensity is captured by $\max\{0,\, \lambda(\tau, \Delta) - \lambda_0\}$, and a compactness weight favors sharp bursts over long, diffuse activity:

$$\left(\frac{\Delta_{\text{ref}}}{\Delta}\right)^{\alpha},$$

where Δ_{ref} is a fixed reference duration (we use a one-day timescale) and $\alpha > 0$ controls how much the system designers reward temporally compact bursts.

The CSS score for a thread is then

$$\text{CSS} \;=\; \max_{\tau, \Delta} \left[\; \max\{0,\, \lambda(\tau, \Delta) - \lambda_0\} \cdot \left(\frac{\Delta_{\text{ref}}}{\Delta}\right)^{\alpha} \;\right].$$

This construction jointly captures *excess* toxicity (above the early baseline) and *compactness* in time, normalizes across threads via λ_0, and yields a single nonnegative scalar suitable for regression or thresholded classification.

Practical sweep: In practice, we discretize the window size Δ over a small grid (e.g., $\{5\text{m}, 15\text{m}, 1\text{h}, 6\text{h}, 24\text{h}\}$) and, for each candidate Δ, slide the window anchor τ over comment timepoints after t_k. We set Δ_{ref} to the longest window in the grid (24 hours) so that the compactness term compares each candidate window to a fixed one-day reference timescale, and we tune the compactness exponent $\alpha \in [0.5, 1.0]$ on the validation split. To stabilize heavy tails, CSS is clipped at the 99.5^{th} percentile *per platform*, computed on the training split and then applied to validation and test. For RoBERTa labeling we retain the fixed toxicity threshold $\theta_{\text{tox}} = 0.5$ used to convert model probabilities $\hat{p}_i$ into binary labels y_i.

V. Implementation

We implement the approach as a simple, reproducible pipeline shared by both regression and classification views: (i) ingest raw conversation data; (ii) label each comment with a unified toxicity model (data pre-processing); (iii) compute the thread-level ground-truth severity, i.e., CSS; (iv) extract early-window features from only the first k=10 comments; (v) train a predictive model with these features as input and CSS as the target (continuous or binarized); and (vi) evaluate on a held-out test set.

A. Pre-processing Data

We harmonize labeling across platforms by re-scoring every comment from Instagram [30] and Reddit [20] with a single RoBERTa-based toxicity classifier. We apply the model to each raw comment using a standard Transformer pipeline (subword tokenization, truncation to a fixed maximum length, and a sigmoid output layer) and obtain a toxicity probability $\hat{p}_i \in [0, 1]$ for comment i.

For each comment we retain: (i) the thread/post identifier and timestamp, (ii) the model's toxicity probability $\hat{p}_i$, and (iii) a binary toxicity label $y_i \in \{0, 1\}$ obtained by thresholding at a fixed probability level $\theta_{\text{tox}} = 0.5$, i.e., $y_i = \mathbb{I}[\hat{p}_i \geq \theta_{\text{tox}}]$, where $\mathbb{I}$ denotes a function that returns 1 if its argument is true and 0 otherwise. This single labeling pipeline ensures that "toxic" has a consistent meaning on both platforms and supports early-window features such as early toxic fraction, streak structure, and inter-arrival statistics. In general, this step can be replicated using any publicly available RoBERTa-based toxicity model trained on a similar data set and applying the same per-comment scoring and thresholding procedure.

a) Thread-level screening.: To reduce very short and extremely long conversations that can distort early-prediction signals, we apply two filters per thread:

1) **Minimum length:** require at least k=10 comments; discard threads with $|\text{comments}| < 10$.
2) **Maximum span:** discard threads whose total timespan (first to last comment) exceeds 180 days (measured from t_1 to t_n, timestamps in seconds).

b) Resulting dataset sizes.: After screening, Instagram retains 1,969 of 2,212 threads ($\approx 89.0\%$; 243 removed), and Reddit retains 199 of 371 threads ($\approx 53.6\%$; 172 removed). All subsequent modeling in this paper focuses on prefixes at k=10.

c) Severity parameterization.: We use $\alpha = 0.6$ and a reference timescale $\Delta_{\text{ref}} = 86{,}400\,\text{s}$ (1 day). Auxiliary windows for time-based features are 60, 300, 900, 3,600, 10,800, 21,600, 43,200, 86,400, 172,800, and 604,800 seconds (i.e., 1/5/15 minutes; 1/3/6/12 hours; 1/2/7 days). This parameterization supports practically meaningful temporal structure while keeping the early prefix fixed at k=10.

B. Ground Truth: CSS Computation

Given the unified labels, we compute thread-level ground truth as CSS: estimate the early baseline toxic rate λ_0 from the first k comments, sweep future windows $[\tau, \tau+\Delta]$ to measure excess intensity above λ_0, weight by compactness $(\Delta_{\text{ref}}/\Delta)^{\alpha}$, and take the maximum score. CSS is a single, timing-aware severity value per thread and serves as the supervised target for learning in both the regression (continuous CSS) and classification (high/low CSS) views.

C. Early-Window Feature Extraction

We compute all predictors strictly from the first k comments of each thread. Let $y_i \in \{0, 1\}$ be the toxic label for comment i, $\hat{p}_i \in [0, 1]$ its toxicity score, and t_i the wall-clock time (minutes). Define gaps $\Delta t_i = t_i - t_{i-1}$ for $i \geq 2$.

a) Content / label–score features (first k).:

- **Early toxic fraction:** $\frac{1}{k}\sum_{i=1}^{k} y_i$ (share of toxic comments).
- **Toxicity intensity stats:** mean and variance of $\{\hat{p}_i\}_{i=1}^{k}$ (how "strong" early toxicity is).

- **Max short-window toxic proportion:** $\max_{w\in\{3,4,5\}} \max_j \frac{1}{w}\sum_{i=j}^{j+w-1} y_i$ (micro-bursts over 3–5 comments).
- **Earliest toxic index:** $\min\{i \mid y_i = 1\}$ (how soon toxicity appears).
- **Streak structure:** number of toxic streaks and the length of the longest toxic streak within $1{:}k$.
- **Alternation rate:** fraction of adjacent pairs that switch label (toxic↔non-toxic).
- **EMA of toxicity:** last values of exponential moving averages $m_i^{(\alpha)} = \alpha\,\hat{p}_i + (1-\alpha)m_{i-1}^{(\alpha)}$, reported at $i = k$ for $\alpha \in \{0.5, 0.8\}$.
- **Monotone tendency (index-wise):** Spearman-like rank correlation between comment index and toxicity score, $\rho_{\text{idx-tox}} = \mathrm{corr}\big(\mathrm{rank}(i),\, \mathrm{rank}(\hat{p}_i)\big)$ for $i = 1{:}k$.
- **Early Burst (EB) rates by minutes:** toxic share within the first 10 minutes and within the first 60 minutes after t_1:

$$\mathrm{EB}_{10} = \frac{1}{|M_{10}|}\sum_{i\in M_{10}} y_i, \qquad \mathrm{EB}_{60} = \frac{1}{|M_{60}|}\sum_{i\in M_{60}} y_i,$$

where $M_x = \{\, i : 0 \le (t_i - t_1) \le x \,\}$.

b) Timing-aware features (from timestamps).:

- **Inter-arrival statistics:** mean, variance, and coefficient of variation (CV) of $\{\Delta t_i\}_{i=2}^{k}$ (tempo).
- **Toxic inter-arrival stats:** the same statistics restricted to indices where $y_i = 1$ (tempo of toxic replies).
- **Burstiness / clumping:** Gini coefficient over $\{\Delta t_i\}_{i=2}^{k}$ (higher implies tighter clumps).
- **Early span:** $t_k - t_1$ (how compressed the first k comments are in time).
- **Growth slope:** growth slope estimated with Ordinary Least Squares (OLS) regression on the cumulative toxic count as a function of time (early acceleration toward toxicity).

D. Train/Test Splits and Protocol

All experiments use the same splitting principles to avoid leakage and keep results comparable. We always partition data *by thread*, so that all comments from a thread stay in the same split, and we split *separately by platform* (Instagram and Reddit) before combining splits across platforms.

For the regression view, we use a fixed **70/30** train/test split at the thread level. For the classification view, we sweep a range of train/test ratios (50/50, 60/40, 70/30, 80/20, 90/10). In each case, threads on each platform are assigned to train and test according to the desired ratio, stratified on the high/low CSS label when applicable; the resulting training portions from Instagram and Reddit are then concatenated, and likewise for the test portions.

Model selection and hyperparameter tuning are performed within the training threads using GroupKFold cross-validation with k=5, grouping by thread ID so that all comments from a thread remain in the same fold. For classification, we reserve a small stratified slice (10–15% of each training fold) solely for probability calibration and decision-threshold selection; the remaining CV data and the test set are never used for calibration.

All preprocessing (imputation, standardization, Term Frequency-Inverse Document Frequency (TF-IDF) vectorization, and any PCA) is fitted *only* on training data and applied unchanged to validation, calibration, and test. Tree-based models use early stopping on validation performance. Random seeds are fixed throughout, and no comment-level mixing occurs across folds or splits.

E. Regression and Classification Objectives

We report both regression and classification results because they serve complementary purposes. For the *regression* view, models are trained to predict the continuous CSS value from the early-window features, and we evaluate performance using MAE, RMSE, and R^2 on the held-out test set.

For the *classification* view, we convert CSS into a binary label (high vs. low severity) using the pooled global median CSS computed on the training split (combining both platforms). Let this median be $m \approx 0.63$. Each thread is labeled

$$y^{\mathrm{cls}} = \mathbb{I}[\mathrm{CSS} \ge m],$$

so that "high CSS" corresponds to the upper half of the training CSS distribution. Once m is estimated on the training data, it is kept fixed and applied unchanged when forming labels for the validation and test sets, keeping the definition of "high CSS" consistent across platforms and avoiding any information leakage from validation or test into label construction.

Each fitted classifier from the shared pipeline is then turned into a calibrated scorer $\hat{p}(x) \in [0, 1]$ using *isotonic* calibration on the held-out calibration slice within each GroupKFold split, where x denotes the feature vector for a thread and $\hat{p}(x)$ is the estimated probability that the thread belongs to the high-CSS class.

VI. Results

Two sample threads from Instagram dataset (one with a *neg storm* and one without) are shown in Fig. 2. The early signals (first ten comments) were useful in forecasting the occurrence of a *neg storm* later in the red colored thread. The early comments show a (small) spike, and rolling toxicity scores reaching around 0.4. It included mixture of comments like "*homie you have no posts and no followers take yo salty as* on somewhere*" (toxicity score: 0.96), "*I looove you all! Seriously love yah _ _ _¬• but just out of curiosity why do you have like no furniture ? You guys have plenty of money so just wanted to know If there was any specific reason for it that's all :)*" (toxicity score: 0.02), and "*Hahaha got them!*" (toxicity score: 0.00). This thread later goes through a *neg storm*, where there are 8 comments over a short period of 60 seconds with significantly toxic content. This included comments like "*It's jut when ppl post pics everyyyyyy five secs it shows in explore AINT noone following a weak as* cheap esc*rt who shares pus*y smellin clothes but YOUR thirsty as* ! So miss me n Hte on what? Havin daddy*" (toxicity score: 0.99), "*To bad I'm*

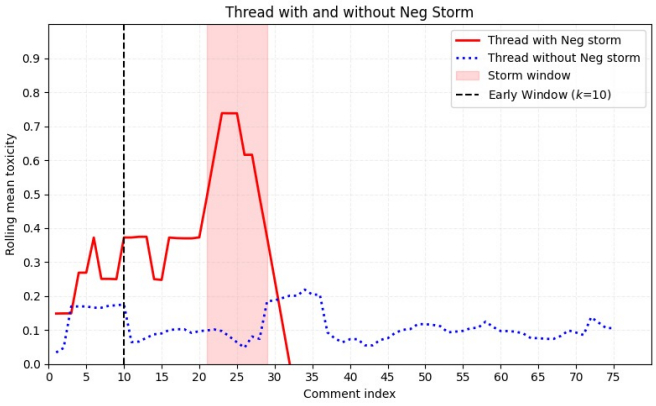

Fig. 2. Comparison between threads with and without neg storm. The red curve indicates a thread exhibiting a neg storm, while the blue (dotted) curve represents a stable thread without major toxicity spikes. The shaded region marks the storm window, and the dashed line denotes the early observation window (k=10).

not haha why is it always u ignorant monkeys replying ? Kik rocks bruh n keep bein thirsty ! Late" (toxicity score: 0.99), and "*a keep inspiring young girls to grow up n pim* there pus*ys for some bla*k ni**a .. Such an inspiration !*" (toxicity score: 0.95). (Note: some characters have been replaced with a *).

The blue (dotted) thread on the other hand has relatively stable toxicity scores in the first ten comments. The average toxicity level is around 0.15 and it includes comments like "*HELLLLLL YEAHHHH!!!!!*" (toxicity score: 0.05). There was no *neg storm* observed later on in this thread and it included comments like "*I'm sat here scrolling through Instagram playing the Kim Kardashian game wishing that was my life*" (toxicity score: 0.00).

Next, we present both a *regression* view, where models predict continuous CSS, and a *classification* view, where CSS is binarized into high vs. low severity via a median split. All results use the same early-window feature set computed from the first k comments.

A. Regression View

We first evaluate *Linear Regression*, *Random Forests (RF)*, and *Gradient-Boosted Trees (GBT)* on the continuous CSS target using the first-k feature set. We report MAE, RMSE, and R^2 on a 70/30 thread split (Table I).

Overall, **GBT** attains the best held-out performance: on Test it achieves MAE 0.315, RMSE 0.642, and R^2=0.236, indicating the most favorable bias–variance trade-off among the three models. **Linear Regression** is competitive (Test MAE 0.357, RMSE 0.660, R^2=0.193), suggesting that a substantial portion of the signal is approximately linear in the engineered features. **RF** is slightly weaker on Test (RMSE 0.667, R^2=0.175) despite being competitive on Train, pointing to mild overfitting to burst/streak patterns. Train→Test gaps in R^2 remain moderate, and the ordering of models is consistent across MAE, RMSE, and R^2, indicating reasonable generalization.

TABLE I
TRAINING (TOP) AND TEST (BOTTOM) RESULTS ACROSS MODELS (AGGREGATED AT k=10).

Split	Model	MAE	RMSE	R^2
Training				
	GBT	0.275	0.483	0.353
	Linear	0.323	0.511	0.276
	RF	0.279	0.525	0.238
Test				
	GBT	0.315	0.642	0.236
	Linear	0.357	0.660	0.193
	RF	0.310	0.667	0.175

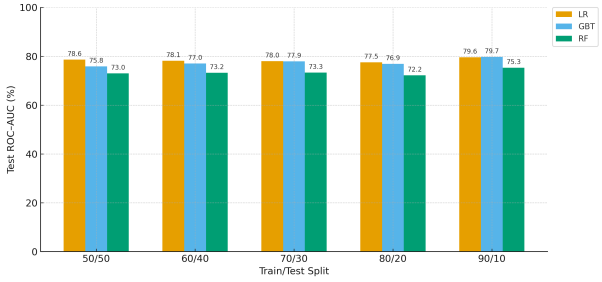

Fig. 3. Test ROC-AUC (%) for each model (LogReg, RF, GBT) across train/test splits (50/50 to 90/10).

B. Classification View: High vs. Low CSS

We next cast early forecasting as a binary classification task by thresholding CSS at the pooled global median on the combined Instagram+Reddit training set. Using the same first-k feature set and three models (Logistic Regression, RF, GBT), we evaluate performance across multiple train/test ratios (50/50, 60/40, 70/30, 80/20, 90/10). Our primary metric is Test ROC-AUC from calibrated probabilities; Accuracy (ACC) serves as a secondary summary. Figure 3 shows Test ROC–AUC for each (split, model) combination, and Table II reports the corresponding numerical values.

The best Test ROC-AUC is 79.7%, obtained by GBT in the 90/10 setting; Logistic Regression performs very similarly in this regime with Test ROC-AUC of 79.6%. Across splits (50/50-90/10), three patterns emerge. First, **Logistic Regression** is a strong baseline at more balanced splits: from 50/50 through 80/20, it attains the highest or nearly-highest Test ROC–AUC. Second, **GBT** benefits more from additional training data and becomes best when training data is most abundant (90/10). Third, **Random Forest** consistently lags behind the other two models in Test ROC-AUC across all splits.

C. Sensitivity to Feature Groups

To understand which early signals matter most, we perform a sensitivity analysis over three feature configurations while keeping the rest of the classification pipeline fixed (same train/test split, GroupKFold protocol, models, and calibration procedure). We train the best performing (GBT) under: (i) a *timing-only*

TABLE II
CLASSIFICATION METRICS (PERCENT) ACROSS ALL TRAIN/TEST SPLITS. AUC = ROC-AUC

Split / Model	Training ACC	CV ACC	Test ACC	Training AUC	CV AUC	Test AUC
50/50 logreg	73.9	71.9	71.1	82.7	80.3	78.6
50/50 gbt	74.8	72.3	70.2	82.5	79.3	75.8
50/50 random_forest	71.4	69.6	67.3	79.5	77.1	73.0
60/40 logreg	73.7	72.3	70.6	82.4	80.4	78.1
60/40 gbt	74.2	72.0	70.0	82.4	78.4	77.0
60/40 random_forest	71.6	68.7	68.1	78.7	76.2	73.2
70/30 logreg	73.7	71.9	69.6	82.3	80.5	78.0
70/30 gbt	75.6	72.7	70.7	83.2	79.7	77.9
70/30 random_forest	71.3	70.0	68.2	78.5	76.6	73.3
80/20 logreg	73.7	72.7	70.3	82.0	80.4	77.5
80/20 gbt	75.4	72.3	68.9	83.0	80.8	76.9
80/20 random_forest	70.7	69.7	67.7	78.3	76.5	72.2
90/10 logreg	73.4	71.9	71.0	81.3	79.7	79.6
90/10 gbt	74.3	72.7	71.9	82.6	80.3	79.7
90/10 random_forest	70.5	68.9	69.6	77.5	75.8	75.3

TABLE III
SENSITIVITY OF TEST PERFORMANCE TO FEATURE GROUPS FOR THE GBT CLASSIFIER (90/10 SPLIT)

Feature group	Model	Test ACC (%)	Test ROC–AUC (%)
Timing-only	GBT	71.0	76.0
Content-only	GBT	63.0	70.0
All features	GBT	71.0	79.7

setting that uses only timestamp-derived features (inter-arrivals, Early Burst (EB) rates, growth slope, span, clumping, tempo, etc.); (ii) a *content-only* setting that uses only label/score and text features (early toxic fraction, streak/alternation features, and TF-IDF aggregates); and (iii) the *full-features* setting used in our main results, which combines both timing and content features.

Table III summarizes the held-out test metrics for the GBT classifier under these three configurations (90/10 split). We report ACC and ROC-AUC as our primary metrics.

These results show two key patterns. First, timing features alone already carry substantial signal: the timing-only configuration outperforms the content-only configuration in both ACC and ROC-AUC, suggesting that *when* toxicity occurs and how it clusters is more informative than early content alone. Second, combining timing and content features yields the best ROC-AUC (79.7%), indicating that the two feature families are complementary and that both are useful for early forecasting of high-CSS *neg storms*.

D. Discussion

Our findings provide initial evidence that early forecasting of *neg storms* is feasible using only the first few comments in a thread. Three insights align with our contributions. First, formalizing CSS as a time-aware metric offers a principled way to quantify escalation beyond isolated toxic remarks. Although predictive performance is modest ($R^2 \approx 0.24$ for regression and ROC-AUC of 79.7% for classification), CSS captures meaningful variance in thread-level harm and validates its role as an early warning signal.

Second, early-prediction models show that initial conversational signals, particularly timing patterns, carry predictive power. Sensitivity analysis reveals that timing-only features outperform content-only features (ROC-AUC 76% versus 70%), indicating that when toxicity occurs and how it clusters is more informative than early lexical cues alone. This supports our hypothesis that escalation is a process rather than an isolated event.

Third, combining timing and content features yields the strongest performance, confirming that these feature families are complementary. While gains are incremental, they point to promising directions for richer multimodal signals such as user interaction graphs and image context, as well as deeper temporal models in future work.

Beyond academic implications, these findings have practical relevance for social media platforms. At k=10, a checkpoint can drive moderator actions: (i) prioritize threads predicted to escalate, (ii) apply gentle and reversible friction such as rate limits, subtle backgorund changes, warning nudges, or temporary slow-mode before storms fully form, and (iii) route uncertain or high-risk cases for human review using calibrated probabilities [31], [32]. These steps make early intervention practical and compatible with existing moderation workflows, enabling seamless integration into trust and safety pipelines.

E. Limitations

We label all comments using a single RoBERTa toxicity model and a fixed threshold, allowing CSS and early features to inherit domain shift and calibration error. We forecast based on only the first $k = 10$ comments-this choice improves timeliness but misses late-emerging patterns and assumes threads reach a length of k. We also screen out very short threads and very long spans (>180 days), which can shift the data distribution and limit external validity. We evaluate only English Reddit/Instagram and a single temporal snapshot; we do not test concept drift or cross-community transfer. Our analysis remains observational (MAE/RMSE/R^2, ROC–AUC), not causal: we do not estimate downstream harm reduction, subgroup fairness, or operational costs (e.g., moderator workload).

VII. CONCLUSION

This paper introduced a proactive framework for forecasting *neg storms*, by formalizing *Comment Storm Severity (CSS)* as a time-aware measure of escalation and *neg storms* as those CSS rising above a certain threshold. We demonstrate that early signals from the first k comments can predict future harm in the form of *neg storms*. Although performance is modest, these results validate the feasibility of anticipating toxic situations before they fully unfold. Practical implications include enabling platforms to prioritize high-risk threads, apply gentle and reversible friction such as rate limits or warning nudges, and route uncertain cases for human review using calibrated

probabilities. Moving beyond piecemeal toxicity detection toward situation-level modeling is essential for securing online communities, and this work provides an important starting point for advancing research on negative storms and proactive moderation.

VIII. Acknowledgments

The authors would like to thank Vipra Singh for suggesting the term *Neg Storm*.

References

[1] R. Hada, S. Sudhir, P. Mishra, H. Yannakoudakis, S. M. Mohammad, and E. Shutova, "Ruddit: Norms of offensiveness for english reddit comments," *arXiv preprint arXiv:2106.05664*, 2021.

[2] A. Sheth, V. L. Shalin, and U. Kursuncu, "Defining and detecting toxicity on social media: context and knowledge are key," *Neurocomputing*, vol. 490, pp. 312–318, 2022.

[3] Anjum and R. Katarya, "Hate speech, toxicity detection in online social media: a recent survey of state of the art and opportunities," *International Journal of Information Security*, vol. 23, no. 1, pp. 577–608, 2024.

[4] A. Massanari, "# gamergate and the fappening: How reddit's algorithm, governance, and culture support toxic technocultures," *New media & society*, vol. 19, no. 3, pp. 329–346, 2017.

[5] P. C. S. Andrews, "Social media futures: What is brigading?" Report, Tony Blair Institute for Global Change, 2021, last modified March 10, 2021. [Online]. Available: https://institute.global/policy/social-media-futures-what-brigading

[6] G. Beknazar-Yuzbashev, R. Jiménez-Durán, J. McCrosky, and M. Stalinski, "Toxic content and user engagement on social media: Evidence from a field experiment," CESifo Working Paper, Tech. Rep., 2025.

[7] M. Sap, D. Card, S. Gabriel, Y. Choi, and N. A. Smith, "The risk of racial bias in hate speech detection," in *Proceedings of the 57th annual meeting of the association for computational linguistics*, 2019, pp. 1668–1678.

[8] S. Kiritchenko, I. Nejadgholi, and K. C. Fraser, "Confronting abusive language online: A survey from the ethical and human rights perspective," *Journal of Artificial Intelligence Research*, vol. 71, pp. 431–478, 2021.

[9] M. Avalle, N. D. Marco, G. Etta, E. Sangiorgio, S. Alipour, A. Bonetti, L. Alvisi, A. Scala, A. Baronchelli, M. Cinelli, and W. Quattrociocchi, "Persistent interaction patterns across social media platforms and over time," *Nature*, vol. 628, no. 8008, pp. 582–589, 2024.

[10] B. Mathew, R. Dutt, P. Goyal, and A. Mukherjee, "Spread of hate speech in online social media," in *Proceedings of the 10th ACM conference on web science*, 2019, pp. 173–182.

[11] V. K. Singh, M. Gao, and R. Jain, "Situation recognition: an evolving problem for heterogeneous dynamic big multimedia data," in *Proceedings of the 20th ACM international conference on Multimedia*, 2012, pp. 1209–1218.

[12] L. Hebert, L. Golab, and R. Cohen, "Predicting hateful discussions on reddit using graph transformer networks and communal context," in *2022 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. IEEE, 2022, pp. 9–17.

[13] E. Chandrasekharan, M. Samory, S. Jhaver, H. Charvat, A. Bruckman, C. Lampe, J. Eisenstein, and E. Gilbert, "The internet's hidden rules: An empirical study of reddit norm violations at micro, meso, and macro scales," *Proceedings of the ACM on Human-Computer Interaction*, vol. 2, no. CSCW, pp. 1–25, 2018.

[14] M.-A. Rizoiu, Y. Lee, S. Mishra, and L. Xie, "Hawkes processes for events in social media," in *Frontiers of multimedia research*, 2017, pp. 191–218.

[15] P. Atrey, M. P. Brundage, M. Wu, and S. Dutta, "Demystifying the accuracy-interpretability trade-off: A case study of inferring ratings from reviews," 2025. [Online]. Available: https://arxiv.org/abs/2503.07914

[16] E. Wulczyn, N. Thain, and L. Dixon, "Ex machina: Personal attacks seen at scale," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 1391–1399.

[17] B. Vidgen and L. Derczynski, "Directions in abusive language training data, a systematic review: Garbage in, garbage out," *PLOS ONE*, vol. 15, no. 12, p. e0243300, 2020.

[18] R. Hanscom, T. S. Lehman, Q. Lv, and S. Mishra, "The toxicity phenomenon across social media," *arXiv preprint arXiv:2410.21589*, 2024.

[19] D. Soni and V. K. Singh, "Time reveals all wounds: Modeling temporal dynamics of cyberbullying sessions," in *Proceedings of the 12th International AAAI Conference on Web and Social Media (ICWSM)*, Stanford, CA, USA, 2018, pp. 684–691.

[20] S. Ranjith, C. R. Chowdary, and P. Tiwari, "Learning models to forecast toxicity in conversation threads by identifying potential toxic users," *Evolving Systems*, vol. 16, no. 1, pp. 8:1–8:16, 2025.

[21] S. Schoenebeck, C. Lampe, and P. Triu, "Online harassment: Assessing harms and remedies," *Social Media + Society*, vol. 9, no. 1, p. 20563051231157297, 2023.

[22] J. Smith, ""It Feels Like a Mini Victory": Alternative routes to justice in experiences of online misogyny," in *Experiences of Punishment, Abuse and Justice by Women and Families: Volume 2*, N. Booth, I. Masson, and L. Baldwin, Eds. Bristol, UK: Policy Press, 2023, pp. 110–131.

[23] S. Datta and E. Adar, "Extracting inter-community conflicts in reddit," in *Proceedings of the Thirteenth International AAAI Conference on Web and Social Media (ICWSM 2019)*, vol. 13, 2019, pp. 146–157.

[24] J. Cheng, M. Bernstein, C. Danescu-Niculescu-Mizil, and J. Leskovec, "Anyone can become a troll: Causes of trolling behavior in online discussions," in *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*. Portland, OR, USA: ACM, 2017, pp. 1217–1230.

[25] E. Menesini and A. Nocentini, "Cyberbullying definition and measurement: Some critical considerations," *Zeitschrift für Psychologie/Journal of Psychology*, vol. 217, no. 4, pp. 230–232, 2009.

[26] H.-T. Kao, S. Yan, D. Huang, N. Bartley, H. Hosseinmardi, and E. Ferrara, "Understanding cyberbullying on instagram and ask.fm via social role detection," in *Proceedings of the World Wide Web Conference (WWW Companion)*, San Francisco, CA, USA, 2019, pp. 183–194.

[27] C. Chelmis and D.-S. Zois, "Dynamic, incremental, and continuous detection of cyberbullying in online social media," *ACM Transactions on the Web*, vol. 15, no. 3, pp. 14:1–14:33, 2021.

[28] Q. Huang, V. Singh, and P. Atrey, "On cyberbullying incidents and underlying online social relationships," *J Comput Soc Sc*, vol. 1, p. 241–260, 2018.

[29] V. U. Gongane, M. V. Munot, and A. D. Anuse, "Detection and moderation of detrimental content on social media platforms: current status and future directions," *Social Network Analysis and Mining*, vol. 12, no. 1, p. 129, 2022.

[30] H. Hosseinmardi, S. A. Mattson, R. I. Rafiq, R. Han, Q. Lv, and S. Mishra, "Analyzing labeled cyberbullying incidents on the instagram social network," in *Social Informatics (SocInfo 2015)*, ser. Lecture Notes in Computer Science. Beijing, China: Springer, 2015, pp. 49–66.

[31] T. Gillespie, "Content moderation, ai, and the question of scale," *Big Data & Society*, vol. 7, no. 2, p. 2053951720943234, 2020.

[32] J. Park and V. K. Singh, "How background images impact online incivility," *Proceedings of the ACM on Human-Computer Interaction*, vol. 6, no. CSCW2, pp. 1–23, 2022.