Risk-aware Flow Tuning for Collective Emotion in Social Media via Multi-Agent Reinforcement Learning

Ruyi Wang
University of Edinburgh
Edinburgh, EH8 9AB, UK
r.wang46@ed.ac.uk

Robin L. Hill
University of Edinburgh
Edinburgh, EH8 9AB, UK
r.l.hill@ed.ac.uk

J. Michael Herrmann
University of Edinburgh
Edinburgh, EH8 9AB, UK
michael.herrmann@ed.ac.uk

ABSTRACT

Engagement-driven ranking can amplify moral and emotional content and destabilise collective affect. We ask whether platforms can reduce such tail risks without altering content or acting on user accounts by intervening only in the exposure flow layer. We model the platform as a controller on top of a topic-conditioned Independent Cascade with user-level affect dynamics and introduce two interpretable controls: friction, which slows and spaces diffusion through probability scaling and cooldowns; and balance, which reweights seeds and candidates to promote corrective and diverse exposure. Policies are learned with centralised training and decentralised execution using MAPPO with shard-level actors and a central critic. Engagement and diversity floors, per-step action budgets, and smoothness constraints are enforced through a primal–dual Lagrangian method. Using temporal risk metrics, including peak negative exposure rate and duration and area above a risk threshold, simulations and offline counterfactual evaluation on two Twitter datasets covering COVID-19 and mpox show consistent reductions relative to engagement-maximising ranking, removal of the top-k nodes, and uniform down-ranking, with modest engagement costs. The learned controls are sparse, interpretable, and auditable, offering a deployable middle ground between passive ranking and strict moderation by acting on exposure dynamics rather than content.

KEYWORDS

Flow control; Independent Cascade; Social-media governance; Multi-agent reinforcement learning; CTDE; MAPPO; Social computing; Mental wellness

ACM Reference Format:
Ruyi Wang, Robin L. Hill, and J. Michael Herrmann. 2026. Risk-aware Flow Tuning for Collective Emotion in Social Media via Multi-Agent Reinforcement Learning. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. https://doi.org/10.65109/KMUB2012

1 INTRODUCTION

Social media platforms increasingly curate exposure through engagement-driven ranking, which can amplify moral–emotional and negative content, and, in turn, destabilise collective affect and widen polarisation. Content removal or account bans may limit harm, but are coarse-grained and carry governance and speech costs. This paper studies whether platforms can mitigate tail risk in collective affect and polarisation *without altering content or user accounts/choices* by intervening only on the *flow layer*—the timing, sequencing, and selection of exposures delivered to users.

We formalise the setting as a diffusion–response system capturing two empirically salient mechanisms. First, information spreads along social ties; we use an Independent Cascade (IC) process with one-shot activation attempts and synchronous rounds [14]. Second, affect responds to exposure; we adopt a lightweight user-level update in which negative-topic impressions depress affect, while *corrective/positive comments and diverse exposures* partially offset this drift, echoing load/clearance/persistence ideas in contagion dynamics [13]. The diffusion kernel is *topic-conditioned* and fixed at control time (estimated from data; §3.1). We *compute* negative-topic exposure and affective responses for modelling and evaluation; however, these signals do *not* enter per-user action selection. Interventions regulate exposure dynamics only—they neither inspect nor edit content, nor alter user accounts or choices.

Designing effective flow-layer interventions raises three challenges: (i) *competing objectives*—managing risk while sustaining a minimum engagement level and respecting per-step action budgets, with topical diversity *monitored or softly encouraged* rather than enforced as a hard constraint; (ii) *multi-scale decision-making*—actions must matter at the user level yet be learnable at the platform scale; and (iii) *auditability*—controls should be interpretable and enforceable by policy. We address these by placing two interpretable levers atop the IC environment: *friction* ($[0, 1]$), which slows and spaces diffusion via probability scaling and cooldowns, and *balance* ($[0, 1]$), which reweighs seed/candidate selection to *pair negative-topic impressions with additional corrective/positive comments* and to increase exposure variety. Policies are learned with Centralised Training and Decentralised Execution (CTDE): shard-level actors choose friction/balance while a central critic supplies value baselines [9, 17, 25]. Engagement floors and per-step action caps are handled via a Lagrangian treatment within PPO with generalised advantage estimation [1, 18, 21, 22].

Under this framework, a policy maps shard-level observations on a directed interaction graph with a learned, topic-conditioned kernel (§2.3) and user-level affect dynamics to (friction, balance) $\in [0, 1]^2$, aiming to reduce (i) the *peak* negative-exposure rate, (ii) the *time/area* above a risk threshold, and (iii) a *polarisation* penalty (distributional divergence from a neutral baseline). Execution operates at user granularity; sharding aggregates decisions during training only (§3.2.2).

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/KMUB2012

1.1 Related work

Despite substantial progress, three gaps remain. *First*, most interventions rely on hard structural moderation or content/seed manipulation; these do not offer a flow-layer mechanism to *de-risk exposure dynamics* in situ (cf. MIR/RLDB designs that remove or throttle users [6, 23]). *Second*, even emotion-aware RL approaches often optimise who to block under contagion assumptions but do not learn *interpretable flow controls*—rate caps, cooldowns, queue batching, corrective mixing—that preserve a fixed, learned IC kernel and sit atop standard ranking; balanced-influence studies support RL for *exposure balance* yet emphasise *seed selection* over runtime scheduling/mixing [24]. *Third*, evaluation commonly reports infection sizes or end states rather than *risk-sensitive temporal* metrics (e.g., peak negative-exposure rate, time-above-threshold) aligned with operations; post-event simulations show managerial levers reshape temporal trajectories [5], but a principled learning framework with *engagement constraints*, *auditable* policies, and explicit *flow-layer* controls is uncommon. We address these gaps; see see supplementary material (Github) for an expanded review.

1.2 Our contribution.

(1) **Flow-layer control without altering content or accounts.** We cast platform intervention as regulating exposure dynamics over a topic-conditioned IC environment with a fixed learned kernel and user-level affect response (§3.1), avoiding content inspection or editing and not modifying user accounts or choices.
(2) **Learned, interpretable levers under operational budgets.** We introduce two interpretable controls—*friction* and *balance*—and learn shard-level policies via MAPPO with a Lagrangian treatment of *engagement* budgets and per-step action caps (§3.2, §3.2.4). Balance explicitly *pairs negative-topic impressions with corrective/positive comments* drawn from existing content.
(3) **Risk-aware evaluation.** We align metrics with governance objectives using temporal-risk measures (peak and time/area above threshold, using a smooth peak proxy in optimisation) and a polarisation metric based on Jensen–Shannon divergence, and evaluate via simulations and offline counterfactuals (§4.3, §4.4).

This approach complements blocking and downranking heuristics by offering an auditable middle ground between passive ranking and hard moderation, targeting exposure dynamics rather than content.

2 PRELIMINARIES

2.1 Topic Clustering

Within each event (COVID–19; mpox) we identify a set of *stable and interpretable* sub-topics that capture semantic regularities in short, noisy social-media texts and serve as conditioning variables for the topic-specific IC kernel in §2.3. Topic discovery (content-side structure) is deliberately separated from decision making (flow-layer control): *we never alter content or user accounts/choices.* Coarse topic valence and affect descriptors are computed only for modelling and evaluation, and they do *not* enter per-user action selection.

We adopt a clustering-based topic-modelling pipeline that combines transformer sentence embeddings, density-based clustering, and class-based term scoring [7, 11, 19]. Tweets are first normalised (language filtering, near-duplicate removal, URL/emoji handling, and hashtag segmentation). We then compute sentence-level embeddings with Sentence–BERT [19]. In the embedding space, we apply HDBSCAN to obtain topics and to isolate noise without pre-specifying the number of clusters [3, 16]. Topic labels and human-auditable descriptors are derived using BERTopic's class-based TF–IDF (c–TF–IDF) procedure [11].

We do not preset the number of topics K. The similarity metric is fixed to cosine distance, and HDBSCAN's `min_cluster_size` and `min_samples` are the only tunable parameters. A *single* selection criterion is used: on an independently held validation window, we compute the average NPMI topic-coherence score across all discovered topics [20] and choose the parameter setting that maximises this score. This setting is then held fixed for all experiments.

To reduce boundary artefacts in downstream diffusion estimation, we compute soft membership for document i to topic k via a temperature-scaled cosine kernel, $\pi_{ik} \propto \exp(\cos(x_i, c_k)/\tau)$, where x_i is the embedding, c_k the topic centroid, and τ a temperature. We monitor centroid drift across adjacent time windows; exceedance of a pre-registered threshold triggers a merge/split review without altering the fixed parameter setting.

For each topic we also report the average valence (with confidence intervals) of constituent tweets using a social-media-oriented analyser (VADER) to provide a coarse positive/negative/neutral descriptor for reporting and fairness diagnostics; this descriptor is not used as a decision variable [12].

Applying the pipeline to the two Kaggle Twitter corpora yields K=28 *topics for COVID–19* and K=9 *topics for mpox*. To preserve space while ensuring auditability, the *complete topic ID–label lists* are provided in the see supplementary material (Github).

2.2 User Engagement Propensity and Sensitivity

Aim. We estimate two user-level quantities that capture behavioural and affective heterogeneity: *engagement propensity* (the probability of interacting conditional on exposure) and *sensitivity* (the affective change per unit of negative effective exposure over a short horizon). Both quantities are treated as fixed *environment parameters* in simulation and descriptive analyses; they are *not* decision variables and are never targeted by the control policy.

Notation. Let i index users, x candidate items (tweets), and t discrete time windows. Let $\text{Engage}_{ixt} \in \{0, 1\}$ denote whether user i interacted with item x when visible at time t. Let $a_i(t) \in [-1, 1]$ be the affect (valence) of user i at t, with increment $\Delta a_{i,t} = a_i(t) - a_i(t-1)$. Let $\text{neg}_i(t)$ denote the (de-duplicated) count of *negative effective exposures* received by user i in window t (topic-conditioned; see §2.3). Let $s(i)$ be the structural–homogeneity shard of user i used only for random-effect pooling.

Engagement propensity. We fit a logistic mixed-effects model with user and shard random intercepts to obtain calibrated exposure–response probabilities,

$$\Pr\big(\text{Engage}_{ixt} = 1 \mid g_{ixt}\big) = \sigma\Big(\theta^\top g_{ixt} + a_i^{(0)} + b_{s(i)}\Big), \qquad (1)$$

where $\sigma(\cdot)$ is the logistic link, g_{ixt} are exposure-time features (recency, prior interactions, structural proximity, position-in-queue, etc.), $a_i^{(0)} \sim \mathcal{N}(0, \sigma_a^2)$ is a user random intercept, and $b_s \sim \mathcal{N}(0, \sigma_b^2)$ a shard random intercept. Parameters are estimated by maximum likelihood with a Laplace approximation for random effects [2]. Predicted probabilities are *isotonically calibrated* on a chronological validation split and evaluated by AUC and Brier score; the calibrated outputs constitute the *engagement propensity* used in the simulator and as a minimum-engagement target [26].

Sensitivity to negative effective exposure. Following our definition, user i's *sensitivity* is the marginal affective change per unit negative exposure over a short horizon $\Delta > 0$:

$$s_i \stackrel{\text{def}}{=} -\frac{\partial\, a_i(t+\Delta)}{\partial\, \mathrm{neg}_i(t)}. \tag{2}$$

The minus sign ensures $s_i > 0$ corresponds to a deterioration in affect after negative exposure. With discrete windows we approximate (2) by a local distributed-lag response (horizon L):

$$\Delta a_{i,t} = -\,s_i\,\mathrm{neg}_i(t-1) \;-\; \sum_{\ell=2}^{L} \eta_\ell\,\mathrm{neg}_i(t-\ell) \;+\; \gamma^\top Z_{i,t} \;+\; u_{s(i)} \;+\; \varepsilon_{i,t}, \tag{3}$$

where $Z_{i,t}$ collects controls (baseline affect, time-of-day dummies, recent engagement volume, topic-mix share), $u_{s(i)} \sim \mathcal{N}(0, \sigma_u^2)$ is a shard random intercept, and $\varepsilon_{i,t} \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. Individual sensitivities follow a Gaussian random-coefficient prior,

$$s_i \sim \mathcal{N}(\mu_s, \sigma_s^2). \tag{4}$$

We estimate (3)–(4) by *maximum marginal likelihood* with a Laplace approximation that integrates out random effects (empirical Bayes). We set short lags ($L \in \{2, 3\}$) to mitigate confounding with slow trends and winsorise extreme exposure counts at the 99th percentile. Empirical Bayes estimates $\hat{s}_i$ (posterior modes) and their standard errors are reported; shard-level distributions of $\hat{s}_i$ summarise heterogeneity.

Diagnostics and splits. All models use chronological *train* (estimation), *validation* (calibration/early stopping), and *test* (reporting) splits. For (1) we report calibrated AUC and Brier on the test window. For (3) we report out-of-sample R^2, sign stability of $\hat{s}_i$ across folds, and residual autocorrelation checks at the reporting granularity. Extreme counts are winsorised.

Use in simulation (environment only). In the user-level IC environment, affect evolves as

$$a_i(t+1) = a_i(t) - \hat{s}_i\,\mathrm{neg}_i(t) + \xi \cdot \mathrm{balHit}_i(t) + \varepsilon_{i,t}, \tag{5}$$

where $\mathrm{balHit}_i(t)$ counts *corrective/diverse impressions* surfaced by the balance action (e.g., counterpoints, replies or positive comments drawn from existing content) and scaled by ξ. We clip $a_i(\cdot) \in [-1, 1]$ and cap $\mathrm{neg}_i(t)$ to unique impressions per window. The MARL agents observe aggregate summaries (engagement, risk, topic mix) but do not access, target, or modify individual $\hat{s}_i$ or propensities; these parameters remain fixed throughout training and evaluation.

2.3 Topic-Conditioned IC Kernel

We estimate a topic-conditioned Independent Cascade (IC) kernel that maps user–user pairs to activation probabilities, conditional on the sub-topic of a message. The kernel provides the baseline diffusion dynamics used by the simulator; policy actions in §3 modulate these probabilities at run-time but do not alter their estimation.

Let $(u \to v) \in E$ denote a directed edge and $k \in \{1, \ldots, K\}$ a discovered topic. The per-topic edge activation probability is

$$p_{uv}^{(k)} \;=\; \sigma\Big(\theta^\top f_{uv} \;+\; b_k \;+\; r_u \;+\; r_v\Big) \in (0, 1), \tag{6}$$

where σ is the logistic link, f_{uv} are time-invariant or slowly varying features (historical interaction counts, structural proximity, content similarity, follower overlap, etc.), b_k is a topic bias, and r_u, r_v are sender/receiver random effects ($r_\cdot \sim \mathcal{N}(0, \sigma_r^2)$) to mitigate popularity imbalance. For additional parsimony one may use a low-rank form $f_{uv} = \phi(u) \odot \psi(v)$ (element-wise product of user embeddings), but we keep (6) general.

Training data and soft topic assignment. From time-ordered interaction logs (retweet/reply/mention), we extract candidate *trigger* pairs $\langle u, t\rangle \rightsquigarrow \langle v, t'\rangle$ with $t<t'$ and within a finite window. Each source message x carries soft topic membership π_{xk} from §2.1. A positive instance $(u \to v, k)$ receives weight $w^+_{uvk} = \sum_{x \in \mathcal{X}_{u\to v}} \pi_{xk}$; time-matched non-activations $(u \to v', k)$ form negatives with weight $w^-_{uv'k}$ (stratified by activity to control class imbalance).

Estimation and calibration. Parameters $\theta, \{b_k\}, \{r_u\}, \{r_v\}$ are obtained by maximising a weighted Bernoulli log-likelihood with ℓ_2 regularisation and Gaussian priors for random effects:

Weighted log-likelihood.

$$\mathcal{L}_{\mathrm{wll}}(\theta, b, r) = \sum_{(u,v,k)\in\mathcal{P}} w^+_{uvk}\,\log p_{uv}^{(k)} \;+\; \sum_{(u,v,k)\in\mathcal{N}} w^-_{uvk}\,\log\big(1-p_{uv}^{(k)}\big). \tag{7}$$

Regularisation.

$$\mathcal{R}(\theta, b, r) = \lambda\,\|\theta\|_2^2 + \frac{1}{2\sigma_r^2}\left(\sum_u r_u^2 + \sum_v r_v^2\right) + \frac{1}{2\sigma_b^2}\sum_k \big(b_k - b_0\big)^2. \tag{8}$$

Objective.

$$\max_{\theta,b,r} \;\; \mathcal{L}_{\mathrm{wll}}(\theta, b, r) \;-\; \mathcal{R}(\theta, b, r). \tag{9}$$

We optimise (9) under a *maximum marginal likelihood* view, using a Laplace approximation to integrate out random effects. On a chronological validation window we perform isotonic calibration of the predicted probabilities; the calibration mapping is then fixed for all subsequent runs. For numerical stability, final probabilities are clipped to $[\varepsilon,\ 1-\varepsilon]$ with a small constant $\varepsilon = 10^{-4}$.

Kernel quality is reported per topic: calibrated AUC, Brier score, and reliability curves. As a behavioural check, we simulate IC cascades on held-out graphs using the *live-edge* view [14] and compare cascade size/depth distributions against empirical ones. We also report a branching proxy $\hat{m}_k = \frac{1}{|A_k|}\sum_{(u,v)\in A_k} p_{uv}^{(k)}$ (average per-edge activation in topic k), which should remain < 1 in realistic regimes.

Interface to simulation and topics with soft membership. At runtime, when a message x with membership π_{xk} is propagated from u to a neighbour v, the baseline activation probability is

$$p_{uv}^{(x)} \;=\; \sum_{k=1}^{K} \pi_{xk}\, p_{uv}^{(k)}. \tag{10}$$

Policy actions in §3 then yield an *effective* probability, e.g. for friction λ_f:

$$p_{\text{eff}}(u \to v \mid x, t) \;=\; \text{clip}\Big((1 - \eta_f\, \tilde{\lambda}_f(u,t))\; p_{uv}^{(x)}\;,\; \varepsilon,\; 1-\varepsilon\Big), \quad (11)$$

where $\tilde{\lambda}_f$ is the shard-to-user mapping, and η_f a small scaling constant. Balance acts primarily through seeding/candidate re-weighting rather than edge probabilities; see §3 for details. The learned kernel $p_{uv}^{(k)}$ itself is fixed across training and evaluation and is not altered by the policy.

3 METHODOLOGY

Overview. We develop a control framework that regulates *exposure dynamics* rather than content or user accounts/choices. The environment is a user-level, topic-conditioned Independent Cascade (IC) simulator (§3.1) whose baseline diffusion kernel is learned in preliminaries and held fixed during control. On top of this environment, we apply a Centralised Training with Decentralised Execution (CTDE) multi-agent scheme (§3.2): during training, actors operate at the shard level to reduce variance (§3.2.1), while evaluation executes the learned policy on the full user graph (§3.2.2). The action space consists of two interpretable, flow-layer levers (§3.2.3): *friction* $[0, 1]$ slows and spaces diffusion via probability scaling and cooldowns, and *balance* $[0, 1]$ re-weights seeds/candidates to inject corrective and diverse exposure drawn from existing content (e.g., counterpoints/replies/positive comments). The objective minimises a *smooth peak* of negative-exposure rate, time/area above a risk threshold, and polarisation, subject to a minimum engagement floor, per-step action budgets, and smoothness limits (§3.2.4). Diffusion and affect are always simulated at user granularity; sharding aggregates *decisions* in training only. Implementation constants (step size, horizon, clipping and cooldown scales) and all training hyperparameters are reported in Section 4.

3.1 Topic-Conditioned IC Environment (User-Level Simulation)

Aim. The environment simulates exposure dynamics at the *user* level under a topic–conditioned Independent Cascade (IC) process with one-shot activation attempts and synchronous rounds [14]. IC cleanly separates (i) fixed, learned edge-wise activation probabilities from (ii) run-time control over exposure timing and selection, matching our flow-layer intervention design.

State and data.

- **Graph layer.** A directed graph $G = (V, E)$ from interaction histories (retweet/reply/mention). Edge features and user metadata are cached for fast access [10].
- **Topic–conditioned kernel.** Baseline edge probabilities $p_{uv}^{(k)}$ are estimated in §2.3. For a message x with soft topic membership π_{xk}, the baseline activation on $(u \to v)$ is

$$p_{uv}^{(x)} \;=\; \sum_{k=1}^{K} \pi_{xk}\, p_{uv}^{(k)}. \quad (12)$$

- **User affect and propensity.** Calibrated engagement propensity and affect sensitivity $\hat{s}_i$ are fixed environment parameters from §2.2; the policy never accesses or targets these per-user parameters.

Initial seeding. At t=0 a seed set $\mathcal{S}_0$ and associated messages $\mathcal{X}_0$ are drawn from logged feeds or a small exogenous pool, consistent with standard IC initialisation [14]. Balance (when active; §3.2.3) re–weights seed sampling only. The initial active set is $\mathcal{A}_0 = \mathcal{S}_0$.

Propagation (batched one–shot IC).. For rounds $t = 0, 1, \ldots,$ let the frontier be $\mathcal{F}_t = \mathcal{A}_t \setminus \mathcal{A}_{t-1}$. Each $u \in \mathcal{F}_t$ attempts to activate each out–neighbour v *once* for the message x carried by u. Friction acts at the flow layer; the effective probability is

$$p_{\text{eff}}(u \to v \mid x, t) = \text{clip}\Big((1 - \eta_f\, \tilde{\lambda}_f(u,t))\; p_{uv}^{(x)}\;,\; \varepsilon,\; 1-\varepsilon\Big), \quad (13)$$

where $\tilde{\lambda}_f(u,t) \in [0,1]$ is the user-level friction mapped from shard actions (§3.2.3), η_f a small scale, and ε a numerical clip. A Bernoulli draw with parameter p_{eff} finalises the one–shot attempt. Cooldowns enforce a minimum inter–attempt delay for repeated exposures of the same topic to a user, operationalising "slowing and spacing" without altering content (concrete settings in §4). Newly activated targets form $\mathcal{N}_{t+1}$ and update $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \mathcal{N}_{t+1}$.

Exposure, engagement, and affect updates. All impressions at time t contribute to $\text{neg}_i(t)$ (de–duplicated within the window). Interaction events are sampled using calibrated engagement propensity; this decouples diffusion mechanics from user response while maintaining realistic utility traces. User affect then evolves as

$$a_i(t{+}1) = a_i(t) - \hat{s}_i\, \text{neg}_i(t) + \xi \cdot \text{balHit}_i(t) + \varepsilon_{i,t}, \quad (14)$$

where $\text{balHit}_i(t)$ counts corrective/diverse impressions surfaced by balance (e.g., counterpoints, replies or positive comments drawn from existing content) and scaled by ξ.

Termination. A cascade stops when $\mathcal{N}_{t+1} = \varnothing$ (no new activations) or a horizon $T_{\max}$ is reached, following the standard IC stopping rule [14]. Multi–message runs interleave messages with the same IC rules and per–topic kernels; a live–edge implementation may be used for efficiency without changing semantics [4].

3.2 CTDE–MAPPO

We adopt Centralised Training with Decentralised Execution (CTDE) to stabilise multi-agent learning while retaining scalable execution [9, 15, 17]. Concretely, we use Multi-Agent PPO (MAPPO) [25] with a central critic and per-shard actors. PPO's clipped surrogate and generalised advantage estimation (GAE) provide robust on-policy updates [21, 22]. Operational constraints are enforced via a primal–dual Lagrangian formulation [1, 18].

3.2.1 Sharding (Decision Aggregation).

Partitioning and mapping. Users are partitioned once into S shards using structural communities (e.g., modularity-based detection) *refined* by user-level homogeneity (engagement propensity/sensitivity). We support hard membership or soft weights $\pi_{i,s}$ with $\sum_s \pi_{i,s} = 1$. Shard actions are mapped to users by

$$\tilde{\lambda}_{\{\cdot\}}(i,t) \;=\; \sum_{s=1}^{S} \pi_{i,s}\, \lambda^{s}_{\{\cdot\}}(t), \quad (15)$$

and applied in the user-level IC environment (§3.1). Diffusion and affect are *always* simulated at user granularity.

Instantiation. We instantiate the refinement by a Cartesian partition: structural communities $\{C_b\}_{b=1}^{B}$ (e.g., Louvain) crossed with feature buckets $\{\mathcal{H}_h\}_{h=1}^{H}$ derived from engagement propensity and sensitivity, yielding $S = B \times H$ shards. Soft membership is computed as the normalised product of structural and feature affinities, $\pi_{i,s} \propto \pi_{i,b}^{\text{struct}} \cdot \pi_{i,h}^{\text{feat}}$ with $\sum_s \pi_{i,s} = 1$. (Specific values of B, H and detection resolutions are given in Section 4.)

3.2.2 Training–Evaluation Regime (Sharded Training, Full-Network Evaluation).

Training (CTDE with sharded actors). During learning, each shard s observes a local summary $o_s(t)$ (engagement volume, negative-exposure rate, polarisation proxies, topic mix, recent actions, remaining budgets, cooldown rate) and outputs $a_s(t) = (\lambda_f^s, \lambda_b^s)$. Actions are mapped via (15) and the user-level environment advances. A central critic V_ψ consumes global aggregates; CTDE permits richer value baselines without breaking decentralised execution.

Observations. Each shard observes a normalised summary

$$o_s(t) = \big[\widehat{\text{exposures}},\ \widehat{\text{engagement}},\ \widehat{r}_t,\ \widehat{\text{JS}},\ \widehat{\text{topic-share}}_{1:K'},\ \lambda_f^s(t-1),\ \lambda_b^s(t-1),\ \widehat{\text{cooldown-rate}},\ \widehat{\text{budget-remaining}}\big]$$

computed over a sliding window and z-score normalised with clipping to $[-3, 3]$. Edge-level friction uses the *source-side* mapping $\tilde{\lambda}_f(u, t)$ for $(u \to v)$.

Evaluation (full graph, user granularity). At test time learning is disabled; the same policy runs on the *full* user graph with user-level diffusion and affect updates. Mapping (15) remains for soft membership; no state averaging is introduced.

3.2.3 Action Space.

Friction $[0, 1]$. Each shard outputs $\lambda_f^s(t) \in [0, 1]$. User-level friction $\tilde{\lambda}_f(i, t)$ modulates *speed and spacing* of diffusion without touching content:

(1) *Probability scaling:* use (13) to scale IC edge probabilities; this implements throttling at the exposure (flow) layer.
(2) *Cooldowns/batching:* enforce a minimum inter-attempt delay $\Delta t \geq \tau_f\, \tilde{\lambda}_f(i, t)$ for repeated exposures of the same topic to i, optionally batching releases (details in §4).

Balance $[0, 1]$. Each shard outputs $\lambda_b^s(t) \in [0, 1]$. User-level balance $\tilde{\lambda}_b(i, t)$ re-weights seed/candidate selection to inject corrective and diverse exposure strictly from existing content:

$$\text{logit}\, q_i(x, t) \ \leftarrow\ \text{logit}\, q_i(x, t) \ +\ \delta_b(x, t)\, \tilde{\lambda}_b(i, t), \tag{16}$$

where $q_i(x, t)$ is the pre-policy sampling/ranking probability and $\delta_b(x, t)$ is a bounded topic-aware offset calibrated on validation. The balance mechanism also contributes to $\text{balHit}_i(t)$ in the affect update (14) when corrective/positive comments or cross-topic counterpoints are successfully surfaced.

3.2.4 Objective & Constraints.

Objective. Let r_t be the population negative-exposure rate and P_t the affect distribution at time t; Q is a neutral baseline. We minimise

$$\mathcal{J}(\pi) = w_1 \underbrace{\tau_{\text{soft}} \log \sum_{t=1}^{T} \exp\big(r_t/\tau_{\text{soft}}\big)}_{\text{smooth peak}} + w_2 \underbrace{\sum_{t=1}^{T} (r_t - \tau)_+}_{\text{time/area above threshold}} + w_3 \underbrace{\tfrac{1}{T} \sum_{t=1}^{T} \text{JS}(P_t \| Q)}_{\text{polarisation}}, \tag{17}$$

where $\tau_{\text{soft}} > 0$ controls the soft-peak sharpness. The weights $w_{1:3}$ and threshold τ are set on a validation split using a pre-registered rule (e.g., baseline-quantile or small grid with Pareto selection); settings are then fixed for test reporting.

Constraints and budgets.

$$\mathbb{E}[\text{Engagement}] \ \geq\ \underline{E}, \tag{18}$$

$$\sum_i \tilde{\lambda}_f(i, t) \ \leq\ B_f, \qquad \sum_i \tilde{\lambda}_b(i, t) \ \leq\ B_b, \quad \forall t, \tag{19}$$

$$|\lambda_{\{\cdot\}}^s(t) - \lambda_{\{\cdot\}}^s(t-1)| \ \leq\ \kappa \quad \text{(per-shard smoothness)}. \tag{20}$$

We enforce (18)–(20) via a primal–dual scheme alongside PPO:

$$g_E(\pi) \ =\ \underline{E} - \widehat{\text{Engagement}}(\pi), \tag{21}$$

$$g_f(\pi, t) \ =\ \Big(\sum_i \tilde{\lambda}_f(i, t) - B_f\Big)_+, \qquad g_b(\pi, t) \ =\ \Big(\sum_i \tilde{\lambda}_b(i, t) - B_b\Big)_+. \tag{22}$$

Instantaneous penalised objective:

$$\mathcal{L}_t(\pi, \alpha, \mu, \nu) \ =\ \mathcal{J}_t(\pi) + \alpha\, g_E(\pi) + \mu\, g_f(\pi, t) + \nu\, g_b(\pi, t), \qquad \alpha, \mu, \nu \geq 0, \tag{23}$$

Episode Lagrangian and saddle-point problem:

$$\min_{\pi}\ \max_{\alpha, \mu, \nu \geq 0}\ \sum_{t=1}^{T} \mathcal{L}_t(\pi, \alpha, \mu, \nu). \tag{24}$$

Dual ascent (projected) updates:

$$\begin{aligned} \alpha &\leftarrow \big[\alpha + \eta_\alpha\ \textstyle\sum_t g_E(\pi)\big]_+, \\ \mu &\leftarrow \big[\mu + \eta_\mu\ \textstyle\sum_t g_f(\pi, t)\big]_+, \\ \nu &\leftarrow \big[\nu + \eta_\nu\ \textstyle\sum_t g_b(\pi, t)\big]_+. \end{aligned} \tag{25}$$

Optimisation notes. Dual ascent uses Adam with learning rates $(\eta_\alpha, \eta_\mu, \eta_\nu)$ tuned on validation; an entropy bonus with coefficient α_{ent} is added to the PPO objective to sustain exploration in continuous $[0, 1]$ actions. Concrete hyperparameters are listed in Section 4.

Training loop (sketch).

(1) Reset the user-level environment. For $t = 1{:}T$, each shard observes $o_s(t)$ and samples $a_s(t) = (\lambda_f^s, \lambda_b^s)$; map to users via (15) and step the IC environment (§3.1).
(2) Accumulate rewards (negative of the terms in (17)) and constraint signals; compute advantages with the central critic V_ψ (GAE).

(3) Update actors by PPO (clipped surrogate) and the critic by TD; ascend dual variables via (25). Freeze the policy for evaluation on the full graph (§3.2.2).

4 EXPERIMENTS

4.1 Datasets and Pre-processing

We use two public Twitter corpora (COVID–19; mpox) from Kaggle. The interaction network is built from retweet/reply/mention edges with timestamps; topic soft-memberships π_{xk} follow §2.1. Estimated engagement propensity and sensitivity are used as fixed environment parameters per §2.2. All splits are chronological (train/validation/test) to respect causal order.

4.2 Experimental Setup

Seeding. Initial seeds $\mathcal{S}_0$ are drawn from logged first-post events (stratified by topic); the exogenous injection ratio is set to 0.

IC environment constants. We fix the step size Δt, horizon $T_{\max}$, probability clipping ε, and friction scale η_f as reported in see supplementary material (Github). **Cooldowns:** repeated exposures of the *same dominant topic* to user i must satisfy a minimum inter-attempt delay

$$\Delta t \;\geq\; \tau_f \,\tilde{\lambda}_f(i,t),$$

applied within a sliding window of W minutes. The execution order per step is: deduplicate within-window → apply cooldown rule → scale probabilities by (13). Concrete choices (τ_f, W) are listed in see supplementary material (Github).

Sharding instantiation (no theory repeated). We instantiate sharding as in §3.2.1: structural communities (e.g., Louvain) crossed with feature buckets of (engagement propensity, sensitivity), yielding $S = B \times H$ shards. We report (B, H, S) and membership type (hard/soft) in see supplementary material (Github).

Training regime. Training uses the CTDE–MAPPO pipeline in §3.2 with sharded decisions and user-level simulation. Evaluation executes the frozen policy on the full graph (no aggregation), per §3.2.2. Kernel parameters are learned on training logs; validation/test episodes are held out chronologically to avoid leakage.

Thresholds and tuning rules. The risk threshold τ used in time-to-clear and exceedance metrics is fixed *a priori* on the validation split as the 80th percentile of baseline (engagement-ranking) r_t across episodes; the same τ is used for all methods on test. The smooth-peak temperature τ_{soft} is selected from a small grid $\{0.01, 0.02, 0.05, 0.1\}$ by validation loss and then fixed.

4.3 Evaluation Protocol

Policies are evaluated on the full user graph with user-level IC diffusion and affect updates (§3.1), rolling topics in chronological order within each episode to avoid look-ahead bias. Model selection uses the validation split; test numbers are reported once at the selected checkpoint. We report means and 95% confidence intervals computed by paired non-parametric bootstrap at the episode level [8]. Where multiple metrics are compared across conditions, p-values are adjusted by Benjamini–Hochberg FDR control.

4.4 Evaluation Metrics

4.4.1 Peak Negative-Exposure Rate. Let r_t denote the population share of negative-topic impressions at time t. We report Peak = $\max_t r_t$ per episode and average across runs (the training objective uses a smooth surrogate; reporting uses the true peak). Peak load is standard for short-term stress quantification in contagion/capacity-risk analyses [13].

4.4.2 Time-to-Clear. Given the fixed threshold τ (§4.2), define $t^\star = \min\{t : r_t \leq \tau\}$ (censored at $T_{\max}+$ if not reached); we also report the exceedance area $\sum_t (r_t - \tau)_+$. These capture *persistence* and above-threshold burden, complementing peak [13].

4.4.3 Polarisation. Let P_t be the empirical affect distribution and Q a neutral baseline estimated from pre-intervention periods. We report $\frac{1}{T}\sum_t \mathrm{JS}(P_t \| Q)$ (Jensen–Shannon divergence) and the inter-quartile range of $a_i(t)$ as dispersion proxy.

4.5 Baselines and Ablations

Protocol parity. All baselines use the *same* seeds (§4.2), IC constants (§4.2), and evaluation regime (§4.3) as our method. Hyperparameters are selected on the validation split to satisfy the *same engagement floor* as in (18); test results are reported once at the selected setting. Topics and user parameters are held fixed across methods (no content or account interventions).

k-Node Removal (Targeted structural suppression). We remove the top-k users prior to simulation (drop all incident edges) and then run the topic-conditioned IC on the remaining graph. Selectors:

(1) *Degree-k*: rank by out-degree (ties by in-degree).
(2) *Betweenness-k*: rank by betweenness centrality on the directed graph.
(3) *Influence-k*: rank by estimated negative-topic influence under the learned IC kernel (expected one-hop negative activations).

We sweep $k \in \{0, \ldots, k_{\max}\}$ on validation and choose the smallest k that meets the engagement floor; this yields a content-agnostic yet *maximally invasive* baseline (account removal/freeze).

Downranking (Uniform transmission throttling). We uniformly scale edge activation probabilities for negative-labelled topics by a constant factor $\rho \in (0, 1)$:

$$p_{uv}^{(k)} \;\leftarrow\; \rho\, p_{uv}^{(k)} \qquad \text{for negative-labelled topic } k.$$

This implements a *global* flow-layer throttle that does not use user- or topic-structure beyond the sign. We sweep ρ on validation and pick the largest ρ that still satisfies the engagement floor. This baseline contrasts with our *stateful*, shard- and topic-aware control (friction/balance).

Ablations (ours). We report *Friction-only* ($\lambda_b \equiv 0$) and *Balance-only* ($\lambda_f \equiv 0$) variants. Both inherit all training/evaluation settings from the full method; per-step action budgets and smoothness constraints (§3.2.4) are kept unchanged.

Reporting. Alongside point estimates we plot validation trade-off curves for k-Node Removal and Downranking (peak risk vs. engagement) and mark the selected operating points used for test

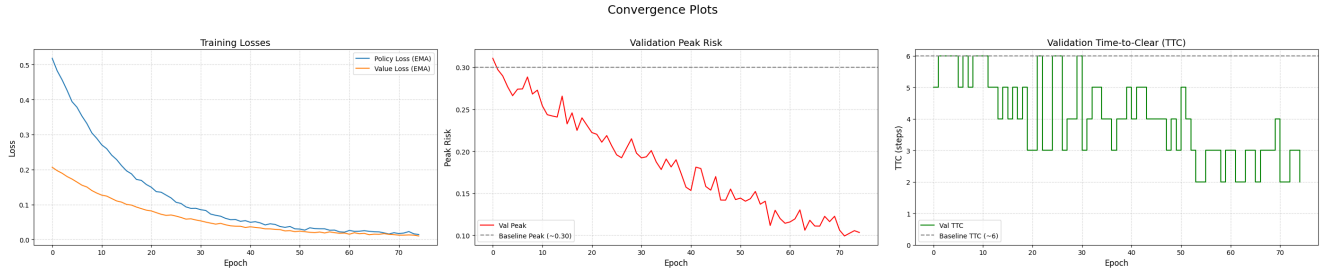

Figure 1: Convergence plots showing the agent's learning progress. (Left) Training losses for the policy and value networks. (Center) Validation Peak Risk over epochs. (Right) Validation Time-to-Clear (TTC) over epochs.

reporting. Full hyperparameter grids and the selected (k, ρ), as well as $(\tau, \tau_{\text{soft}}, \tau_f, W)$, are listed in see supplementary material (Github).

4.6 Implementation Details

We report CTDE–MAPPO hyperparameters (PPO/GAE, entropy coefficient, learning rates, rollout length, minibatch and epochs, parallel environments, gradient clipping), sharding instantiation (B, H, S), and environment constants $(\Delta t, T_{\max}, \varepsilon, \eta_f, \tau_f, W, \tau, \tau_{\text{soft}})$ in see supplementary material (Github). Experiments run on a single GPU; random seeds and config files are released for reproducibility.

5 RESULTS & DISCUSSION

To evaluate the effectiveness of our proposed risk-aware flow tuning method, we conducted a series of experiments. We first demonstrate the convergence of our MARL agent, then compare its performance against standard baselines across the three core objectives, and finally, perform an ablation study to understand the contribution of its core components.

5.1 Convergence of the Learning Process

A prerequisite for an effective policy is the ability of the agent to learn from its environment. Figure 1 illustrates the learning dynamics of our MARL agent over 75 training epochs.

The training process demonstrates successful learning and convergence. The **Training Losses** (left panel) for both the policy and the value networks show a steady decrease, eventually stabilizing at a low value, which indicates that the optimization algorithm is working correctly. More importantly, the primary validation metrics show clear improvement. The **Validation Peak Risk** (centre panel) exhibits a consistent downward trend, starting from a baseline of approximately 0.30 and being reduced by more than 50

5.2 Main Results: Comparison to Baselines

We compare our method against three alternatives: Non-Operation (i.e. no intervention), k-Node Removal, and uniform Downranking. Figure 2 presents a comprehensive comparison across the three core objectives defined in our study.

The results clearly demonstrate the superiority of our proposed method. In terms of controlling the risk trajectory, **Our Method** achieves the lowest **Peak Risk** (0.237), outperforming both k-Node Removal (0.283) and Downranking (0.418). Furthermore, it is most effective at reducing the duration and severity of the risk event,

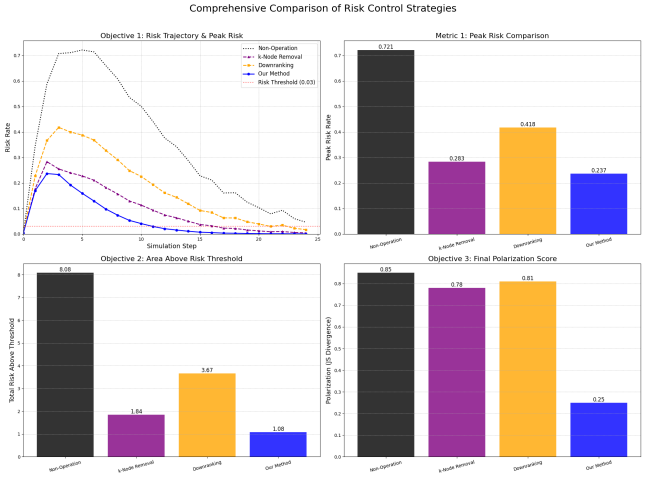

Figure 2: Comprehensive comparison of our method against baselines across three objectives: risk trajectory, peak risk, area above threshold, and final polarization score.

achieving the lowest **Area Above Threshold** (1.08), significantly better than k-Node Removal (1.84) and especially Downranking (3.67), whose long-tail effect is evident in its large integrated risk.

The most significant advantage of our method is revealed in the **Final Polarization Score**. While k-Node Removal and Downranking are blunt instruments that do little to mitigate emotional divergence (scores of 0.78 and 0.81, respectively), our method's 'balance' lever actively manages the emotional landscape, resulting in a drastically lower polarization score of 0.25. This shows that our method is unique in its ability to simultaneously control the spread of negative content and promote collective emotional stability.

5.3 Ablation Studies

To understand the individual contributions of the 'friction' and 'balance' levers in our model, we conducted an ablation study. We compare the full model ("Our Method") against two "crippled" versions: one that can only use friction ("Friction-Only") and one that can only use balance ("Balance-Only"). The results are shown in Figure 3.

The study reveals that each lever plays a distinct and crucial role.

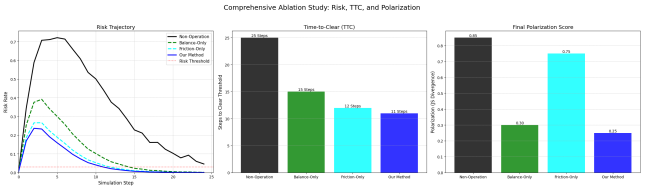

Figure 3: Ablation study results, comparing the full method against versions with only one lever active. (Left) Risk Trajectory, (Center) Time-to-Clear, (Right) Final Polarization Score.

5.3.1 Friction Only. The **Friction-Only** agent is highly effective at controlling the spread of the cascade. As seen in the left and centre panels of Figure 3, its risk trajectory is significantly suppressed, and it achieves a short Time-to-Clear (13 steps), nearly as effective as the full method (11 steps). However, its weakness is exposed in the right panel; without the 'balance' lever, it is unable to manage the emotional dynamics of the user population, resulting in a very high **Final Polarization Score** (0.75). This demonstrates that friction is the primary mechanism for containing the spread but is insufficient for mitigating emotional harm.

5.3.2 Balance Only. Conversely, the **Balance-Only** agent excels where the Friction-Only agent fails. As shown in the right panel, it is highly effective at reducing polarization, achieving a low score of 0.30. This confirms that the 'balance' lever is the key mechanism for promoting emotional stability. However, the left and centre panels show that without the 'friction' lever to act as a brake, the Balance-Only agent struggles to contain the cascade's spread, resulting in a higher peak risk and a longer Time-to-Clear (15 steps) compared to friction-enabled methods.

In conclusion, the ablation study confirms that the two levers are not redundant but complementary. The superior performance of the full method is achieved through the synergy of 'friction' actively suppressing the spread and 'balance' concurrently healing the emotional divide.

5.4 Further Discussion

Interpretability and Auditing. Our controls are *auditable by construction*: the learned topic-conditioned IC kernel and user parameters remain fixed; policies act only on the flow layer via two bounded scalars per shard—*friction* and *balance*—which are linearly mapped to users and realised through probability scaling and cooldowns. We release human-auditable telemetry (action traces, budget utilisation, cooldown hit rate, negative-exposure curves with peak and exceedance overlays, and JS divergence) and ablations (friction-only / balance-only), enabling external reviewers to trace when, where, and how strongly throttling or corrective mixing was applied without inspecting content.

Ethics and Limitations. The framework regulates *exposure flow* (timing and candidate re-weighting) only: it does not alter content, does not remove accounts, and does not feed topic/valence labels into per-user action selection; these signals are used solely for modelling and evaluation. Risk threshold and soft-peak temperature are pre-registered on validation and reported transparently. Limitations include reliance on a simplified diffusion simulator, possible topic/valence labelling noise, normative choices in cooldowns and budgets, and potential heterogeneous impacts across communities; we therefore report shard/topic breakdowns and sensitivity analyses, and recommend any deployment proceed via small, reversible trials with appropriate transparency and independent oversight.

6 CONCLUSION

We framed platform intervention as *flow-layer* control over a learned, topic-conditioned IC environment: policies never alter content or accounts, but regulate *exposure timing and selection* via two interpretable levers—*friction* and *balance*. Using CTDE–MAPPO with shard-level decisions and user-level simulation, we optimise risk-aware temporal objectives (smooth peak, time/area above a fixed threshold) under an engagement floor, per-step budgets, and smoothness constraints. Across two public Twitter corpora (COVID-19, mpox), the approach consistently reduces transient negative-exposure risk and polarisation metrics at comparable engagement, and ablations confirm complementary roles for friction (slowing/spacing) and balance (corrective/diverse counter-exposure drawn from existing content).

Our design is auditable by construction: the diffusion kernel and user parameters remain fixed; telemetry exposes action traces, budgets, cooldown effects, and outcome curves for external scrutiny. Limitations include reliance on a simplified diffusion simulator and potential label noise; future work will couple flow controls with adaptive thresholding, richer user adaptation models, and pre-registered, reversible field trials to assess external validity and long-run dynamics.

REFERENCES

[1] Eitan Altman. 1999. *Constrained Markov Decision Processes*. Chapman & Hall/CRC.

[2] Douglas Bates, Martin Mächler, Benjamin Bolker, and Steve Walker. 2015. Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software* 67, 1 (2015), 1–48. https://doi.org/10.18637/jss.v067.i01

[3] Ricardo J. G. B. Campello, Davoud Moulavi, Arthur Zimek, and Jörg Sander. 2015. Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection. *ACM Transactions on Knowledge Discovery from Data* 10, 1 (2015), 5:1–5:51. https://doi.org/10.1145/2733381

[4] Wei Chen, Chi Wang, and Yajun Wang. 2010. Scalable Influence Maximization for Prevalent Viral Marketing in Large-scale Social Networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1029–1038. https://doi.org/10.1145/1835804.1835948

[5] Meijie Chu, Wentao Song, Zeyu Zhao, Tianmu Chen, and Yi-chen Chiang. 2024. Emotional contagion on social media and the simulation of intervention strategies after a disaster event: a modeling study. *Humanities and Social Sciences Communications* 11 (2024), 968. https://doi.org/10.1057/s41599-024-03397-4

[6] Qin Deng, Xiaoliang Chen, Peng Lu, Yajun Du, and Xianyong Li. 2025. Intervening in Negative Emotion Contagion on Social Networks Using Reinforcement Learning. *IEEE Transactions on Computational Social Systems* 12:6 (2025), 4469 – 4480. https://doi.org/10.1109/TCSS.2025.3555607 Early Access.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*. Association for Computational Linguistics, Minneapolis, USA, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[8] Bradley Efron and Robert J. Tibshirani. 1994. *An Introduction to the Bootstrap*. Chapman & Hall/CRC.

[9] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual Multi-Agent Policy Gradients. In *AAAI*, Vol. 32:1. 11794.

[10] Amit Goyal, Francesco Bonchi, and Laks V. S. Lakshmanan. 2010. Learning Influence Probabilities in Social Networks. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM)*. ACM, 241–250. https://doi.org/10.1145/1718487.1718518

[11] Maarten Grootendorst. 2022. BERTopic: Neural Topic Modeling with a Class-based TF–IDF Procedure. https://arxiv.org/abs/2203.05794

[12] C. J. Hutto and Eric Gilbert. 2014. VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media (ICWSM)*. AAAI Press, Ann Arbor, USA, 216–225. https://ojs.aaai.org/index.php/ICWSM/article/view/14550

[13] Matt J. Keeling and Pejman Rohani. 2008. *Modeling Infectious Diseases in Humans and Animals*. Princeton University Press. https://doi.org/10.1515/9781400841035

[14] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the Spread of Influence through a Social Network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 137–146. https://doi.org/10.1145/956750.956769

[15] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In *NeurIPS*. 12.

[16] Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software* 2, 11 (2017), 205. https://doi.org/10.21105/joss.00205

[17] Frans A. Oliehoek and Christopher Amato. 2016. *A Concise Introduction to Decentralized POMDPs*. Springer. https://doi.org/10.1007/978-3-319-28678-5

[18] Alex Ray, Joshua Achiam, and Dario Amodei. 2019. Benchmarking Safe Exploration in Deep Reinforcement Learning. arXiv:1910.01708 [cs.LG] https://arxiv.org/abs/1910.01708

[19] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP) and the 9th International Joint Conference on Natural Language Processing (IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 3982–3992. https://doi.org/10.18653/v1/D19-1410

[20] Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the Space of Topic Coherence Measures. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining (WSDM)*. ACM, Shanghai, China, 399–408. https://doi.org/10.1145/2684822.2685324

[21] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015. High-dimensional continuous control using generalized advantage estimation.

[22] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347 [cs.LG] https://arxiv.org/abs/1707.06347

[23] Ruidong Yan, Deying Li, Weili Wu, Ding-Zhu Du, and Yongcai Wang. 2020. Minimizing Influence of Rumors by Blockers on Social Networks: Algorithms and Analysis. *IEEE Transactions on Network Science and Engineering* 7, 3 (2020), 1067–1078. https://doi.org/10.1109/TNSE.2019.2903272

[24] S. Yang, Q. Du, G. Zhu, J. Cao, L. Chen, W. Qin, and Y. Wang. 2024. Balanced influence maximization in social networks based on deep reinforcement learning. *Neural Networks* 169 (2024), 334–351. https://doi.org/10.1016/j.neunet.2023.10.030

[25] Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre Bayen, and Yuxiao Wu. 2021. The Surprising Effectiveness of MAPPO in Cooperative Multi-Agent Games. arXiv:2103.01955 [cs.LG] https://arxiv.org/abs/2103.01955

[26] Bianca Zadrozny and Charles Elkan. 2002. Transforming Classifier Scores into Accurate Multiclass Probability Estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 694–699. https://doi.org/10.1145/775047.775151