Supplementary Information for "Dynamics of collective mind in online news communities"

April 10, 2025

Supplementary Notes

List of Figures

List of Tables

1 Computational model

Here, we provide an analytic description of the computational model of the collective mind dynamics of the online news community proposed in the main manuscript. We first initialized the general and community semantic network and iterated the frequency and weight update T times to get the simulated result of the model. The following description is aimed at formulating a general framework of our computational model, thus, all of the specific model settings, hyperparameters, and functional forms of the distributions used in our main study are explicitly specified in the table S1 for the reader's convenience. Note that most of the model settings that we employed are chosen to reflect the empirical findings from our data, and one can freely alter settings of our framework according to one's data at hand.

1.1 Semantic network definition

At any given time t, the **general semantic network** from the current time t is expressed as $G_t^g = (V^g, E^g, F_t^g, W_t^g)$, where V^g and E^g denotes the set of vertices and edges, respectively. Here, we assume the topic vertices and edges between them are persistent through time, and there are a total of $|V^g| = N$ vertices (topics) and $|E^g| = N(N-1)/2$ edges since the network is complete without self-loops. Each vertex $v_i^g \in V^g$ indicates a single topic and has a **normalized frequency value** $f_{i,t}^g \in F_t^g$ where F_t^g is the set of all normalized frequency values at time t and $\sum_i f_{i,t}^g = 1$. From this normalized frequency, we can define a **normalized frequency ranking** $r_{i,t}^g \in R_{i,t}^g$ where $r_{i,t}^g =$

$\text{rank}(f^g_{i,t})/N$ and $\text{rank}(f^g_{i,t})$ denotes a ranking of $f^g_{i,t}$ among F^g_t. Note that by definition, r_i lies in between 0 and 1. Each edge $e^g_{ij,t} \in E^g_t$ indicates the semantic closeness between two topics, and has a **weight value** $w^g_{ij,t} \in W^g_t$ where $0 \leq w^g_{ij,t} \leq 1$ and W^g_t is the set of all weight values at time t. Also, we consider K different community with respective **community semantic network** $G^k_t = (V^k, E^k, F^k_t, W^k_t)$ at time t, which shares vertices and edges ($V^k = V^g, E^k = E^g$) but with (potentially) different values for F^k_t (R^k_t) and W^k_t. Hence, we drop the superscript for $V(v)$ and $E(e)$ from here for simplicity.

1.2 Semantic network initialization

Without loss of generality, we set the general semantic network's initial ranking order to follow the indices, i.e., topic 1 is the first most frequent, topic 2 is the second most frequent, and so on ($r^g_{i,0} = i$). We achieve this ranking by setting initial frequencies $f^g_{i,0} = F_f(i)$, where F_f is a monotonically decreasing **initial frequency distribution**. In our study, we chose $F_f \propto r_i^{-1}$, which leads to $f^g_{i,0} = {r^g_{i,0}}^{-1}/C = i^{-1}/C$ where $C = \sum_1^N i^{-1}$. Note that we preserve this initial distribution after the updating (See 1.6), so the while the ranking of each topic changes with respect to the frequency at the given time ($f^g_{i,t} = {r^g_{i,t}}^{-1}/C$), the frequency distribution remains the same. We also sample the weights $w^g_{ij,0}$ from the **initial weight distribution** F_w, finishing the initialization of the general semantic network. We employed the log-normal distribution for the initial weight distribution, $F_w \propto e^{-\ln^2(\frac{x-a}{b})/2s^2}$, where a, b, and s are the parameters that control the distribution's shape.

We further initialize the community semantic network depending on the initial settings. For each community, we first copy the frequencies and weights from the general semantic network and perturb them by adding noise. In our study, we used $f^k_{i,0} = f^g_{i,0} + \mathcal{N}(0, \sigma_{\text{fp}}) \times F_f(i)$ to ensure the noise scale matters for all frequency ranges, where σ_{fp} denotes the standard deviation for the frequency perturbation. Similarly, $w^k_{ij,0} = w^g_{ij,0} + \mathcal{N}(0, \sigma_{\text{wp}})$, where σ_{wp} denotes the standard deviation for the weight perturbation. We used $\sigma_{\text{fp}} = 0$ and $\sigma_{\text{wp}} = 0$ in most of the cases, which assumes the community semantic network is identical to the general semantic network at the beginning (equilibrium state). For the simulation in a non-equilibrium state (e.g., Alignment (Fig. 4a) and Membership turnover (Fig. 5a) scenario), we used $\sigma_{\text{fp}} = 1.0$ and $\sigma_{\text{wp}} = 0.05$.

1.3 Events generation

At each time step t, the general semantic network G^g_t generates a new set of **events** $X_t = \{x_{1,t}, x_{2,t}, \cdots, x_{N_x,t}\}$ for the current timestep, where N_x denotes the number of events per each time step. Each event is consists of N_w number of topics, $x_{a,t} = \{v_{i,t,1}, v_{i,t,2}, \cdots, v_{i,t,N_w}\} = \{v_{z_1}, v_{z_2}, \cdots, v_{z_{N_w}}\}$, where the z_q denotes q-th tier topic of the event. This definition implies that the q-th tier topic of the i-th event at time t ($v_{i,t,q}$) is a topic numbered as z_q (v_{z_q}). For this work, we choose $N_w = 3$, so the event is described as a triplet of topics. For each

event, we sample topics for each tier q with a probability proportional to the $v_{i,t,q} \sim F_{ns}(r^g_{z_q,t})$, where the $F_{\rm ns}$ denotes the **event sampling distribution**. In our study, we chose $F_{\rm ns} \propto -\ln(r^g_{i,t})$, independent of tier and a monotonically decreasing function of the ranking $r^g_{i,t}$. Also, we ensured the $v_{i,t,q}$ are unique for each tier by sequentially sampling each tier while excluding all the previous tier's topics and renormalizing $F_{\rm ns}$ accordingly.

If one needs to ensure the time correlation of the event topics, one can consider the previous event topics as a prior for the current event topics. This can be achieved by first sampling the new title topic frequency distribution from the given distribution $\hat{F}_{ns}(r^g_{i,t})$ every time step, and creating a set of events by choosing topics from linearly interpolated distribution $F_{\rm ns}(r^g_{i,t}) = (1-\lambda_e)\hat{F}_{\rm ns}(r^g_{i,t}) + \lambda_e F_{\rm ns}(r^g_{i,t-1})$. To ensure the uniqueness of the topic in the events, we employed rejection sampling, where we repeated the sampling till there were no events with the duplicate topic in the events set. In this study, we adopted this setting with the event memory strength $\lambda_e = 0.5$.

1.4 Filter definition

After the event generation, each event $x_{j,t}$ passes through a filter of each community and is determined whether it will be filtered or not and posted as news, i.e., the filtered event becomes news. We first specify the editors' criteria for the filtering, which is determined by their view on both general and community semantic networks, defined as follows.

$$\bar{f}^k_{i,t} = \lambda_f f^k_{i,t} + (1-\lambda_f) f^g_{i,t} \tag{1}$$

$$\bar{w}^k_{ij,t} = \lambda_f w^k_{ij,t} + (1-\lambda_f) w^g_{ij,t} \tag{2}$$

Here, both the view on normalized frequency rank $\bar{r}^k_{i,t}$ (which is derived from $\bar{f}^k_{i,t}$) and weight $\bar{w}^k_{i,t}$ are controlled by the filter strength $0 \le \lambda_f \le 1$. The filter strength λ_f serves as a role of linear interpolation parameter between general and community semantic networks and determines whether the editors' criteria are more inclined to the outside world or their community.

With these, the filter of each community consists of a two-stage sampling process; one considering the frequency of topics ($\bar{f}^k_{i,t}$), and another considering the similarity between topics ($\bar{w}^k_{i,t}$). The filtering ratio $0 \le R_1, R_2 \le 1$ determines how much of the events will survive for the first and second filtering, respectively. First, we calculate the product of exponentiated frequencies of topics as $\prod_q (\bar{r}^k_{z_q,t})^{\alpha_q}$, where α_q denotes q-th tier **filter exponent**, and normalize them as a probability for each event. We then sample R_1 of the events (without replacement) according to this probability. With this filtered events, we further calculate the product of similarities between topics as $\prod_{q_1,q_2} \bar{w}^k_{z_{q_1} z_{q_2},t}$ for each event $x_{i,t} = \{v_{z_1}, v_{z_2}, \cdots, v_{z_{N_w}}\}$, and we keep only the top R_2 of the filtered event by sorting them based on this sum. Finally, we keep a total of $R_1 R_2 N_x = \bar{N}_x$ events that pass both filters, and the rest of the events are filtered out. We denote filtered events as $X^k_t = \{x^k_{1,t}, x^k_{2',t}, \cdots, x^k_{\bar{N}_x,t}\} \subseteq X_t$ for

each community k at time t. This process is equivalent to considering both the perceived importance (frequency) and inter-topic similarity (weight) of the topics in the event, in order to decide whether the editors accept it as news in their community or not.

In practice, if we want to calibrate the model with α_q, we apply α_q/R_2 at the filtering stage. This is because the exponent gets decreased due to the second stage of the filtering, which is effectively random (since there is no correlation between weight and frequency in the beginning). Intuitively, random sampling reduces the steepness of the original distribution, which is equivalent to scaling down the exponent. Strictly speaking, the assumption of the non-correlation between the weight and frequency is not always true, as the correlation slowly builds up as the model evolves because the inter-topic weight(similarity) increases as the co-occurrence between two topics happens, and the topic with higher frequency generally has more chance to get this. However, we found that this effect is negligible in practice, and the calibration with α_q/R_2 is sufficient to capture the overall behavior of the model, especially in the early stage.

1.5 Comment semantic network generation

The filtered events (news) will elicit responses from the collective mind of the community as a form of comments. Based on empirical evidence, we make two model assumptions. First, the frequency of comments that match the subject of the news, which we'll call on-topic comments, increases. Also, the appearance of the specific topic pair in the news increases the weight between those topics. Combining these two, we define the **comment network** for community k at time t as $A_t^k = (V, E, \hat{F}_t^k, \hat{W}_t^k)$, which shares vertices and edges with other semantic networks, but with comment frequency $\hat{f}_{i,t}^k \in \hat{F}_{i,t}^k$ and comment weight $\hat{w}_{ij,t}^k \in \hat{W}_{ij,t}^k$.

First, we need to construct the **comment frequency**, which is a direct sum of all comment frequency distributions under the news. For given news $x_{i,t}^k = \{v_{z_1}, v_{z_2}, \cdots, v_{z_{N_w}}\}$ (where the q-th tier topic is z_q), we first assign the relative number of comments under this news by sampling from a comment number distribution, $c_{i,t}^k \sim P_c(c, x_{i,t}^k)$ (In our implementation, we used topic-independent sampler, hence $P_c(c, x_{i,t}^k) = P_c(c)$). We then determine whether the comment multiplier would be zero or non-zero (for each tier) by sampling a uniform random number from 0 to 1 and comparing it to the tier-wise zero ratio, $Z_q(r)$, and setting it to zero if it is smaller than the ratio. If the value is higher and comment multiplier is determined to be a non-zero value, now we sample tier-wise comment multipliers from a (non-zero) tier-wise **comment multiplier distribution**, $m_{i,t,q}^k \sim P_{m,q}(m, r_{z_q}^k, c_{i,t}^k)$. More precisely, the multiplier distribution is a function of the tier q itself (denoted in the subscript), community topic ranking for each tier $r_{z_q}^k$, and the comment number $c_{i,t}^k$. From these comment multipliers, we get the comment frequency distribution under the news $x_{i,t}^k$ as follows.

$$\hat{f}^k_{j,t}(x^k_{i,t}) = \begin{cases} c^k_{i,t} m^k_{i,t,q} f^k_{j,t} & \text{if } v_j = v_{z_q} \\ c^k_{i,t} f^k_{j,t} / C^k_{i,t} & \text{if } v_j \notin x^k_{i,t} \end{cases} \tag{3}$$

$$C^k_{i,t} = \frac{1 - \min(\sum_q m^k_{i,t,q} f^k_{q,t}, 1)}{1 - \sum_q f^k_{q,t}} \tag{4}$$

Here, $C^k_{i,t}$ is the normalization constant for off-topic comments to keep the assigned comment number, and the subscripts i and j denote the i-th news and j-th topic, respectively. Basically, this means that we would like to multiply the frequency of the q-th tier on-topic comments by m_q, and the rest of the assigned comments simply follow the previous community frequency distribution. Note that this implementation sometimes results in the sum of the frequency of the comments being more than the assigned number of comments. We find that this exception happens rarely in practice (less than 2%), hence it does not affect the overall comment number distribution. The overall comment frequency at time t then becomes the sum of all comments frequency distribution under the news, $\hat{f}^k_{j,t} = \sum_i \hat{f}^k_{j,t}(x^k_{i,t})$.

Referencing the observation from the empirical data (See Fig. S1 and Supplementary Note 2), we implemented the comment multiplier distribution by splitting the distribution into two parts: one with zero multiplier and one without. For the zero case, we assign the zero multiplier ratio $Z_q(r) = C_{z,q} r$ for each tier q, which is a linear function of normalized comment frequency ranking r and denotes the probability that the multiplier of interest is zero. We perform the *zero-check* by using a Bernoulli trial with $p = Z_q(r)$. If it passes this zero check, we sample the multiplier from the non-zero distribution $P^{nz}_{m,q}$, which is a function of the tier q itself, the community topic ranking for each tier $r^k_{z_q}$, and the comment number $c^k_{i,t}$.

Now, we need to construct the **comment weight**. For each news $x^k_{i,t} = \{v_{z_1}, v_{z_2}, \cdots, v_{z_{N_w}}\}$, we first define a set of co-occurring pairs between news topics S and assign comment weights as follows,

$$S^k_{i,t} = \{\, (a,b) \mid v_a \in x^k_{i,t} \wedge v_b \in x^k_{i,t} \,\}, \tag{5}$$

$$\hat{w}^k_{ab,t}(x^k_{i,t}) = \begin{cases} c^k_{i,t} & \text{if } (a,b) \in S^k_{i,t} \\ 0 & \text{if } (a,b) \notin S^k_{i,t}, \end{cases} \tag{6}$$

where $c^k_{i,t}$ is the assigned comment number for the news $x^k_{i,t}$. This setting implies that the co-occurring topic pairs in the news with many comments will have a high impact on the increase of inter-topic similarity. Similar to the comment frequency, the overall comment weight at time t becomes the sum of all comment weight under the news, $\hat{w}^k_{ij,t} = \sum_i \hat{w}^k_{ij,t}(x^k_{i,t})$. Note that both comment frequency and weight are not properly normalized at this point, and we will normalize them at the update step.

1.6 Community semantic network update

From the comment semantic network, we finally update the community semantic network to complete the feedback loop. For the frequency, we adopt a **memory strength** $0 \leq \lambda \leq 1$ to keep the previous frequency distribution and update the frequency as follows. First, we construct a proxy frequency distribution for this time step as

$$\hat{f}^k_{i,t+1} = \lambda f^k_{i,t} + (1-\lambda)\hat{f}^k_{i,t} / \sum_j \hat{f}^k_{j,t}. \tag{7}$$

With this proxy frequency, we update the frequency by first computing the rank according to the proxy frequency and assigning the frequency of that rank, $f^k_{i,t+1} = F_f(\mathrm{rank}(\hat{f}^k_{i,t+1})/N)$. This effectively quantizes the possible frequency and ensures the initial frequency distribution F_f is preserved after the update. Note that this only enforces the distribution of the frequency (unobservable in real data), not the comment frequency (observable in real data), which is a result of an additional sampling process.

For the weight, we employed a Hebbian learning scheme[1] for the update. For each pair of topics v_i and v_j in the community semantic network, we update the weight as follows.

$$w^k_{ij,t+1} = \eta(w_{\max} - |w^k_{ij,t}|)\hat{w}^k_{ab,t}/D^k_t - \gamma w^k_{ij,t} + \epsilon_{ij} \tag{8}$$

Here, η is a learning rate, $w_{\max}$ is a maximum cap for a weight value, γ is a decaying rate, and ϵ_{ij} is a Gaussian noise with $\mathcal{N}(0, \sigma^2_{wn})$. Again, the weight is normalized by $D^k_t = \frac{N_w(N_w-1)}{2}\sum_i c^k_{i,t}$ before the update, which considers the number of total comments and the possible number of topic pairs based on the event N_w. Also, to ensure stability, we used an adaptive decaying rate $\gamma(w^k_{ij,t}, \hat{w}^k_{ij,t})$ as

$$\gamma(w^k_{ij,t}, \hat{w}^k_{ij,t}) = \eta \frac{\sum_{i,j}(w_{\max} - |w^k_{ij,t}|)\hat{w}^k_{ab,t}/D^k_t + \epsilon_{ij}}{\sum_{ij} w^k_{ij,t}}, \tag{9}$$

which normalizes the decaying rate by the relative scale of the Hebbian learning term.

With these updated frequencies and weights of each community semantic network, the full iteration is ended, and we repeat this process T times to get the simulated result of the model.

We mainly calibrated and chose the functional form of our model from the counterpart in the empirical data, except for some notable cases. For the semantic network, since it's not directly observable, we used the distribution from the comment network in the empirical data (See Supplementary Note 7 for discussion). Filtering ratios $(0.5, 0.5)$ are arbitrarily but feasibly chosen, and can be easily modified if one has prior knowledge of the filtering behavior of the community (for instance, the survival rate of the initial draft). For the learning rate and weight noise s.d., we chose the parameters to ensure the stability of the model.

Table S1: Computational model implementation. The normalization constants are omitted for simplicity. Items with * indicate that the parameters are selectively used depending on the specific scenario. $\bar{H}(n)$ denotes $nH(n)$, where $H(n) = \sum_i (1/n)$.

Process	Components	Functional form	Constants
Network initialization	Initial frequency dist. (F_f)	$F_f(i) \propto r_i^{-\alpha_c}$	$\alpha_c = 1.0$
	Initial weight dist. (F_w)	$F_w(w) \propto e^{-\ln^2(\frac{w-a}{b})/2s^2}$	$a = -0.65$, $b = 1.0$, $s = 0.12$
	Frequency perturbation s.d. (σ_{fp})	Const.	$\sigma_{\text{fp}} = 0.0, 1.0^*$
	Weight perturbation s.d. (σ_{wp})	Const.	$\sigma_{\text{wp}} = 0.0, 0.05^*$
Events generation	Event sampling dist. (F_{ns})	$F_{\text{ns}}(r_{i,t}^g) \propto -\ln(r_{i,t}^g)$	-
	Event memory strength (λ_e)	Const.	$\lambda_e = 0.5$
Filter definition	Filtering ratio (R_1, R_2)	Const.	$R_1 = 0.5$, $R_2 = 0.5$
	Filter exponent (α_q)	Const.	$\alpha_1 = 0.4$, $\alpha_1 = 0.2$, $\alpha_3 = 0.1$
Response generation	Comment number dist. (P_c)	$P_c(c, x_{i,t}^k) \propto e^{-\ln^2(\frac{c-a}{b})/2s^2}$	$a = 5.7 \times 10^{-6}$, $b = 1.0 \times 10^{-4}$, $s = 1.5$
	Zero multiplier ratio (Z_q)	$Z_q(r) = C_{z,q} r$	$C_{z,1} = 0.7$, $C_{z,2} = 0.9$, $C_{z,3} = 0.9$
	Non-zero distribution ($P_{m,q}$)	$P_{m,q}(m, r_{z_q}^k, c_{i,t}^k) \propto e^{-\lambda_q(r_{z_q}^k)m}$, $m \in [a, b]$	$a = C_{\text{com}}\bar{H}(r_{z_q}^k)/c_{i,t}^k$, $b = \bar{H}(r_{z_q}^k)$, $C_{\text{com}} = 1.0 \times 10^{-6}$
	Non-zero exponent (λ_q)	$\lambda_q(r_{z_q}^k) = a_q e^{-b(r_{z_q}^k)}$	$a_1 = 0.005$, $a_2 = 0.01$, $a_3 = 0.02$, $b = 0.8$
Network update	Learning rate (η)	Const.	$\eta = 10.0$
	Maximum weight ($w_{\max}$)	Const.	$w_{\max} = 0.8$
	Weight noise s.d. (σ_{wn})	Const.	$\sigma_{\text{wn}} = 0.001$

2 Analysis on empirical findings and verification of model assumptions

Here, we provide more analysis on statistical features in our data that were used to initialize our model, and empirical evidence to support some of the implicit model assumptions in the proposed computational model.

2.1 Title frequency distribution modeling

We observed that the title topic frequency follows an interesting distribution, a product of negative log and power-law distribution with tier-specific exponent (Fig. 2 in the main manuscript). Considering that this title topic distribution corresponds to the title topic distribution of the filtered events in our computational model, the distribution should come from the combined effect of both event generation and the filtering process. The event generation and the filtering process are independent in our model, so the most natural assumption is that each process is responsible for one of the two distributions (although a more complex division is not impossible).

While either combination is mathematically plausible, and both are monotonically decreasing functions with a heavy tail, we chose the negative log distribution for the event generation process and the power-law distribution for

Table S2: Additional fitting parameters for the empirical data (Global TM)

Name	# of comment		
	$a(\times 10^{-5})$	$b(\times 10^{-4})$	s
Mother Jones	26.6973	29.9659	1.7790
Atlantic	12.3054	14.4338	2.0085
The Hill	0.5928	1.0383	2.1505
Breitbart	0.6471	1.0575	2.2161
Gateway Pundit	-1.8909	8.8054	0.7850

the filtering process in this study for the following reasons. We find that the exponents of the power-law distribution for each tier are empirically different for each community (Fig. 2), while the log distribution is universal across communities. This suggests that the filtering process, which is a community-specific process, is more likely to be responsible for the power-law distribution, while the negative log part is more likely to be accountable for the negative log part. Also, note that the choice of filtering process as a power-law implicitly assumes that this process heavily emphasizes the high-frequency topics and is responsible for the extremely high frequency of popular topics (see Fig. 2b, where the differences in exponents are only meaningful for the popular topics), which is a reasonable assumption considering the nature of the filtering process.

2.2 Number of comment distribution

Each article in the online news communities has a different number of comments, and the distribution of the number of comments can be an important factor in understanding the dynamics of the collective mind, especially considering that our computational model explicitly samples the number of comments to simulate the comment distribution (by multiplying the sampled number of comments to the normalized topic distribution). In this work, we introduce the concept of the relative number of comments, which is the number of comments divided by the total number of comments in the given period (in this case, we chose 1 month). With this, we can construct the distribution of the number of comments without dealing with the volumetric change of the comment through time. We show the distribution of the relative number of comments for each community (Extended Data Fig. 2a), which nicely fits the log-normal distribution. The fitting parameters for the empirical data are summarized in Table S2, and we used $a = 5.7 \times 10^{-6}$, $b = 1.0 \times 10^{-4}$, and $s = 1.5$ for the model simulation.

2.3 Comment multiplier distribution

In our computational model, we use the concept of comment multiplier to describe the behavior of the comment topic distribution under certain news articles. From the time series of comment topic distribution, we can calculate the comment multiplier for each topic, which is defined as the ratio of the comment topic frequency under the news article to the expected (previous) comment

topic frequency. For stability, we use 12-month average topic distribution as the expected frequency.

First, we find that a considerable amount of comment multiplier is zero, which indicates that no comment corresponds to the title topic, and the frequency of zero increases as the topic rank gets larger (i.e., less frequent topics). We show the zero multiplier ratio ($Z_q(r)$) for each community and tier in the Extended Data Fig. 2b. We used the linear approximation for the model calibration for simplification, although a more complex fitting function can be used. In our model, we used $Z_1 = 0.7r$, $Z_2 = 0.9r$ and $Z_3 = 0.9r$ where the r is the normalized comment topic rank.

For the non-zero multipliers, we show that it follows the exponential distribution, with different decay rates λ for different topic ranks and tiers (Extended Data Fig. 2b, c). We further show that the comment multiplier in the specific article has both the theoretical upper and lower bound (Extended Data Fig. 2e). Let $m_{n,q}$ be the comment multiplier for topic n with tier q. Due to the power-law distribution of the comment topic frequency, the expected frequency for topic n is proportional to $n^{-\alpha_c}$. In case of $\alpha_c = 1$, the normalization constant becomes the harmonic series $H(n) = \sum_{x=1}^{n} \frac{1}{x} = \ln(n) + \gamma + \frac{1}{2}n^{-1} - \frac{1}{12}n^{-2} + \mathcal{O}(n^{-3})$ where $\gamma = 0.5772...$ is the Euler–Mascheroni constant. Considering that the expected frequency of topic n is $(1/n)/H(n)$, the (expected) maximum multiplier for topic n is the reciprocal of this, $B_{\max} = nH(n)$ (note that this is the case where all of the comment under that article is topic n). Conversely, since the number of comments is a natural number, the minimum multiplier happens when there is exactly 1 comment with topic n (since 0 comments would yield the zero multiplier, which we handled separately). So, if we know the number of comments c under the article in interest, we can simply calculate the minimum multiplier as $B_{\min}(c) = \frac{1}{c}/\frac{1}{nH(n)} = \frac{1}{c}nH(n)$. Here, we have two problems: (1) the number of comments c is different for each article, and (2) we would like to express this with the *relative* number of comments, x. The first issue can be handled by relaxing the boundary to *expected* minimum boundary, using the *expected* number of comments $\mathbf{E}(c)$ by averaging over all news (at a given time interval, 1 month in this case). $\mathbf{E}(B_{\min}(c)) = \frac{1}{\mathbf{E}(c)}nH(n)$.

Now, we can resolve the second issue by first expressing x with c as

$$x_i = \frac{c_i}{\sum_i c_i}, \tag{10}$$

where x_i and c_i is the i-th article's relative and raw number of comments. The expected number of comments is $\mathbf{E}(c) = \sum_i c_i/\bar{N_x}$ where the $\bar{N_x}$ is the number of news articles (notation is aligned with the computational model, see Supplementary Note 7). Hence, to express this with expected *relative* number of comments $\mathbf{E}(x) = \sum_i x_i/\bar{N_x}$,

$$\mathbf{E}(c) = \sum_i c_i/\bar{N_x} = \sum_i \frac{x_i \sum_j c_j}{\bar{N_x}} = \sum_j c_j \mathbf{E}(x). \tag{11}$$

Hence, with C_{com} as the inverse of the total number of comments per month ($C_{\text{com}} = 1/\sum_j c_j$), we can express the expected minimum multiplier as

$$\mathbf{E}(B_{\min}) = \frac{1}{\mathbf{E}(c)} nH(n) = \frac{1}{\sum_j c_j \mathbf{E}(x)} nH(n) = \frac{C_{\text{com}}}{\mathbf{E}(x)} nH(n), \tag{12}$$

These two lines greatly match with the empirical maximum and minimum values in the Extended Data Fig. 2e, and we used $C_{com} = 1.0 \times 10^{-6}$ for the model calibration (which matches with the scale of a bigger community like The Hill and Breitbart, since it assumes the number of comments per month as 1.0×10^6).

All of these findings are reflected in the choice of the comment multiplier sampling distribution in the computational model (See Supplementary Table 8).

2.4 Off-topic frequency distribution follows previous community frequency distribution

We modeled the response of the community to the news by assuming that the off-topic frequency distribution is the same as the previous community semantic network's frequency distribution, which follows a power-law distribution with the exponent of -1 (Extended Data Fig. 1). In Fig. S2a, we show the relative comment topic frequency distribution for off-topic comments from the online news communities by removing all of the on-topic comments under the news in the data aggregation stage. Considering its fitted power-law exponents α_c (see captions), we can confirm that this off-topic distribution is also roughly a power-law distribution with the exponent of -1, which supports our model assumption. Note that a more detailed investigation by comparing the off-topic frequency distribution at time t with the previous community frequency distribution at time $t-1$ is also possible.

2.5 Similarity decays without cooccurrence

In our model, we adopted the updated scheme similar to the Hebbian learning for the topic similarity dynamics. This is based on two assumptions: one is that the similarity between two topics increases with the co-occurrence in the news, and the other is that the similarity decays without the co-occurrence. We verified the latter assumption by calculating the difference in topic similarity as a function of the time absent from the news title in the online news communities (Fig. S2b). Atlantic, Thehill, and Gatewaypundit show a relatively strong decaying trend. At the same time, Motherjones was relatively weak and Breitbart showed a positive trend (but it was only fitted from merely 6 datapoints since no topic pair once existed and did not appear for more than 6 months, which greatly reduces the fidelity of Breitbart case for this analysis). Overall, (considering the fact that the overall aggregated fitting line shows a strong negative slope), we can confirm that the similarity decays without the co-occurrence, which supports our model assumption.

2.6 Inter-topic similarity is enhanced by the cooccurrence in the news

We also verified the former assumption (inter-topic similarity increases with the co-occurrence) by comparing the similarity between on-topic and off-topic comments for the same topic pairs. We calculated the average cosine similarity between off-topic and on-topic comment embeddings for the same topic pairs and compared the similarity distribution for each community (Fig. S3). The rationale behind this comparison is that the on-topic comments under the news with certain topic pairs are more likely to be similar to each other since there is a much higher chance that the comment is talking about both topics or the relation of those on-topic at the same time, compared to the null-case off-topic comments.

Note that this averaged pair-wise similarity is not directly comparable to the similarity between the topic pairs used in the model, since the similarity in the model is calculated by first constructing the topic representation by averaging all of the embeddings first, and the cosine similarity is calculated from the averaged embeddings. The reason we used average pair-wise cosine similarity here is because of the systematic difference in the number of on-topic and off-topic comments, where on-topic comments for each topic pair is much smaller (sometimes three orders of magnitude) than the off-topic comments, hence the variance in on-topic cosine similarity gets too high. Still, this averaged pair-wise similarity can be used as a proxy to investigate the relative magnitude of similarities for this analysis.

In Fig. S3, we observe that the similarity between on-topic comments (blue) is generally higher than the off-topic comments (orange) for 3 online news communities, which indicates that the similarity between two topics is enhanced by the co-occurrence in the news. This supports our model assumption that the similarity between two topics increases with the co-occurrence in the news.

3 Empirical data preparation

Here, we provide a detailed description of the empirical data from online news communities used in the main manuscript.

We collected data from five online news communities, namely, Mother Jones (MJ), Atlantic (AT), The Hill (TH), Breitbart (BB), and Gateway Pundit (GP). The collected data consists of news articles (hereafter 'news') and comments on the respective websites within varying periods. We crawled the data using the `Disqus` API, which functioned as a common platform for commenting on various websites during the period. The data includes mainly the news title text and comment text along with the timestamp, but other metadata were also collected, such as the number of likes on comments and user ID (which are not used in this study).

We first preprocessed the whole data by applying several cleansing steps to the data. For the news title, we removed all the news that contains HTML

Table S3: Online news communities data summary after data cleansing.

Name	Inclination	Data period (months)	# of news (k)		# of comments (k)	
			Before	After (%)	Before	After (%)
Mother Jones	Far-left	12/06 ∼ 19/09 (87)	35.968	31.510 (87.61)	4783.86	4763.04 (99.56)
Atlantic	Left	12/06 ∼ 18/05 (71)	46.262	32.144 (69.48)	6736.16	6675.60 (99.10)
The Hill	Center	12/06 ∼ 22/03 (117)	380.62	313.67 (82.41)	176263.19	175989.96 (99.84)
Breitbart	Right	12/06 ∼ 23/04 (130)	591.04	400.03 (67.68)	205816.32	205280.91 (99.74)
Gateway Pundit	Far-right	15/01 ∼ 23/04 (99)	85.20	83.77 (98.32)	31279.42	31271.54 (99.97)

Table S4: Online news communities data summary after overdue / outlier filtering.

Name	# of news (k)		# of comments (k)		
	Before	After (%)	Before	After (%)	Non-outlier (%)
Mother Jones	31.510	23.92 (75.91)	4763.04	3707.93 (77.85)	2027.33 (42.56)
Atlantic	32.144	25.10 (78.10)	6675.60	6223.72 (93.23)	3030.48 (45.40)
The Hill	313.67	284.86 (90.81)	175989.96	172172.40 (97.83)	88812.39 (50.46)
Breitbart	400.03	360.94 (90.23)	205280.91	199875.94 (97.37)	103869.43 (50.60)
Gateway Pundit	83.77	79.49 (94.89)	31271.54	30439.70 (97.34)	15306.64 (48.95)

addresses (since these are typically not genuine news, but rather corrupted data or a duplicate of another news), and removed all news from further analysis that has equal to or fewer than $\theta_n = 10$ comments. For the comments, we removed all the HTML tags and consecutive spaces for further processing. The summary of the collected empirical data is provided in Table S3, and the time series of the number of news and comments before and after the filtering is shown in Fig. S4.

In this study, we used aggregated data for all analyses where data were pooled and added together over a given period. For the aggregated data, as mentioned in the main manuscript, we merged the news posted during 1-month intervals, and only the comments made within 7-days from the news post date were valid to be aggregated. During the process, we also removed news that is classified (in its top-3 classification) as an outlier (topic "-1") or contains less than θ_n non-outlier comments, to focus on a more meaningful (non-outlier) distribution. The rationale behind this removal is that articles that only have outlier comments (and less than θ_n non-outlier comments) have a high chance of only containing simple expressions and not significantly contributing to the landscape of the collective mind. Note that we did not remove all outlier comments at this stage, although most of the analysis in this study (unless specified) was done with non-outlier comments distribution. Finally, after both of the filterings (removing overdue comments and outliers) we further removed all news that had less than θ_n comments. The summary of the filtered data is provided in Table S4.

4 Topic modeling with BERTopic

In this work, we employed BERTopic[2] for the construction of topic models (TMs). With the given model settings (See the method section in the main manuscript), The construction consists of two steps: (1) the fitting phase, where we fit the model with sampled comments from the full data, and (2) the transforming phase, where the rest of the comments are classified based on the fitted model. We performed the following procedures to construct the global TM, which used data from all 5 communities combined, and also for the local TM, which used data from each community separately. Note that we mainly used the result from global TM (which is referred to as plain "topic model" in the main manuscript) for the analysis, and the local TM was used for the validation of the overall results.

4.1 Local topic model

For the fitting phase, we sample 2 million comments from each of the five communities using a variant of stratified sampling to better preserve the overall trend of comments without ignoring the influence of smaller news articles. Precisely, given the histogram of comment numbers, we choose the sampling threshold k' that matches the following condition,

$$k^* = \arg\max_{k^*} \left\{ k^* \mid \sum_{k=1}^{k^*} k \cdot X(k) + k^* \cdot \sum_{k=k^*}^{k_{\max}} X(k) < S \right\} \tag{13}$$

where $X(k)$ is a histogram of the number of news articles depending on the number of comments k, $k_{\max}$ is the maximum number of comments, and S is a sampling size (2 million). Simply, given a threshold k^*, we collect all of the comments from the news articles that have less than k^* comments and randomly sample k^* comments from the news articles that have more than k^* comments, so every news article has at most k^* sampled comments. The k^* values for each community are 80 for Mother Jones, 87 for Atlantic, 4 for The Hill, 6 for Breitbart, and 23 for Gateway Pundit. We repeated the sampling process to construct a 5 different set of sampled comments (by changing random seeds from 1 to 5) for later purposes.

With the sampled comments and their BERT embeddings, we ran the grid search on the hyperparameter space to find the optimal hyperparameters for the BERTopic model. The hyperparameters we tuned are the number of neighbors (neighbors, n) in UMAP, the minimum cluster size for HDBSCAN (cluster size, c), and the random seed for the fitting dataset (seed, s). We performed a two-stage grid search for each TM, where we first searched the coarse-grained hyperparameter space to find a local peak and then searched the fine-grained hyperparameter space around the optimal hyperparameters found in the first stage. Coarse-grained hyperparameter space is defined as follows: neighbors $\in \{30, 60, 90\}$ and cluster size $\in \{200, 300, 400\}$. If the optimal hyperparameters

Table S5: Outlier/0 ratio of TM hyperparameter grid search (coarse-grained), bold-faced values indicate the lowest ratio for each community.

Name	$c = 200$			$c = 300$			$c = 400$		
	$s = 30$	$s = 60$	$s = 90$	$s = 30$	$s = 60$	$s = 90$	$s = 30$	$s = 60$	$s = 90$
Global	0.7199	0.8976	0.8172	**0.6373**	0.7971	0.8875	0.8581	0.8888	0.7105
Mother Jones	0.8214	0.9905	0.9881	0.8073	0.8299	0.8807	0.8474	0.8341	**0.7123**
Atlantic	0.8876	0.8381	**0.7757**	0.8879	0.8056	0.8993	0.8887	0.9508	0.8836
The Hill	0.6954	0.6915	0.8501	0.6597	0.6758	**0.6116**	0.6883	0.7051	0.7588
Breitbart	**0.5291**	0.6151	0.5929	0.6153	0.6308	0.7253	0.7199	0.6720	0.6707
Gateway Pundit	0.8314	0.7894	0.8257	0.7850	0.8255	0.8313	**0.6875**	0.8221	0.7885

found in the first stage are called n_1 (neighbors) and c_1, respectively, the hyperparameter stage of the second stage is given by neighbors $\in \{n_1-10, n_1, n_1+10\}$ and cluster size $\in \{c_1 - 25, c_1, c_1 + 25\}$. For both stages, the random seed is chosen from $\{1, 2, 3, 4, 5\}$.

For the coarse-grained search, we chose the pair of hyperparameters (n_1, c_1) based on the "outlier/0 ratio", which is defined as a frequency ratio between the sum of topic -1 (outlier) and topic 0 (which we found to be quite typical and not very well separated in most of the cases) and rest of the comments. The smaller this ratio is, the better the model is, as it better represents the other topics other than outliers and topic 0. For each pair of hyperparameters, we averaged this value for the 5 seeds and chose the best pair of hyperparameters that minimize the mean outlier/0 ratio. Table S5 shows the coarse-grained search results for each community.

For the fine-grained search, we aim to find the local peak around the (n_1, c_1) as well as the best-performing seed. First, We chose top 5 triplets of hyperparameters (n_2, c_2, s) that minimize the outlier/0 ratio as intial candidates. We chose the final triplet among the candidates according to the following criteria: (1) First, we sorted them according to the DBCV [3] metric (2) Next, we chose 4 significant topics (Guns, Abortion, Vaccine, and Climate) and manually checked whether these topics are well-separated in the final candidate. If the model didn't separate these topics distinctly, we discarded them from the candidates. (3) Finally, from the remaining candidates, the hyperparameter triplet with the lowest DBCV metric was chosen to be the representative model for the community. The final hyperparameters for each community are summarized in Table S6.

4.2 Global topic model

For the global TM, we gathered locally sampled comments from 5 communities (which share the random seed) and further sampled 0.4 million comments each by using the same random seed, constructing 5 sets of 2 million sampled comments (as same as the local case). The rest of the procedures are the same as the local TM construction, and both the coarse-grained and fine-grained search results are summarized in Tables S1 and S2.

Table S6: Outlier/0 ratio of TM hyperparameter grid search (coarse-grained). The DBCV rank is calculated among the top 5 candidates. Cases, where the DBCV rank is not 1st, indicate that the higher rank models were discarded due to the topic separation check.

Name	coarse-grained result (n_1, c_1)	fine-grained result (n_2, c_2, s)	DBCV (rank)
Global	$(300, 30)$	$(325, 20, 1)$	0.2901 (1st)
Mother Jones	$(400, 90)$	$(425, 90, 5)$	0.3918 (1st)
Atlantic	$(200, 90)$	$(200, 80, 4)$	0.1958 (2nd)
The Hill	$(300, 90)$	$(300, 80, 2)$	0.2695 (1st)
Breitbart	$(200, 30)$	$(225, 20, 3)$	0.2496 (1st)
Gateway Pundit	$(400, 30)$	$(400, 30, 4)$	0.2305 (3rd)

5 Survey results for the topic model quality assessment

To validate the quality of (both global and local) topic models constructed by BERTopic, we conducted a survey using the social experiment platform, `Prolific` [4]. The survey consists of the following 6 tasks with a total of $1,028$ participants, which are representative of the U.S. public. Note that the descriptions for a topic are given by a set of top-4 representative keywords, chosen by the topic model.

1. T1: Word intrusion: test whether a model-generated topic has human-identifiable semantic coherence. Subjects must identify a spurious word from 5 words, 4 from the topic description (from the topic model), and 1 randomly selected from another topic description. (81 tasks per subject, $10 \sim 15$ seconds per task)

2. T2: Topic assignment (comment): test whether a comment from news communities can be correctly assigned to the model-generated topic. Given the comment, subjects must identify a correct topic for the comment from 4 topic descriptions, where 3 of them are randomly chosen. (40 tasks per subject, $20 \sim 30$ seconds per task)

3. T3: Topic assignment (title): test whether a news article title can be correctly assigned to the model-generated topic. Given the news title, subjects must provide a score (from 0: 'not at all related' to 5: 'very related') to each of 4 presented topic description, where 3 of them are tier 1, 2, and 3 topics of the given news title and the other is a randomly chosen topic. (60 tasks per subject, $15 \sim 20$ seconds per task)

4. T4: Topic similarity (description): test whether a cosine similarity between a pair of topic embeddings (averaged BERT embeddings) correctly aligns with the human-evaluated semantic similarity. Subjects must provide a score (from 0: 'not at all similar' to 5: 'very similar') to a given pair of topic descriptions. In this task, the topic descriptions are given

Table S7: Fitting parameters for the empirical data in Fig. 2 (Global TM)

Name	Title			Comment	Similarity		
	α_1	α_2	α_3	α_c	a	b	s
Mother Jones	0.2269	0.1754	0.1024	1.0026	0.1315	-0.6535	0.9846
Atlantic	0.1399	0.0597	-0.0114	0.9665	0.1322	-0.6595	0.9733
The Hill	0.2363	0.1863	0.1706	1.0893	0.0989	-0.8295	1.1807
Breitbart	0.1849	0.1061	0.0519	1.0203	0.1164	-0.6515	0.9942
Gateway Pundit	0.3785	0.2847	0.2578	1.0359	0.0948	-0.8778	1.2302

by top-10 representative keywords instead of 4. (60 tasks per subject, 20 seconds per task)

5. T5: Topic similarity (comment): test whether a cosine similarity between a pair of comment (BERT) embeddings correctly aligns with the human-evaluated semantic similarity. Subjects must provide a score (from 0: 'not at all similar' to 5: 'very similar') to a given pair of comments. (50 tasks per subject, $20 \sim 30$ seconds per task)

We aimed to get 6 participants per survey item, but the number of subjects for each task and each topic (survey items) consists of a Gaussian distribution (centers at 6) due to the random assignment of the platform. The survey results are summarized in the Fig. S5, S6, S7, S8, and S9.

6 Fitting parameters for the empirical data

6.1 Global topic model

Here, we provide a detailed description of the fitting parameters for the empirical data from online news communities used in the main manuscript (Fig. 2), where the global topic model is used. In Fig. 2b, the relative title topic frequency of the news ("Title") is fitted to a $y \propto \ln(x)x^{-\alpha_q}$, where q indicates the tier $(1, 2, 3)$. In Fig. 2c, the relative comment topic frequency ("Comment") is fitted to a power-law distribution, $y \propto x^{-\alpha_c}$. In Fig. 2d and 2e, both the probability density of the topic similarity histogram ("Similarity") and the relative number of comments histogram ("# of comment") is fitted to a log-normal distribution, $y \propto e^{-\ln^2(\frac{x-a}{b})/2s^2}$. All of the parameters for each community are summarized in Table S7.

6.2 Local topic model

In the main manuscript, we have shown that the statistical distribution of the empirical data, which is an outcome of the classification of the global topic model, matches our model results. For further verification and to demonstrate the robustness of the data distribution, we also present empirical data, which is

Table S8: Fitting parameters for the empirical data in Extended Data Fig. S10 (Local TM)

Name	Title			Comment	Similarity		
	α_1	α_2	α_3	α_c	s	a	b
Mother Jones	0.2670	0.2373	0.1603	1.4487	0.1365	-0.6650	0.9586
Atlantic	0.0951	-0.0239	-0.0946	0.9831	0.1798	-0.5377	0.8108
The Hill	0.1908	0.1471	0.0765	0.9076	0.1307	-0.5675	0.9146
Breitbart	0.1838	0.1491	0.2533	0.8983	0.1177	-0.6546	0.9869
Gateway Pundit	0.2321	0.2854	0.2533	0.9870	0.1178	-0.6501	0.9985

classified by the respective local topic model and their fittings in Fig. S10. The fitting parameters for local models are summarized in Table S7.

7 Discussion on the correspondence between empirical semantic network and comment network

In our study, we calibrated the initial frequency (and similarity) distribution of both general and community semantic networks from the empirical data. However, there are some noteworthy points to rigorously address the validity of this approach.

The point here is that the semantic network is not directly observable from the empirical data; rather, it's a structural concept that we employed to explain the underlying dynamics of the collective mind and to construct the computational model. The only thing we can directly observe are comments, which correspond to the comment network in our model. Hence, we need to justify that the semantic network also follows the same distribution as the empirical comment distribution.

In the case of the community semantic network, the reason is quite straightforward; if we update our community semantic network to a comment network with memory strength $\lambda_m \neq 1$, the distribution of the community semantic network will eventually converge to the comment network. This can be easily shown by considering the update rule of the community semantic network. For example, if we consider the frequency update rule, the community semantic network's frequency at time $t+1$ is given by

$$f_{i,t+1}^k = \lambda_m f_{i,t}^k + (1-\lambda_m)\hat{\bar{f}}_{i,t}^k, \tag{14}$$

where the term $\hat{\bar{f}}_{i,t}^k = \hat{f}_{i,t}^k / \sum_j \hat{f}_{j,t}^k$ denotes relative comment frequency distribution. If we assume the frequency distribution of the semantic network is stationary, i.e., $f_{i,t+1}^k = f_{i,t}^k$, the equation becomes

$$f_{i,t}^k = \lambda_m f_{i,t}^k + (1-\lambda_m)\hat{\bar{f}}_{i,t}^k, \tag{15}$$

and therefore $f_{i,t}^k = \hat{\bar{f}}_{i,t}^k$, which means the community semantic network's frequency distribution will converge to the comment frequency distribution in the long run as a steady state. The same logic applies to the weight update rule as well. More rigorous proof can be done by showing the distance between probability distributions (either L1 norm or KL divergence) decreases as the iteration goes to infinity, and is related to concepts like mixing in the Markov process.

For the general semantic network, if we assume the general semantic network is an averaged version of all existing community semantic networks (since it represents the general popularity and semantic structure of the entire population), the general semantic network's distribution will also converge to the comment network's distribution.

8 Analysis on basic behavior of computational models

In the main manuscript (especially Fig. 4b-c, Fig. 5b-c, and Extended Data Fig. 3), we showed that the comment topic profile is getting closer to or moving away from the topic profile of the general semantic network, depending on its initial state. In this section, we describe these behaviors in more detail and discuss the underlying mechanism. Hereafter, we consider the computational model with $\lambda_m \neq 1$, since the transition of comment topic profile is impossible with an unchanging community ($\lambda_m = 1$).

The general semantic network is the main source of events, hence greatly affecting the topic distribution of the filtered events (news) as well. More precisely, in our model, the $q-$th tier news topic frequency is roughly proportional to $-\ln(r_i^g)(r_i^g)^{\alpha_q/2}$ (factor of $1/2$ comes from the near-randomness of similarity-based second filter), and this proportionality becomes exact in the extreme case of $\lambda_f = 0$. Naturally, the high frequency of the news topic will lead to the high frequency of the comment frequency (amplified by the sampled comment multiplier), which will affect the community semantic network's frequency via memory strength. While it is nearly infeasible to analytically solve the full model, with a similar argument as above (Supplementary Note 8), we can expect that this effect will lead the community semantic network's frequency closer to the general frequency distribution (and especially the rank of them) in the long run. A similar argument can be made for the weight as well, since the weight is updated by the co-occurrence of the topics in the news, which is directly affected by the general semantic network's similarity pattern.

But there is another factor that prevents the community semantic network from fully converging to the general semantic network: the randomness in the comment generation process. Since the comment generation process is stochastic, the comment topic profile will not be exactly the same as the general semantic network's topic profile, even if the community semantic network is fully converged to the general semantic network. This randomness then affects the

community semantic network and repels it from the general semantic network till the two forces are balanced. This effect is well shown in the Extended Data Fig. 3, where the distance between two semantic networks converges to the same non-zero value regardless of its starting position (SD 0.0 or 1.0).

Interestingly, we find that this equilibrium distance is inversely proportional to both filter strength (λ_f) and memory strength (λ_m). It is straightforward to see that the distance is inversely proportional to the memory strength since high memory strength suppresses the randomness in the comment generation process and affects the community semantic network. The inverse proportionality to the filter strength is somewhat counterintuitive at first glance since the low filter strength should lead the community semantic network to be closer to the general semantic network. On closer inspection, we find that the distance of the high filter strength case (0.8) from SD 0.0 in fact decreases over time after the initial soaring (around $t = 50$), suggesting that the source of inverse proportionality comes from something that is changing during the iteration. Given that the only thing that changes during the iteration is the community semantic network, we can infer that the community semantic network that is already attracted and become similar to the general one reinforces its effect, with the aid of high filter strength. This paradoxical trend is well-aligned with the findings described in the effect of influence (in the main manuscript), where the community with high filter and high memory strength is more prone to internalize and keep the influence from the influences. Further analyzing the asymptotic behavior of the coarse-grained, simplified (and thus analytical tractable) version of this framework would be a promising direction for theoretical future work.

9 Hypersensitive filter ($\lambda_f > 1$)

In the main manuscript, we set our model's filter strength (λ_f) between 0 and 1. However, our formulation enables us to expand this into the case where the filter strength λ_f is greater than 1, which we call a hypersensitive filter. The hypersensitive filter is not only more inclined to the community semantic network but also actively avoids the general semantic network by negatively assessing their frequency and weights during the filtering process. Since it extrapolates from the original linear interpolation range, the criteria frequency and weight (which represents the worldview of the filter) in both equations 5 and 6 can be negative. Although negative frequency and weight are not meaningful in our model, it doesn't matter since they only appear in the intermediate step of the filtering process. Precisely, we only use the rank of those values, which is perfectly valid even if any of the values are negative.

We first investigate the behavior of the model with a hypersensitive filter by varying the filter strength λ_f from 0.2 to 3.0 and fixing the memory strength $\lambda_m = 0.9$ (Supplementary Fig. S10a). We found that the model with a relatively weak hypersensitive filter shows a similar trend as the model with $\lambda_f < 1$; the distance between the general and community semantic network decreases over time. But, as the filter strength increases (typically $\lambda_f > 1.5$), the distance

between two semantic networks increases over time, suggesting that a strong hypersensitive filter can repel the community semantic network from the general semantic network. This is well shown in the t-SNE plot of the comment frequency profile (Supplementary Fig. S10b), where the model with $\lambda_f = 3.0$ shows a clear separation from the general semantic network while the model with $\lambda_f = 0.2$ is attracted. This separation resembles the behavior of a community with an extreme echo chamber effect, which strongly rejects the conventional norm and reinforces the community-specific view that is drastically different from the rest of society. With these demonstrations, we show that our model is capable of capturing those radical behaviors of the community by simply tuning the filter parameter.

References

[1] Hebb, D. O. *The organization of behavior: A neuropsychological theory* (Psychology press, 2005).

[2] Grootendorst, M. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794* (2022).

[3] Moulavi, D., Jaskowiak, P. A., Campello, R. J., Zimek, A. & Sander, J. Density-based clustering validation. In *Proceedings of the 2014 SIAM international conference on data mining*, 839–847 (SIAM, 2014).

[4] Palan, S. & Schitter, C. Prolific. ac—a subject pool for online experiments. *Journal of behavioral and experimental finance* **17**, 22–27 (2018).

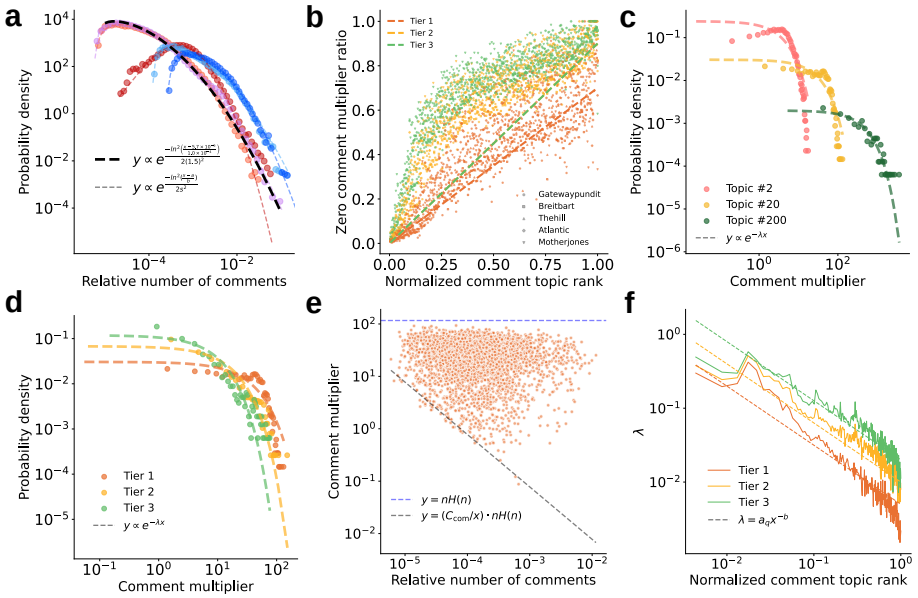

Figure S1: **Empirical data distribution for computational model calibration.** **a**, Histograms of the relative number of comments for each community and their fittings. Individual fittings are drawn in dotted lines, and thick dashed lines indicate the distribution used to calibrate the computational model. **b**, Zero comment multiplier ratio (Z_q) for community and tier, with the linear approximation used for the model calibration. Note that we used the same linear function for Z_2 and Z_3. **c**, comment multiplier histogram for different topic ranks $(2, 20, 200)$ and their fittings. **d**, comment multiplier histogram for different tiers $(1, 2, 3)$ and their fittings. **e**, Scatter plot between the comment multiplier of topic 20, tier 1 and their relative number of comments. Two theoretical boundaries are drawn in dashed lines, where $H(n)$ is a harmonic series to n and C_{com} is the inverse of the mean total number of comments (in the time interval of 1 month). **f**, Fitting exponent λ for the exponential fitting of comment multiplier distribution with respect to their normalized topic rank (and their fittings). The data in **c-f** is aggregated from the all-time data of The Hill.

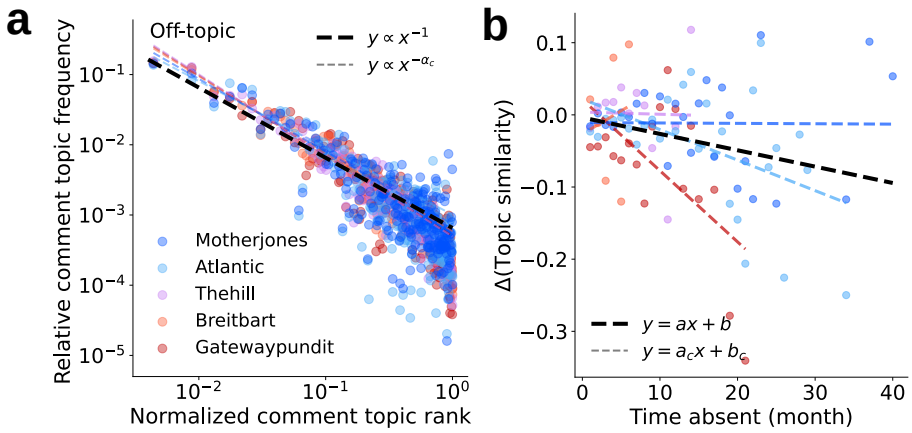

Figure S2: **Verification of the model assumptions.** **a**, Relative comment topic frequency distributions for off-topic comments from online news communities and their best fitting lines. The fitted exponents α_c are 1.03 for Motherjones, 0.95 for Atlantic, 1.15 for Thehill, 1.10 for Breitbart, and 1.14 for Gatewaypundit. **b**, Difference in topic similarity as a function of the time absent from the news title in online news communities and their best linear fitting lines. For each point, all data from instances of topic pairs that were missing for the same months in the same community were averaged, to highlight the dependence between the time absent and the similarity difference. The slope for individual linear fitting lines are -5.35×10^{-5} for Motherjones, -4.23×10^{-3} for Atlantic, -3.01×10^{-4} for Thehill, 6.43×10^{-3} for Breitbart, and -9.86×10^{-3} for Gatewaypundit. Black dashed line indicates the aggregated fitting line for all 5 communities, where its slope is -2.27×10^{-3}. The legend from panel **a** is shared with panel **b**.

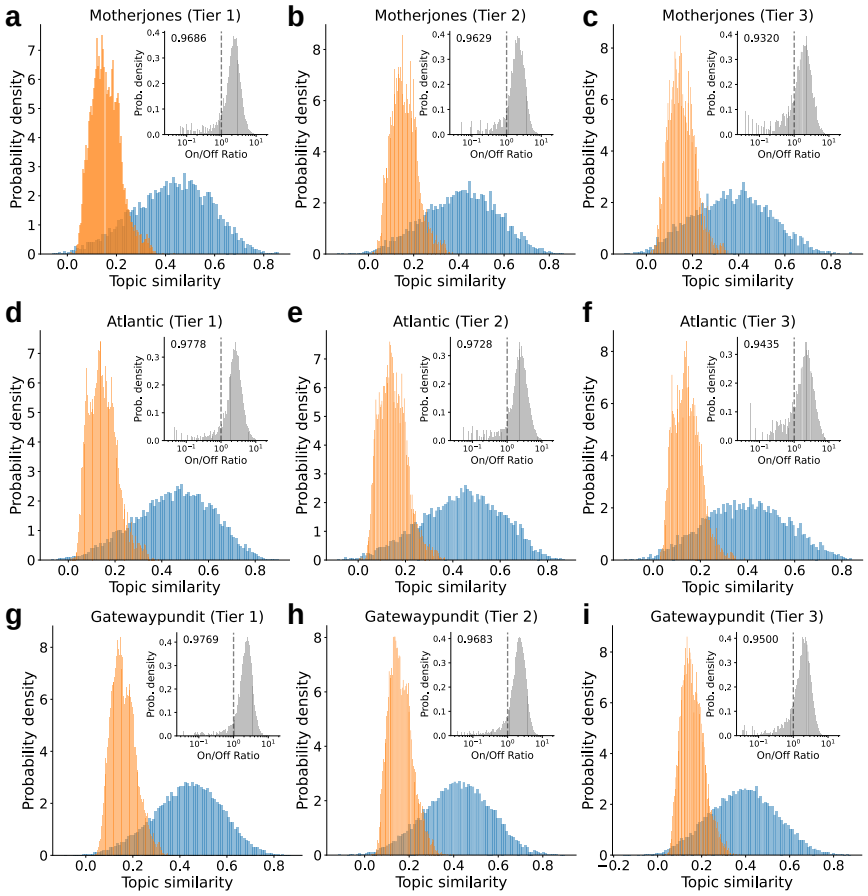

Figure S3: **Comparison between on-topic and off-topic comment similarity for the same topic pairs.** For each panel, the orange and blue histograms indicate the distribution of average cosine similarity between off-topic and on-topic comment embeddings for the same topic pairs, aggregated from the entire data of the respective community. The Inset histogram shows the ratio between on-topic and off-topic average cosine similarity for the same topic pairs, where the dashed line indicates the ratio of 1 and the annotated number indicates the ratio of this ratio is greater than 1. Each panel is titled with the community name and the tier of the title topic that is used to determine on-topic comments.

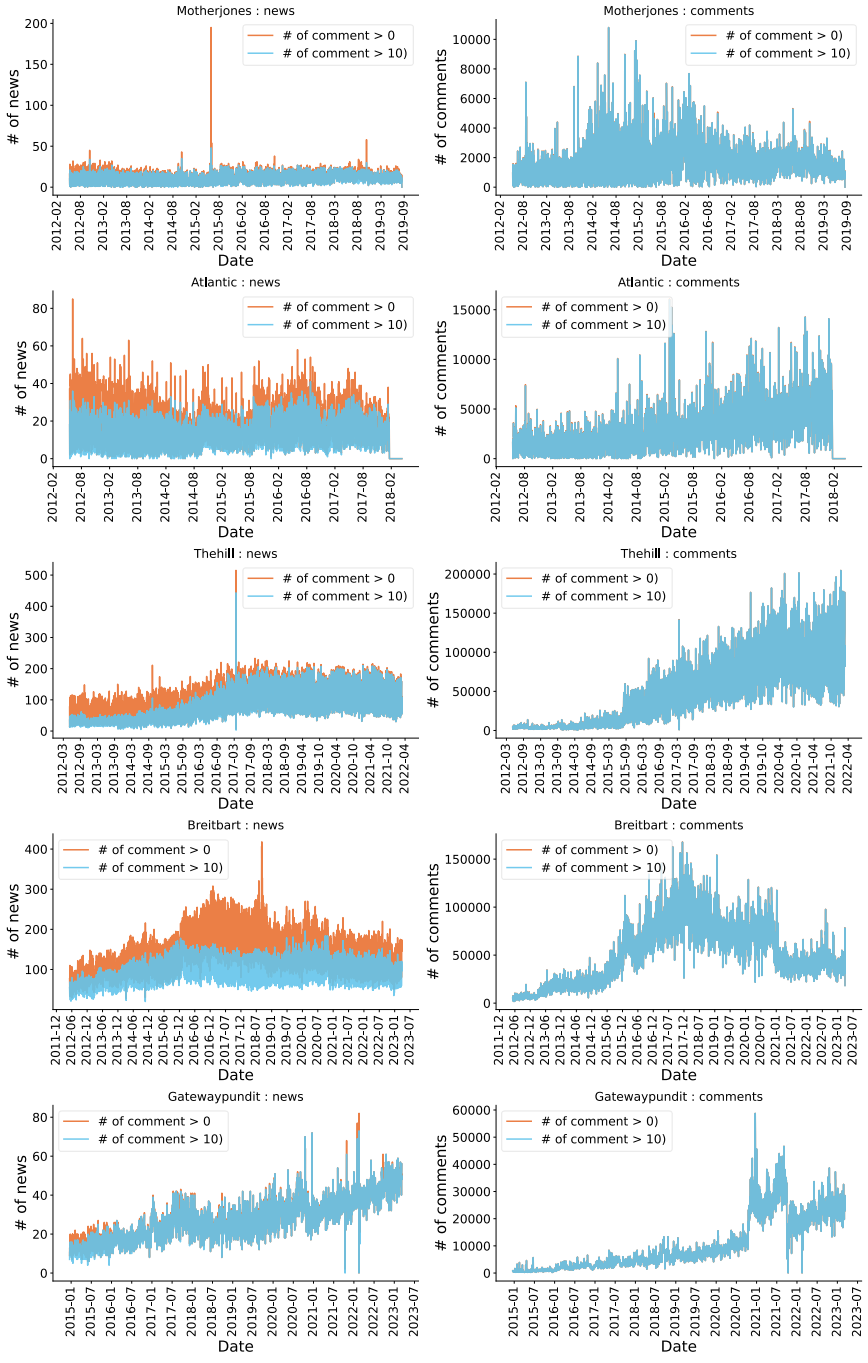

Figure S4: **Time series of the number of news and comments for** 5 **online news community.** Before (orange) and after (blue) the filtering with $\theta_n = 10$ is plotted.

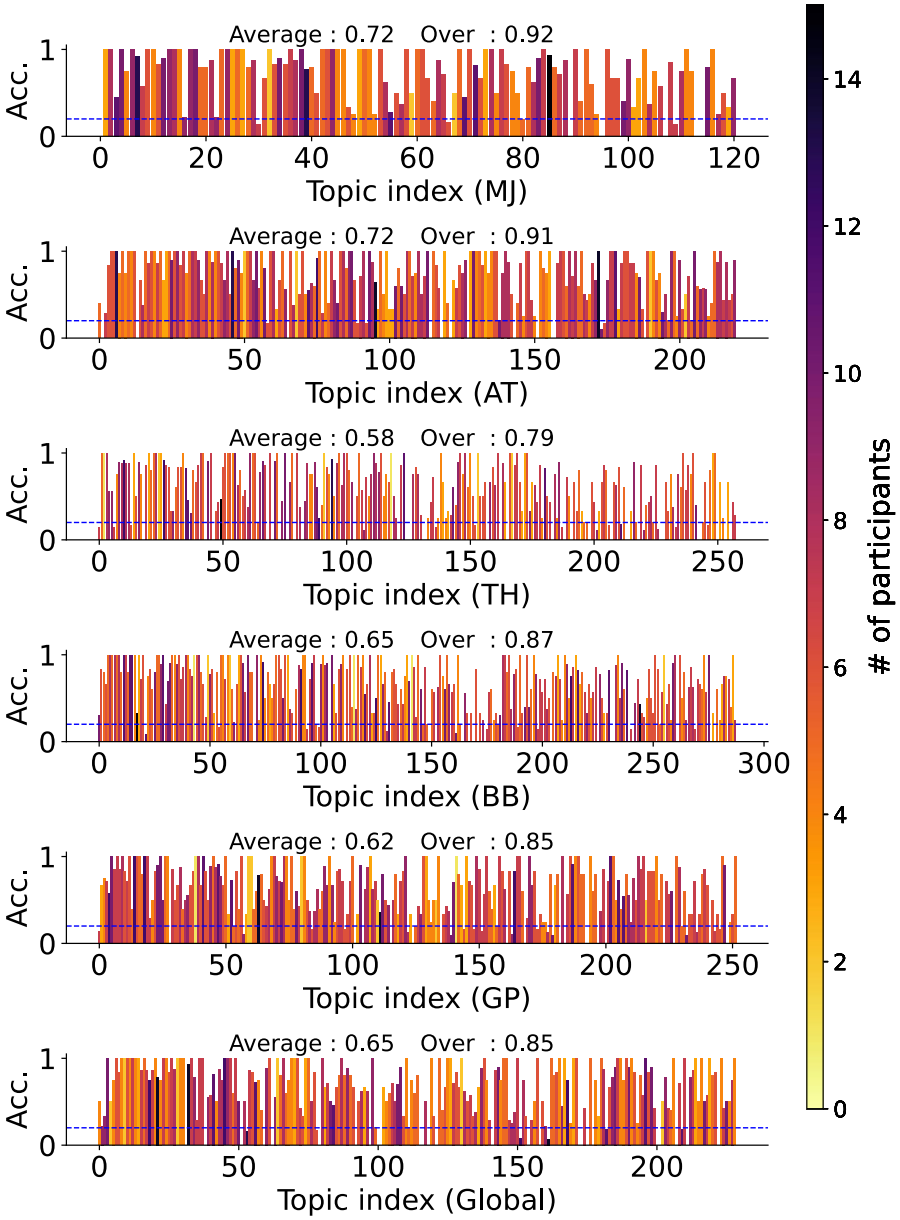

Figure S5: **Results from the topic model survey, word intrusion task (T1).** The average indicates the mean accuracy of all topics' results, while Over indicates the percentage of topics that have an accuracy over 20% (chance level).

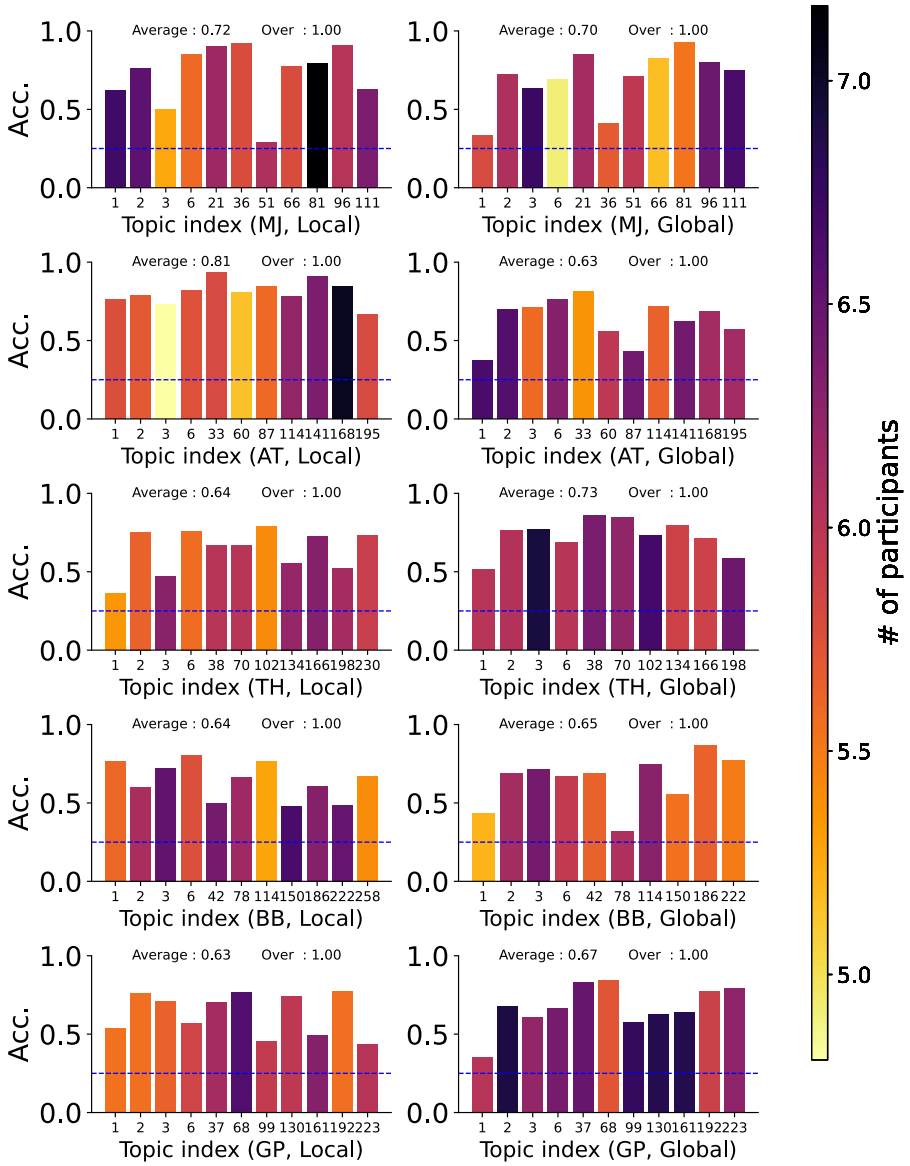

Figure S6: **Results from topic model survey, topic assignment (comment) task (T2).** Topics from both the local topic model and the global topic model from the same site are displayed next to each other. The average indicates the mean accuracy of all topics' results, while Over indicates the percentage of topics that have an accuracy over 25% (chance level).

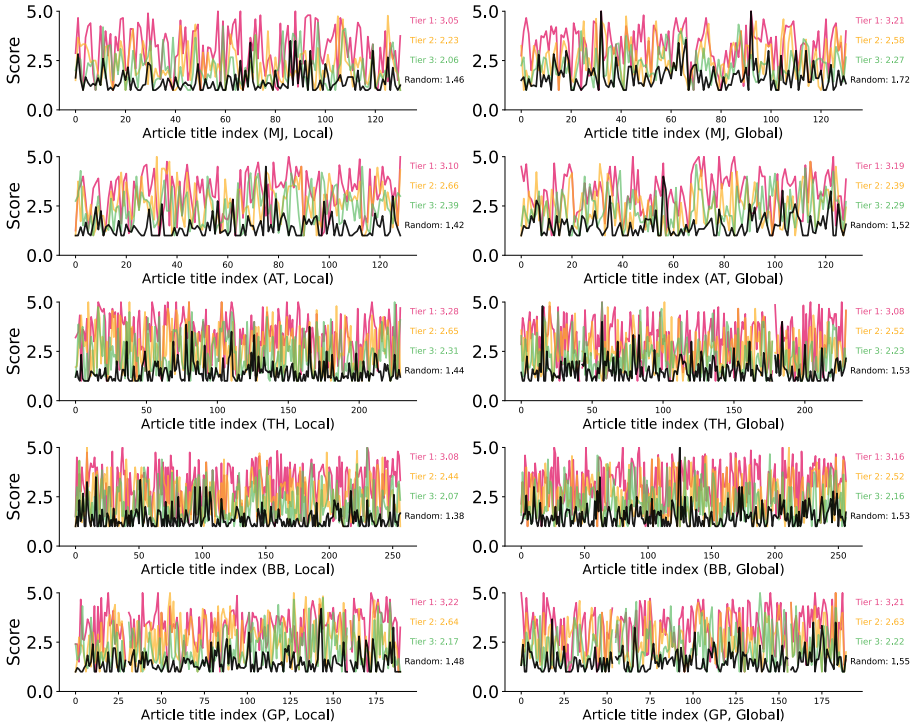

Figure S7: **Results from topic model survey, topic assignment (title) task (T3).** Red, yellow, green, and black lines indicate the averaged scores for the (correct) tier 1, 2, and 3 topics, and a random topic, respectively. Topics from both the local topic model and the global topic model from the same site are displayed next to each other. Average scores for each category are shown on the right side of the plot.

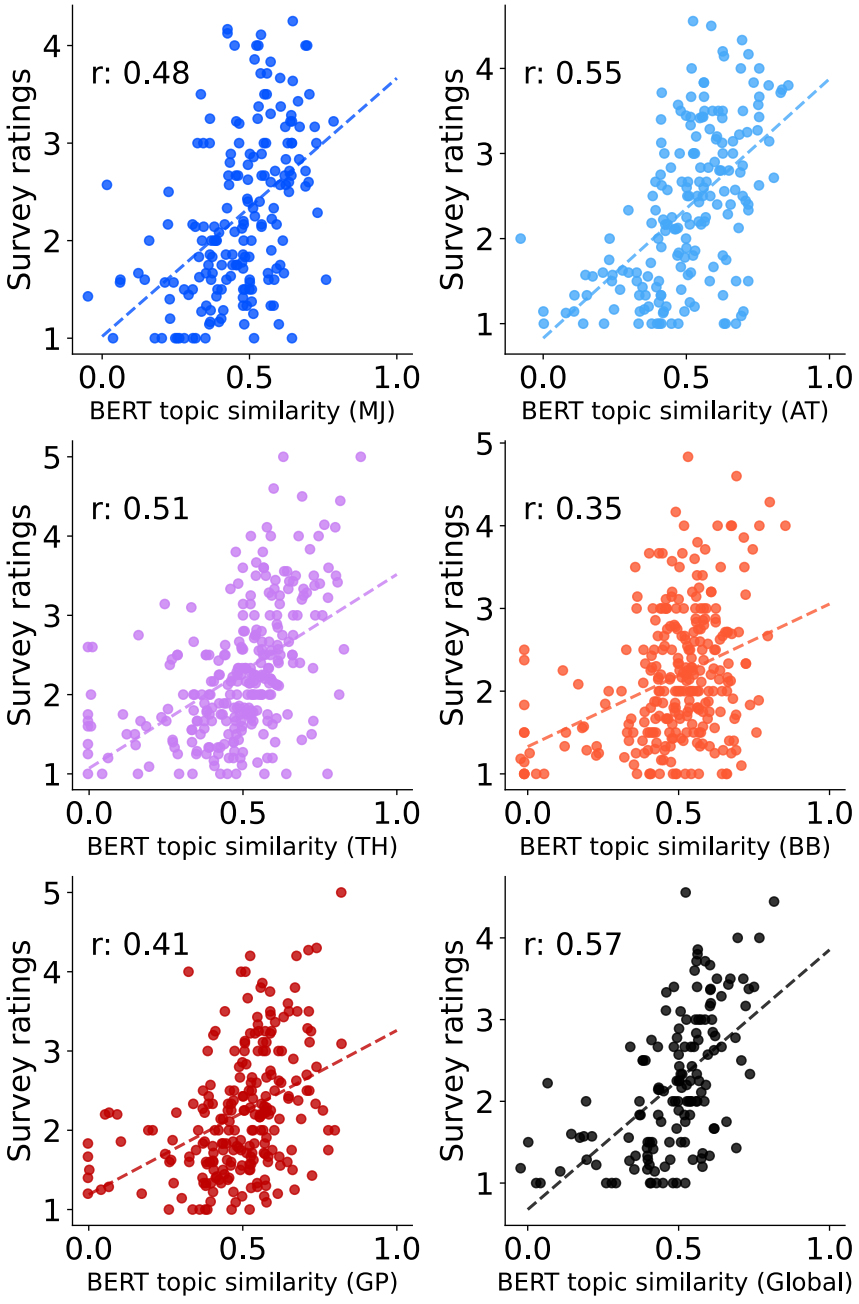

Figure S8: **Results from topic model survey, topic similarity (description) task (T4).** Pearson correlation r is shown in each plot, and the linear fit is shown as a dashed line.

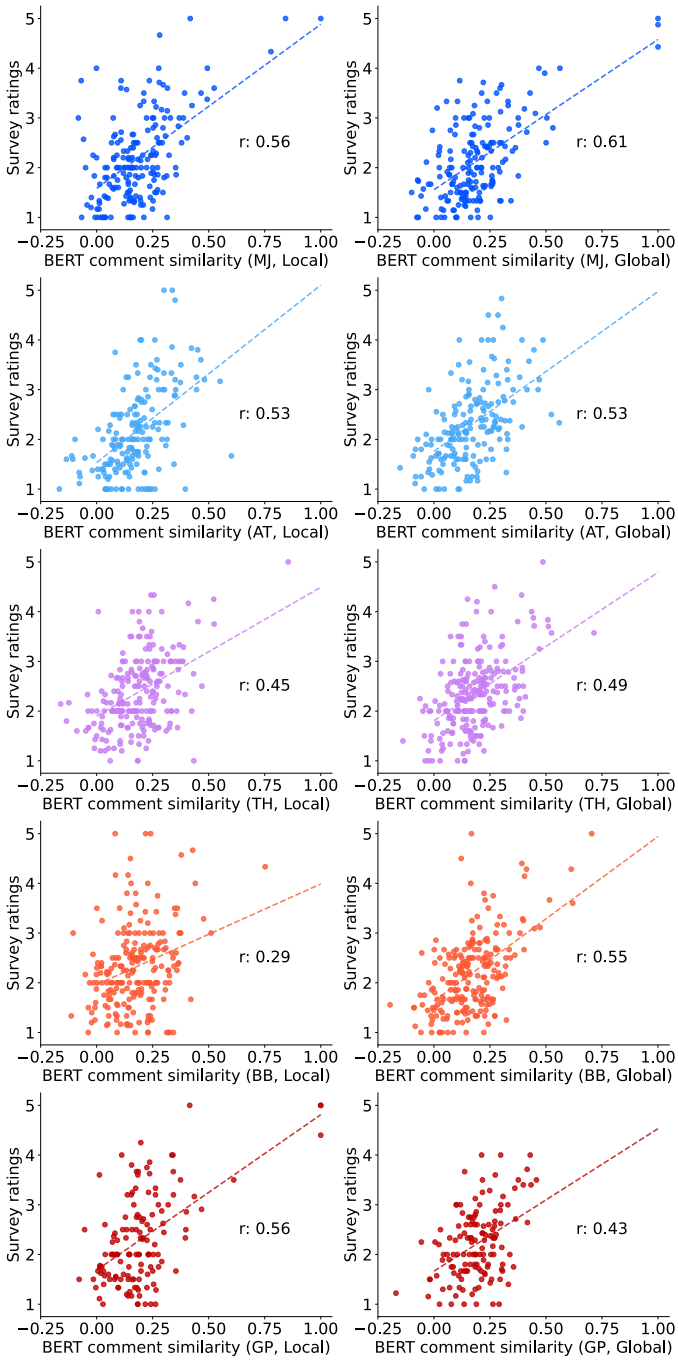

Figure S9: **Results from topic model survey, topic similarity (comment) task (T5).** Pearson correlation r is shown in each plot, and the linear fit is shown as a dashed line.

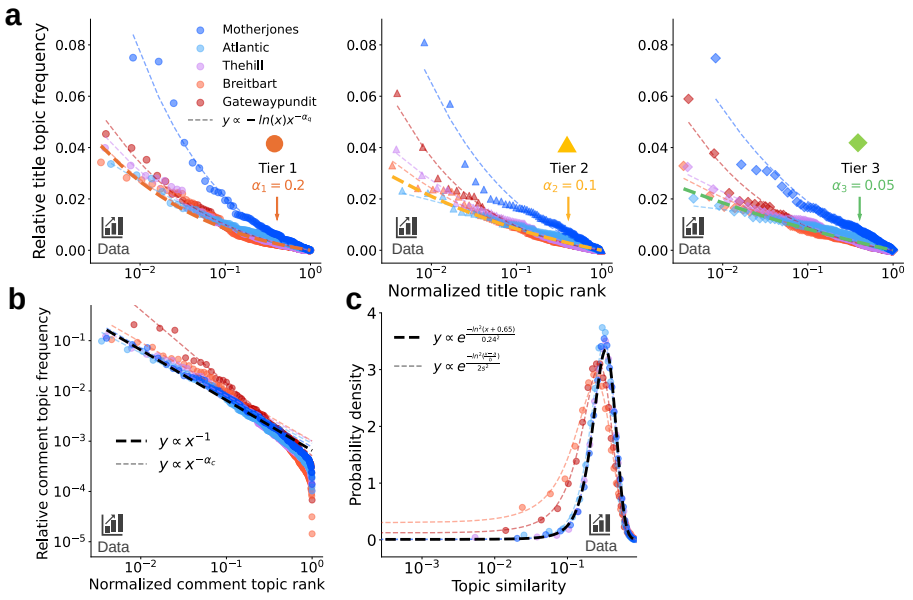

Figure S10: **Quantitative comparison of the real data (local TM) and the model output.** **a**, Relative article title topic frequency of tier 1 (left), tier 2 (middle) and tier 3 (right) from each online news communities. **b**, Relative comment topic frequency distribution from online news communities. **c**, Topic similarity histogram from online news communities. Individual fittings are drawn in dotted lines, and thick dashed lines indicate the distribution used to calibrate the computational model. All of the topic frequency distributions (**a, b**) are sorted by their normalized topic rank. A legend in **a**(left) shows the color scheme used to represent the data from each online community, which is applied consistently across panels **b** and **c**. All of the fitting parameters for real data ($\alpha_q, \alpha_c, a_c, b_c, s_c$) are listed in Supplementary Table S8.

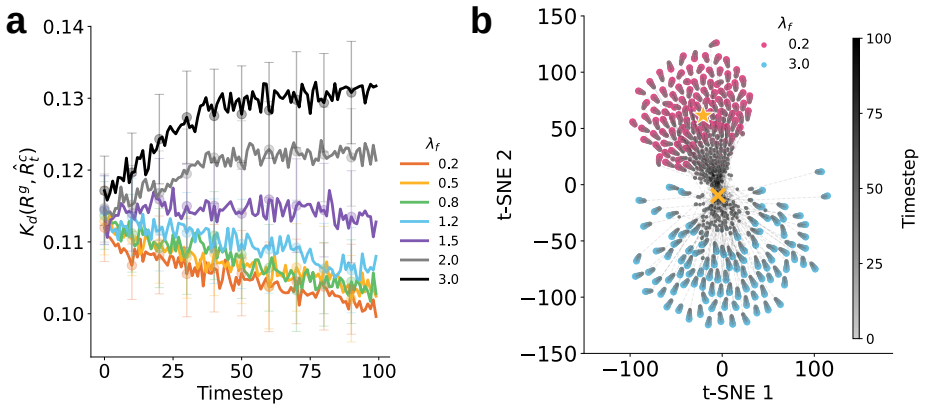

Figure S11: **Behavior of model with hypersensitive filter ($\lambda_f > 1$). a**, Kendall-tau rank distance (K_d) between relative topic frequencies of general semantic network (R^g) and comment frequencies of community semantic network at time step t ($\hat{R}_t^c$) with various λ_f ranging from 0.2 fo 3.0, where the initial community frequencies are perturbed from general frequencies by log-normal noise with standard deviation of 0.2. Data is gathered from $1,000$ iterations, and the errorbar indicates ± 1 standard deviation and is plotted every 10 time step. **b**, The t-SNE plot of 100 trajectories of the comment frequency profile for the model simulation with $\lambda_f = 0.2$ (red) and $\lambda_f = 3.0$ (blue), all started from the same initial frequency (orange cross) and attracted by the same general semantic network (orange star). $\lambda_m = 0.9$ was used for all simulations.

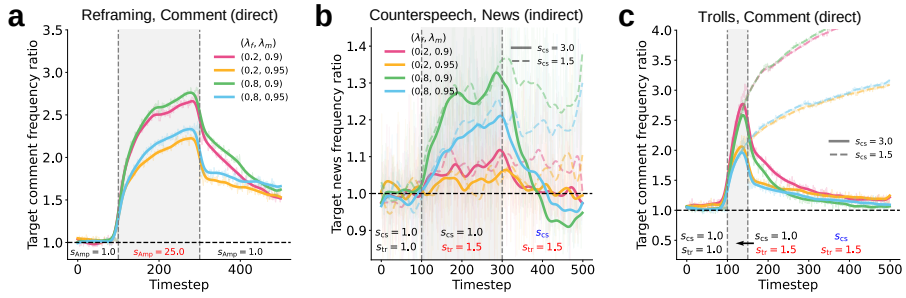

Figure S12: **Auxiliary plots from the influence result. a**, Ratios between baseline and influenced case of target topic frequency in the comment (reframing). **b**, Ratios between baseline and influenced case of target topic frequency in tier 1 news topic (counterspeech). **c**, Ratios between baseline and influenced case of target topic frequency in the comment but with earlier removal of trolls with $t = 150$ (trolls). All of the other details are the same as the corresponding plots in Fig.4 and 5 in the main manuscript.

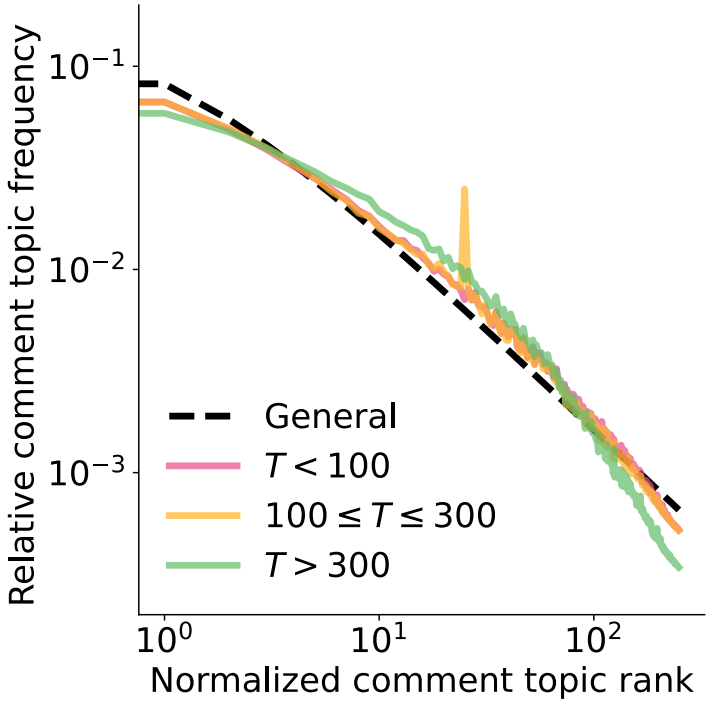

Figure S13: **Relative comment topic frequency distribution after the counterspeech.** The red, yellow, and green line indicates the relative frequency of the target topic in the comment topic profile before the trolls, after the trolls, and after the counterspeech (with trolls), respectively. $s_{\mathrm{tr}} = 1.5$ and $s_{\mathrm{tr}} = 3.0$ is used. The green line (with both troll and counterspeech existing) does not match with the original red line (before the trolls), instead, it overrepresents the high-rank topics ($r > 100$, except for the top few) while underrepresenting the low-rank ($r < 100$) topics.