**Online Supplement: Combining Crowd and Machine Intelligence to Detect False News on
Social Media**

Forthcoming at *MIS Quarterly*

Xuan Wei

Antai College of Economics and Management, Shanghai Jiao Tong University,
Shanghai, China {weix@sjtu.edu.cn}

Zhu Zhang*

State Key Laboratory of Management and Control for Complex Systems, Institute of
Automation, Chinese Academy of Sciences, Beijing, CHINA {zhu.zhang@ia.ac.cn}

Mingyue Zhang*

School of Business and Management, Shanghai International Studies University, Shanghai,
China {zhangmy@shisu.edu.cn}

Weiyun Chen

School of Management, Huazhong University of Science and Technology,
Wuhan, China {chenweiyun@hust.edu.cn}

Daniel Dajun Zeng

State Key Laboratory of Management and Control for Complex Systems, Institute of
Automation, Chinese Academy of Sciences, Beijing, and School of Economics and Management
and School of Artificial Intelligence, University of Chinese Academy of Sciences,
Beijing, CHINA {dajun.zeng@ia.ac.cn}

*Corresponding authors

Note: Because of space limitations and the recommendations regarding advancing research transparency (Burton-Jones et al. 2021), we include an online supplement for this forthcoming paper (Wei et al., forthcoming). This online supplement is organized as follows. Section A provides definition of false news and relevant classification schemes. Sections B and C introduce additional details regarding method development and experimental design respectively. Section D provides additional experimental results.

Abstract

The explosive spread of false news on social media has severely affected many areas such as news ecosystems, politics, economics, and public trust, especially amid the COVID-19 infodemic. Machine intelligence has met with limited success in detecting and curbing false news. Human knowledge and intelligence hold great potential to complement machine-based methods. Yet they are largely underexplored in current false news detection research, especially in terms of how to efficiently utilize such information. We observe that the crowd contributes to the challenging task of assessing the veracity of news by posting responses or reporting. We propose combining these two types of scalable crowd judgments with machine intelligence to tackle the false news crisis. Specifically, we design a novel framework called CAND, which first extracts relevant human and machine judgments from data sources including news features and scalable crowd intelligence. The extracted information is then aggregated by an unsupervised Bayesian aggregation model. Evaluation based on Weibo and Twitter datasets demonstrates the effectiveness of crowd intelligence and the superior performance of the proposed framework in comparison with the benchmark methods. The results also generate many valuable insights, such as the complementary value of human and machine intelligence, the possibility of using human intelligence for early detection, and the robustness of our approach to intentional manipulation. This research significantly contributes to relevant literature on false news detection and crowd intelligence. In practice, our proposed framework serves as a feasible and effective approach for false news detection.

Keywords: false news, fake news, wisdom of crowds, hybrid intelligence, graphical model

A: Definition of False News and Its Classification

There are no universal definitions for false news and related concepts, including fake news, misinformation, and rumor. We clarify these concepts based on the most recent literature. First, we need to define the term news. Following the literature (Vosoughi et al. 2018), we purposefully use a broad definition. News refers to any news article or message that is published and propagated through media and has an assertion in it (Sharma et al. 2019; Vosoughi et al. 2018; Zhou and Zafarani 2020). In our research context, we treat each Weibo post or tweet as news. False news refers to news that contains false information. Based on the intention, false news can be divided into fake news (intentionally and verifiably false news) and unintentionally false news (Sharma et al. 2019).

In the literature, most classification schemes are about false information. Although information is a broader concept than news, we focus on the classification of false information in the following because we can easily apply these classification schemes to false news by limiting the context to news.

Based on the intention, false information can be divided into misinformation and disinformation. Misinformation refers to the “unintentionally spread false information that can be a result of misrepresentation or misunderstanding stemming from cognitive biases or lack of understanding or attention,” and disinformation refers to the “false information created and spread specifically with the intention to deceive” (Kumar and Shah 2018; Sharma et al. 2019; Wardle 2017).

Based on the knowledge, false information can be divided into opinion-based and fact-based. The most common opinion-based false information is fake reviews on e-commerce websites

where a unique ground truth does not exist, while fact-based false information consists of lies about entities that have a unique ground truth value (Kumar and Shah 2018).

Based on the means employed to falsify information, false information can be divided into seven potentially overlapping types: “satire or parody (i.e., no intention to cause harm but has potential to fool), fabricated content (i.e., completely false), misleading content (i.e., misleading use of information to frame an issue), imposter content (i.e., genuine sources impersonated with false sources), manipulated content (i.e., genuine information or imagery manipulated to deceive), false connection (i.e., headlines, visuals, or captions that do not support the content), and false context (i.e., genuine content shared with false contextual information)” (Sharma et al. 2019; Wardle 2017).

Another concept related to false news is rumor. The definition of rumor from the Oxford Dictionary is “a currently circulating story or report of uncertain or doubtful truth,” and it’s also defined as the “social phenomena of a news story or claim spreading or diffusing” through social media (Vosoughi et al. 2018). Overall, for rumor, we are not sure about the veracity and the intention.

In this work, we focus on the veracity of news, regardless of the intention and the means of falsifying information, because intention is often unverifiable in reality (Vosoughi et al. 2018). As the definition in the literature often varies, we may still use the phrase “fake news” or “rumor” in certain cases, depending on the original context that we are referring to.

B: Method Development

B1. News Features

We represent each piece of news by its content and context features. The content features include the news text and whether each piece of news has external links, images, and videos. Context

features include posting time and author profiles. Table B1 lists all features, their descriptions, and whether they are included in the Weibo and Twitter datasets.

Table B1. News Features Considered in Our Datasets

News Features			Included in Dataset?	
Type	Name	Description	Weibo	Twitter
Content Features	Text	Text content of the news	•	•
	Has_external_link	Whether the news has external links	•	•
	Has_image	Whether the post has images	•	•
	Has_video	Whether the post has videos	•	•
Context Features	Post_hour	The hour when the news is posted (dummy variable)	•	•
	Gender	Gender of the author	•	
	Is_verified	Whether the author is a verified user	•	•
	VIP	VIP level (0 represents no VIP)	•	
	Account_type	Whether the account is personal or organizational	•	
	Has_avatar	Whether the author has an avatar	•	•
	Has_bio	Whether the author has a short bio		•
	Log_tenure	Log of # of days since the author joined the platform		•
	Log_num_followings	Log of # of users the author has followed	•	•
	Log_num_followers	Log of # of the author's followers	•	•
	Log_num_posts	Log of # of posts the author has posted	•	•
	Log_num_likes	Log of # of likes the author has received		•
	Log_num_lists	Log of # of lists the author has		•

B2. Update Formulas for the Inference of the CLNAM Model

For ease of notation, for all Y 's, let X 's denote their logits. For example, $X_{n,m}^{(1)} = \text{logit}(Y_{n,m}^{(1)})$.

The formulas for updating each coordinate are as follows:

- Formulas for updating the posterior of $\mu_m^{(1)}$ and $\sigma_m^{(1)}$:

$$\begin{aligned}
\omega_m^{(1)} &= \nu_m^{(1)} \left(\frac{\omega_0^{(1)}}{\nu_0^{(1)}} + \sum_{n=1}^N \lambda_n X_{n,m}^{(1)} \right), \quad \nu_m^{(1)} = \{(\nu_0^{(1)})^{-1} + \sum_{n=1}^N \lambda_n\}^{-1} \\
\alpha_m^{(1)} &= \alpha_0^{(1)} + \frac{1}{2} \sum_{n=1}^N \lambda_n, \quad \beta_m^{(1)} = \beta_0^{(1)} + \frac{1}{2} \left\{ \frac{(\omega_0^{(1)})^2}{\nu_0^{(1)}} - \frac{(\omega_m^{(1)})^2}{\nu_m^{(1)}} + \sum_{n=1}^N \lambda_n (X_{n,m}^{(1)})^2 \right\}
\end{aligned} \tag{B1}$$

To update the posterior of $\bar{\mu}_m^{(1)}$, and $\bar{\sigma}_m^{(1)}$, we only need to replace λ_n with $1 - \lambda_n$ in the above formulas.

- Formulas for updating the posterior of $\mu^{(2)}$ and $\sigma^{(2)}$:

$$\begin{aligned}
\omega^{(2)} &= \nu^{(2)} \left(\frac{\omega_0^{(2)}}{\nu_0^{(2)}} + \sum_{n=1}^N \sum_{k=1}^{K(n)} \tau_{n,k} X_{n,k}^{(2)} \right), \quad \nu^{(2)} = \{(\nu_0^{(2)})^{-1} + \sum_{n=1}^N \sum_{k=1}^{K(n)} \tau_{n,k}\}^{-1} \\
\alpha^{(2)} &= \alpha_0^{(2)} + \frac{1}{2} \left\{ \sum_{n=1}^N \sum_{k=1}^{K(n)} \tau_{n,k} \right\}, \quad \beta^{(2)} = \beta_0^{(2)} + \frac{1}{2} \left\{ \frac{(\omega_0^{(2)})^2}{\nu_0^{(2)}} - \frac{(\omega^{(2)})^2}{\nu^{(2)}} + \sum_{n=1}^N \sum_{k=1}^{K(n)} \tau_{n,k} (X_{n,k}^{(2)})^2 \right\}
\end{aligned} \tag{B2}$$

To update the posterior of $\bar{\mu}^{(2)}$ and $\bar{\sigma}^{(2)}$, we only need to replace $\tau_{n,k}$ with $1 - \tau_{n,k}$ in the above formulas.

- Formulas for updating the posteriors of ϕ and $\bar{\phi}$:

$$\begin{aligned}
g &= g_0 + \sum_{n=1}^N \lambda_n Y_n^{(3)}, \quad h = h_0 + \sum_{n=1}^N \lambda_n \\
\bar{g} &= \bar{g}_0 + \sum_{n=1}^N (1 - \lambda_n) Y_n^{(3)}, \quad \bar{h} = \bar{h}_0 + \sum_{n=1}^N (1 - \lambda_n)
\end{aligned} \tag{B3}$$

- Formulas for updating the posterior of $d_{n,k}$:

$$\begin{aligned}
\tau_{n,k} &\propto \exp \left\{ \lambda_n \mathbb{E}[\log p] + (1 - \lambda_n) \mathbb{E}[\log \bar{\eta}] \right. \\
&\quad \left. - \frac{1}{2} \log(2\pi) - \frac{1}{2} \{ \log \beta^{(2)} - \psi(\alpha^{(2)}) \} - \frac{1}{2} \nu^{(2)} - \frac{\alpha^{(2)}}{2\beta^{(2)}} (X_{n,k}^{(2)} - \omega^{(2)})^2 \right\} \\
1 - \tau_{n,k} &\propto \exp \left\{ \lambda_n \mathbb{E}[\log(1 - p)] + (1 - \lambda_n) \mathbb{E}[\log(1 - \bar{\eta})] \right. \\
&\quad \left. - \frac{1}{2} \log(2\pi) - \frac{1}{2} \{ \log \bar{\beta}^{(2)} - \psi(\bar{\alpha}^{(2)}) \} - \frac{1}{2} \bar{\nu}^{(2)} - \frac{\bar{\alpha}^{(2)}}{2\bar{\beta}^{(2)}} (X_{n,k}^{(2)} - \bar{\omega}^{(2)})^2 \right\}
\end{aligned} \tag{B4}$$

- Formulas for updating the posterior of z_n :

$$\begin{aligned}
\lambda_n &\propto \exp \left\{ \mathbb{E}_q[\log \gamma] + Y_n^{(3)} \{ \psi(g) - \log(h) \} - \frac{g}{h} - \log(Y_n^{(3)}!) \right. \\
&\quad + \sum_{m=1}^M \left\{ -\frac{1}{2} \log(2\pi) - \frac{1}{2} \{ \log \beta_m^{(1)} - \psi(\alpha_m^{(1)}) \} - \frac{1}{2} \nu_m^{(1)} - \frac{\alpha_m^{(1)}}{2\beta_m^{(1)}} (X_{n,m}^{(1)} - \omega_m^{(1)})^2 \right\} \\
&\quad \left. + \sum_{k=1}^{K(n)} \{ \tau_{n,k} \mathbb{E}[\log \eta] + (1 - \tau_{n,k}) \mathbb{E}[\log(1 - \eta)] \} \right\} \\
1 - \lambda_n &\propto \exp \left\{ \mathbb{E}_q[\log(1 - \gamma)] + Y_n^{(3)} \{ \psi(\bar{g}) - \log(\bar{h}) \} - \frac{\bar{g}}{\bar{h}} - \log(Y_n^{(3)}!) \right. \\
&\quad + \sum_{m=1}^M \left\{ -\frac{1}{2} \log(2\pi) - \frac{1}{2} \{ \log \bar{\beta}_m^{(1)} - \psi(\bar{\alpha}_m^{(1)}) \} - \frac{1}{2} \bar{\nu}_m^{(1)} - \frac{\bar{\alpha}_m^{(1)}}{2\bar{\beta}_m^{(1)}} (X_{n,m}^{(1)} - \bar{\omega}_m^{(1)})^2 \right\} \\
&\quad \left. + \sum_{k=1}^{K(n)} \{ \tau_{n,k} \mathbb{E}[\log \bar{\eta}] + (1 - \tau_{n,k}) \mathbb{E}[\log(1 - \bar{\eta})] \} \right\}
\end{aligned} \tag{B5}$$

- Formulas for updating the posterior of γ :

$$e = e_0 + \sum_{n=1}^N \lambda_n, \quad f = f_0 + \sum_{n=1}^N (1 - \lambda_n) \tag{B6}$$

- Formulas for updating the posteriors of η and $\bar{\eta}$:

$$\begin{aligned}
a &= a_0 + \sum_{n=1}^N \sum_{k=1}^{K(n)} \lambda_n \tau_{n,k} & b &= b_0 + \sum_{n=1}^N \sum_{k=1}^{K(n)} \lambda_n (1 - \tau_{n,k}) \\
\bar{a} &= \bar{a}_0 + \sum_{n=1}^N \sum_{k=1}^{K(n)} (1 - \lambda_n) \tau_{n,k}, & \bar{b} &= \bar{b}_0 + \sum_{n=1}^N \sum_{k=1}^{K(n)} (1 - \lambda_n) (1 - \tau_{n,k})
\end{aligned} \tag{B7}$$

B3. Details of Learning Prior Beliefs from Data

In our proposed two-stage CAND framework, we observe that we can learn the prior beliefs for the CLNAM model from the information extraction stage. Specifically, in Task T1, the rate of false news in the validation dataset should be similar to the test set in which we want to assess the veracity. Likewise, the distribution describing the number of received reports in the

validation dataset of Task T1 should be similar to the one in the final test set. In tasks T1 and T2, when we train a classifier in the information extraction stage, we can foresee how the classifier will perform in an unknown test dataset by examining its behavior in the validation dataset that is used to estimate the prediction error for model selection. To leverage such information, we propose another graphical model to depict the underlying generative process of the validation datasets in the information extraction stage.

Figure B1. Graphical Representation for Learning Prior Beliefs from Data

Figure B1 presents the graphical representation. Let I denote the amount of news in the validation dataset of Task T1. The ground truth of news i is $\tilde{z}_i$ and follows $\text{Bernoulli}(\gamma)$. Following the same generative assumption of $Y_{n,m}^{(1)}$, we draw the prediction of news i by classifier m in T1 (denoted as $\tilde{Y}_{i,m}^{(1)} \in (0,1)$) depending on $\tilde{z}_i$. When news i is false (i.e., $\tilde{z}_i$ is 0), the prediction will follow a logistic-normal distribution with parameters $\mu_m^{(1)}$ and $\sigma_m^{(1)}$; when news i is true, the corresponding logit-normal distribution is parameterized by $\bar{\mu}_m^{(1)}$ and $\bar{\sigma}_m^{(1)}$. For news i in the same validation dataset, the number of received reports (denoted as $\tilde{Y}_i^{(3)} \in \{0,1,\dots\}$) is generated based on the ground truth $\tilde{z}_i$: $\tilde{Y}_i^{(3)} \sim \text{Poisson}(\tilde{z}_i\phi + (1 - \tilde{z}_i)\bar{\phi})$. In Task T2, when training a machine learning classifier for debunking response detection, we also have a validation

dataset. Let L denote the size and d_l be whether response l is a debunking response. The prediction generated by the classifier (denoted as $\tilde{Y}_l^{(2)} \in (0,1)$) is drawn according to the ground truth d_l . When response l in the validation dataset is a debunking response, the predicted probability follows a logistic-normal distribution with parameters $\mu^{(2)}$ and $\sigma^{(2)}$; and the parameters are $\bar{\mu}^{(2)}$ and $\bar{\sigma}^{(2)}$ when it is not.

We observe that the latent variables in Figure B1 can be divided into independent groups: $\{\mathcal{Y}\}$, $\{\mu_m^{(1)}, \sigma_m^{(1)}\}$, $\{\bar{\mu}_m^{(1)}, \bar{\sigma}_m^{(1)}\}$, $\{\mu^{(2)}, \sigma^{(2)}\}$, $\{\bar{\mu}^{(2)}, \bar{\sigma}^{(2)}\}$, $\{\phi\}$ and $\{\bar{\phi}\}$. Within each group, the posterior can be derived in a closed form easily. We treat the learned posteriors as priors of the CLNAM model. In detail, the priors are as follows:

- Priors of $\mu_m^{(1)}$ and $\sigma_m^{(1)}$:

$$\begin{aligned}\omega_0^{(1)} &= \nu_0^{(1)} \left(\sum_{i=1}^I \tilde{z}_i \tilde{X}_{i,m}^{(1)} \right), & \nu_0^{(1)} &= \left(\sum_{i=1}^I \tilde{z}_i \right)^{-1} \\ \alpha_0^{(1)} &= \frac{1}{2} \left(\sum_{i=1}^I \tilde{z}_i \right), & \beta_0^{(1)} &= \frac{1}{2} \left\{ -\frac{(\omega_0^{(1)})^2}{\nu_0^{(1)}} + \sum_{i=1}^I \tilde{z}_i (\tilde{X}_{i,m}^{(1)})^2 \right\}\end{aligned}\tag{B8}$$

To get the priors of $\bar{\mu}_m^{(1)}$ and $\bar{\sigma}_m^{(1)}$, we only need to replace $\tilde{z}_i$ with $1 - \tilde{z}_i$ in the above formulas.

- Priors of $\mu^{(2)}$ and $\sigma^{(2)}$:

$$\begin{aligned}\omega_0^{(2)} &= \nu_0^{(2)} \left(\sum_{l=1}^L \tilde{d}_l \tilde{X}_l^{(2)} \right), & \nu_0^{(2)} &= \left(\sum_{l=1}^L \tilde{d}_l \right)^{-1} \\ \alpha_0^{(2)} &= \frac{1}{2} \left(\sum_{l=1}^L \tilde{d}_l \right), & \beta_0^{(2)} &= \frac{1}{2} \left\{ -\frac{(\omega_0^{(2)})^2}{\nu_0^{(2)}} + \sum_{l=1}^L \tilde{d}_l (\tilde{X}_l^{(2)})^2 \right\}\end{aligned}\tag{B9}$$

To get the priors of $\bar{\mu}^{(2)}$ and $\bar{\sigma}^{(2)}$, we only need to replace $\tilde{d}_l$ with $1 - \tilde{d}_l$ in the above formulas.

- Priors of ϕ and $\bar{\phi}$:

$$\begin{aligned}
g_0 &= \sum_{i=1}^I \tilde{z}_i \tilde{Y}_i^{(3)}, & h_0 &= \sum_{i=1}^I \tilde{z}_i \\
\bar{g}_0 &= \sum_{i=1}^I (1 - \tilde{z}_i) \tilde{Y}_i^{(3)}, & \bar{h}_0 &= \sum_{i=1}^I (1 - \tilde{z}_i)
\end{aligned} \tag{B10}$$

- Prior of γ :

$$e_0 = \sum_{i=1}^I \tilde{z}_i, \quad f_0 = \sum_{i=1}^I (1 - \tilde{z}_i) \tag{B11}$$

C: Additional Details about the Experimental Design

C1. Values of Hyperparameters

For fairness of comparison, we conducted random grid search of 40 trials to tune the hyperparameters of all methods including the benchmark methods. We report the hyperparameters in Table C1, Table C2, and Table C3. Note that we regarded datasets with different IRs as different datasets and conducted separate random grid searches. The evaluation is fair as long as we repeat the same procedure for our method and the benchmark methods.

Table C1. Hyperparameters of Base Classifiers and the Benchmark Methods (Dataset: Weibo)

Method	Name of Hyperparameter	Imbalance Ratio (IR)													
		1:1	1:3	1:5	1:7	1:10	1:15	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:100
SVM	Penalty	0.61	2.41	3.01	3.01	1.21	1.81	1.21	1.81	1.81	3.61	3.01	2.41	3.01	4.81
	Kernel	L	L	L	L	L	L	L	L	L	L	L	L	R	R
CNN	Kernel Size	64	128	32	128	64	32	32	64	32	64	32	32	64	32
	Hidden Dim.	32	128	96	96	96	128	96	160	160	160	160	128	96	160
	D.P. of Input	0.4	0.4	0.4	0.3	0.4	0.25	0.4	0.3	0.35	0.4	0.4	0.3	0.3	0.4
	D.P. of Hidden	0.4	0.4	0.45	0.3	0.2	0.25	0.45	0.4	0.3	0.2	0.25	0.2	0.2	0.2
LSTM	# of Units	192	128	192	64	128	32	64	128	32	64	64	128	32	192
	Hidden Dim.	0.3	0.25	0.25	0.3	0.35	0.4	0.2	0.2	0.35	0.45	0.2	0.3	0.45	0.25
	D.P. of Input	0.25	0.45	0.45	0.4	0.35	0.25	0.4	0.3	0.35	0.2	0.3	0.2	0.25	0.35
	D.P. of LSTM	32	160	160	64	64	64	32	64	160	128	32	32	128	96
	D.P. of Hidden	0.25	0.25	0.2	0.35	0.35	0.35	0.2	0.25	0.2	0.3	0.25	0.25	0.3	0.45
Bi-LSTM	# of Units	192	192	192	32	128	32	32	64	128	32	128	32	32	32
	Hidden Dim.	0.3	0.35	0.35	0.3	0.35	0.25	0.2	0.3	0.3	0.4	0.35	0.3	0.45	0.35
	D.P. of Input	0.45	0.45	0.4	0.35	0.35	0.3	0.2	0.45	0.2	0.2	0.35	0.3	0.25	0.4
	D.P. of LSTM	96	128	128	160	64	128	128	128	96	128	128	128	128	160
	D.P. of Hidden	0.2	0.35	0.25	0.4	0.35	0.35	0.3	0.45	0.25	0.2	0.4	0.2	0.3	0.2
BERT	1 st Hidden Dim	128	128	160	160	160	160	160	160	160	128	128	160	128	128
	2 nd Hidden Dim	64	128	64	96	96	128	96	96	128	96	64	64	128	64
	D.P. of Input	0.2	0.2	0.2	0.2	0.3	0.2	0.2	0.45	0.25	0.2	0.25	0.2	0.2	0.2
	D.P. of Hidden	0.45	0.2	0.35	0.2	0.25	0.25	0.35	0.2	0.4	0.3	0.4	0.3	0.25	0.2
Concat-12	Kernel Size	192	192	128	192	192	128	64	128	128	32	128	64	128	64
	Hidden Dim.	64	64	160	96	32	160	64	160	160	64	64	32	96	32
	D.P. of Input	0.35	0.35	0.4	0.4	0.25	0.3	0.4	0.45	0.4	0.3	0.35	0.2	0.4	0.35
	D.P. of Hidden	0.35	0.25	0.4	0.25	0.25	0.45	0.4	0.35	0.4	0.2	0.35	0.3	0.45	0.35
Concat-123	Kernel Size	192	192	32	128	128	128	64	64	128	64	128	32	128	128
	Hidden Dim.	32	128	96	160	128	64	160	128	160	160	32	128	160	96
	D.P. of Input	0.3	0.4	0.45	0.45	0.25	0.2	0.35	0.35	0.35	0.35	0.25	0.35	0.35	0.3
	D.P. of Hidden	0.4	0.45	0.35	0.4	0.4	0.3	0.4	0.4	0.2	0.4	0.45	0.35	0.2	0.45
HSA-12	# of Units	128	192	128	192	128	128	192	64	64	128	64	128	64	64
	Hidden Dim.	0.4	0.3	0.2	0.25	0.2	0.3	0.3	0.3	0.2	0.4	0.3	0.3	0.4	0.35
	D.P. of Input	0.35	0.35	0.45	0.4	0.3	0.35	0.35	0.45	0.45	0.3	0.2	0.35	0.4	0.35
	D.P. of LSTM	64	32	32	64	128	128	64	128	96	128	160	32	128	128
	D.P. of Hidden	0.2	0.45	0.2	0.2	0.4	0.45	0.2	0.35	0.45	0.35	0.45	0.35	0.3	0.2
	# of Responses	15	15	15	15	15	15	15	15	15	15	15	15	15	15
HSA-123	# of Units	32	192	192	192	128	192	192	128	192	64	32	128	32	32
	Hidden Dim.	0.4	0.45	0.4	0.4	0.35	0.2	0.4	0.35	0.4	0.45	0.3	0.2	0.45	0.45
	D.P. of Input	0.2	0.4	0.35	0.35	0.2	0.3	0.25	0.45	0.35	0.45	0.2	0.3	0.35	0.25
	D.P. of LSTM	128	96	96	160	128	64	64	160	160	32	32	96	64	160
	D.P. of Hidden	0.4	0.45	0.25	0.45	0.35	0.45	0.4	0.35	0.3	0.35	0.2	0.3	0.35	0.3
	# of Responses	15	15	15	15	15	15	15	15	15	15	15	15	15	15

Note: For the kernel of SVM, L means linear kernel and R means radial basis function kernel; D.P. refers to dropout probability; Hidden Dim. means the dimension of the fully connected layer. For HSA, we fixed the number of responses to use as 15, because a large value will make the model hard and slow to train.

Table C2. Hyperparameters of Base Classifiers and the Benchmark Methods (Dataset: Twitter)

Method	Name of Hyperparameter	Imbalance Ratio (IR)			
		1:1	1:1.5	1:2	1:2.5
SVM	Penalty	1.81	0.61	0.61	1.21
	Kernel	L	L	L	L
CNN	Kernel Size	64	128	32	32
	Hidden Dim.	160	32	96	32
	D.P. of Input	0.35	0.35	0.35	0.4
	D.P. of Hidden	0.45	0.45	0.4	0.35
LSTM	# of Units	32	192	64	192
	Hidden Dim.	0.2	0.2	0.25	0.2
	D.P. of Input	0.2	0.35	0.25	0.35
	D.P. of LSTM	128	160	96	128
	D.P. of Hidden	0.3	0.25	0.25	0.25
Bi-LSTM	# of Units	64	32	192	192
	Hidden Dim.	0.35	0.3	0.4	0.45
	D.P. of Input	0.45	0.2	0.3	0.35
	D.P. of LSTM	96	128	128	128
	D.P. of Hidden	0.3	0.35	0.25	0.3
BERT	1st Hidden Dim	128	128	128	160
	2nd Hidden Dim	96	96	64	96
	D.P. of Input	0.2	0.25	0.25	0.25
	D.P. of Hidden	0.25	0.4	0.45	0.2
Concat-12	Kernel Size	64	192	32	32
	Hidden Dim.	64	128	64	160
	D.P. of Input	0.25	0.35	0.3	0.2
	D.P. of Hidden	0.2	0.25	0.25	0.2
HSA-12	# of Units	192	192	64	128
	Hidden Dim.	0.45	0.2	0.3	0.35
	D.P. of Input	0.4	0.3	0.25	0.45
	D.P. of LSTM	160	160	160	160
	D.P. of Hidden	0.35	0.25	0.35	0.2
	# of Responses	15	15	15	15

Note: Use of the abbreviations is the same as Table C1.

Table C3. Hyperparameters of the CNN Classifier in Task T2

Method	Hyperparameter Name	Dataset	
		Weibo	Twitter
CNN in Task T2	Kernel Size	64	32
	Hidden Dim.	32	64
	D.P. of Input	0.35	0.4
	D.P. of Hidden	0.25	0.35

Note: Use of the abbreviations is the same as Table C1.

D: Additional Experimental Results

D1. Sensitivity to the Dataset Split Scheme

In the first robustness check, we chronologically split the data and varied the training set percentage from 40% to 90% under the training/validation/test split scheme. Figure D1 compares the performance of CAND under two different split schemes (i.e., random split and chronological split) in both datasets. Results show that CAND performs comparably under two split schemes. The potential reason is that the patterns of false news remain similar over time. In addition, we detect false news at the event level to learn the intrinsic characteristics of false news, rather than some event-specific or trend-related patterns. Figure D2 shows the results of varying the training set percentage. In general, the maximum performance is achieved at about 80%. This is because a small training set impedes effective learning for feature-based classifiers, especially those data-hungry deep learning approaches, and a large training set leads to a small test set to run our unsupervised Bayesian results aggregation model. But overall, the performance is relatively stable against the training set percentage. We also observe that graphs in the Weibo dataset deliver smaller performance variance compared with those in the Twitter dataset. This shows that the impact of training set percentage can be further mitigated by using larger datasets.

In the second robustness check, we used k -fold cross-validation and Leave-One-Out Cross-Validation (LOOCV) instead of the training/validation/test split to evaluate the performance. We varied the value of k from 3 to 25 and finally to the size of the dataset (i.e., LOOCV¹), as shown in Figure D3. The results similarly demonstrate that our evaluation is not sensitive against the

¹ The LOOCV evaluation is too computationally expensive as it needs to repeat the model fitting for N times, where N is the number of data points. Hence, we randomly sample 500 posts to conduct LOOCV model fitting.

selection of k , even when LOOCV is used. Again, the performance of CAND in the Weibo dataset is more stable than in the Twitter dataset, suggesting that the effect of k becomes more marginal when a larger dataset is used.

Figure D1. Performance of CAND under Different Data Split Schemes (Ref. lines: AUC of CAND (R))

Note: (R) and (C) refer to random split and chronological split respectively.

(a) CAND-1 (Weibo, IR=1:7)

(b) CAND-12 (Weibo, IR=1:7)

(c) CAND-1 (Twitter, IR=1:1.5)

(d) CAND-12 (Twitter, IR=1:1.5)

Figure D2. Performance of CAND-1 and CAND-12 with Different Training Set Percentages

(a) CAND-1 (Weibo, IR=1:7)

(b) CAND-12 (Weibo, IR=1:7)

(c) CAND-1 (Twitter, IR=1:1.5)

(d) CAND-12 (Twitter, IR=1:1.5)

Figure D3. Performance of CAND-1 and CAND-12 with Different k in k -Fold Cross-Validation

Note: The error bar of LOOCV is not reported as this procedure is too computationally expensive to repeat.

D2. Effect of Number of Responses and Reports and Early Detection

This section examines the effect of the number of responses and reports on the performance of our proposed framework. First, we varied the cutoff values for the number of responses from 0 to 200 and examined the performance of CAND-12 in both datasets. The results in Figure D4 suggest that CAND-12 continues performing better before the cutoff value reaches certain small thresholds (20 for Weibo and 10 for Twitter) and stabilizes thereafter. Similarly, we varied the

maximum number of reports from 0 to 25 in the Weibo dataset² (IR=1:7). Results in Figure D5 show that human reports start playing a role even when each piece of news receives up to one report (i.e., zero or one report) and the performance improvement is the most significant before the cutoff value reaches a certain small threshold (e.g., 2 or 3). The result indicates that a small number of reports already serves as a strong signal for false news, which is actually consistent with our common sense. Overall, the results suggest that collecting more human intelligence will help to detect false news; yet, the improvement becomes quite marginal when the number of responses and reports goes beyond certain thresholds.

The above analysis also serves as a heuristic way to answer the question whether our framework enables early detection of false news because the number of responses and reports usually increases over time. The early stabilization in Figure D4 and Figure D5 means that the proposed framework is capable of detecting false news at the early stage, which is consistent with our conclusion in the Early Detection section.

² The Twitter dataset is not examined because the Twitter platform doesn't allow users to flag posts as false.

(a) Weibo (IR=1:7)

(b) Twitter (IR=1:1.5)

Figure D4. Performance of CAND-12 with Varying Cutoff Values for the Number of Responses

Note: CAND-1 (i.e., “Ref.” in the graph) is listed as a reference.

Figure D5. Performance of CAND-123 with Varying Cutoff Values for the Number of Reports (Weibo, IR=1:7)

Note: CAND-1 and CAND-12 (i.e., “Ref1” and “Ref2” in the graph) are listed as references.

D3. Sensitivity to News Type

We use the Weibo dataset to examine the sensitivity of CAND to news type. Before labeling Weibo posts based on the news type, two of the authors carefully compared the definitions from the literature with real-world Weibo posts to make sure they understood what each type of false news means. Then, they independently labeled each piece of false news. As a result, the number for each type of false news is 106, 586, 282, 145, 190, 162, and 44 respectively. The agreement rate is 84.67%, which indicates good inter-annotator consistency given that this annotation procedure is a difficult multi-label task³ (i.e., one post may fall within multiple categories). Figure D6 reports the AUC scores of CAND and the selected benchmark methods for different types of Weibo posts with the Imbalance Ratio (IR) being 1:7. When only news content and context features are available, the performance of these methods (i.e., SVM, CNN, Bi-LSTM, BERT, and CAND-1) exhibits similar distribution over the type of news. Overall, false news with false context is the easiest to identify because the conflict between the genuine content and the contextual information may provide an evident signal for recognizing false news. False news with manipulated content is hard to identify because these posts are manipulated to deceive, which may hide obvious language clues and hence fool the classifier. False news with false connection is also hard to identify because the content of images and videos is not considered in our task. When human intelligence is taken into account, the performance on all types of news increase for all methods (i.e., CAND-12, CAND-123, Concat-12, and HSA-12), and the performance distribution remains similar. This is consistent with that all types of posts have a high debunking rate. We also investigate the sensitivity of CAND to news type in finer

³ We conducted a sensitivity analysis by gradually excluding the inconsistent annotations and examining how much the inconsistent annotations impact our results. When more inconsistent annotations are excluded, AUC scores exhibit very negligible differences, suggesting that the impact of the annotator bias is small.

granularity by reporting the confusion matrix in Table D1. Examining the results leads to the same conclusion.

Figure D6. Sensitivity of CAND and the Selected Benchmark Methods to News Type
(Weibo, IR=1:7)

Note: When evaluating the performance on a specific type of post, only positive instances are available. To make the evaluation feasible, we randomly sample negative instances based on the imbalance ratio.

Table D1. Confusion Matrix of CAND with Different News Types (Weibo, IR=1:7)

		Method					
		CAND-1		CAND-12		CAND-123	
Type of False News	Satire	TN=97.00	FP=1.70	TN=97.30	FP=1.40	TN=97.90	FP=0.80
		TN=1.90	FP=12.20	TN=1.00	FP=13.10	TN=1.30	FP=12.80
	Misleading	TN=462.20	FP=8.90	TN=461.40	FP=9.70	TN=467.10	FP=4.00
		TN=8.90	FP=58.40	TN=7.10	FP=60.20	TN=5.00	FP=62.30
	Manipulated	TN=258.90	FP=2.90	TN=257.60	FP=4.20	TN=260.00	FP=1.80
		TN=5.80	FP=31.60	TN=4.60	FP=32.80	TN=3.50	FP=33.90
	False Context	TN=130.20	FP=2.80	TN=131.10	FP=1.90	TN=132.10	FP=0.90
		TN=3.10	FP=15.90	TN=2.20	FP=16.80	TN=2.60	FP=16.40
	Fabricated	TN=166.80	FP=4.00	TN=168.00	FP=2.80	TN=169.30	FP=1.50
		TN=5.00	FP=19.40	TN=4.10	FP=20.30	TN=4.20	FP=20.20
	Imposter	TN=135.90	FP=2.70	TN=136.40	FP=2.20	TN=137.70	FP=0.90
		TN=3.70	FP=16.10	TN=2.30	FP=17.50	TN=2.50	FP=17.30
	False Connection	TN=34.80	FP=0.90	TN=35.40	FP=0.30	TN=35.10	FP=0.60
		TN=0.30	FP=4.80	TN=0.20	FP=4.90	TN=0.10	FP=5.00

Note: The results are averaged over 10 runs.

D4. Robustness to Intentional Manipulation of Responses

In this section, we show how robust our approach is to maliciously manipulated crowd responses. We assume there exist six types of malicious users (listed in Table D2), who blindly manipulate the responses in a certain way. Here “blindly” means such users will manipulate the responses to one type of news indiscriminately, without conducting news-level manipulation (i.e., without manipulating news one by one). For example, true-only random users will randomly post responses on all true news with a certain probability. For each type of users listed in Table D2, we varied the percentage of randomly manipulated responses from 0% to 90% in

CAND-12 and compared its performance with that of CAND-1. The results are reported in Figure D7. Based on the AUC score, we find that CAND-12 always performs better than, or at least comparably to, CAND-1. In Figure D7a, as more responses are manipulated by indifferent users, the performance of CAND-12 decreases and is finally comparable to CAND-1. In Figure D7c and Figure D7d, the existence of false-only malicious users actually improves the performance of CAND-12; and the improvement increases as a function of the percentage of manipulated responses. In Figure D7b, Figure D7e, and Figure D7f, as more responses are manipulated, the performance of CAND-12 first decreases; then, after reaching the performance of CAND-1, it rebounds. Although the results seem to be counterintuitive, we next conduct some analytical analysis to demonstrate that they are consistent with our probabilistic framework, which explains why our method is robust to maliciously manipulated crowd responses.

Table D2. Six Types of Simulated Malicious Users

	News Types to Manipulate		
	Both false and true	False only	True only
Random user: post random responses with a certain probability	Indifferent random user	False-only random user	True-only random user
Reversal user: reverse the news debunking with a certain probability	Indifferent reversal user	False-only reversal user	True-only reversal user

(a) Indifferent Random User

(b) Indifferent Reversal User

(c) False-Only Random User

(d) False-Only Reversal User

(e) True-Only Random User

(f) True-Only Reversal User

Figure D7. Performance of CAND-12 with Various Types of Malicious Users (Weibo, IR=1:7)

Note: In x axes, “Ref.” refers to CAND-1. Its AUC score is used to draw the horizontal reference lines.

In our proposed Bayesian CAND framework, we consider different debunking rates for false and true news in the generative process. Then, the estimated difference will contribute to the estimation of the ground truth at the model inference stage. Intuitively, a larger difference will lead to better performance. For convenience, we assume the debunking rates of false and true news in the raw dataset are p_f and p_t respectively, and the percentage of the manipulated responses is r . Under each type of simulated malicious users, we compare the theoretical debunking rates of false and true news and predict the theoretical performance trend of CAND-12 based on the absolute difference between these two debunking rates. The results are presented in Table D3. Comparing them with the experimental results in Figure D7, we found that the actual trends (especially the curve of AUC score) are largely consistent with the theoretical ones. For example, in the Weibo dataset, $p_f = 26.50\%$ and $p_t = 4.46\%$. When there exist true-only reversal users, CAND-12 will achieve its theoretical minimum 0 when $r = 0.24$, which is consistent with Figure D7f.

Table D3. Theoretical Debunking Rates and Performance Trends

User Type	Debunking Rate after Simulation		Absolute Difference	Performance Trend
	False News	True News		
Indifferent random	$\frac{r}{2} + p_f - rp_f$	$\frac{r}{2} + p_t - rp_t$	$ (p_f - p_t)(1 - r) $	Decrease as r increases; minimum 0 achieved at $r = 1$
Indifferent reversal	$r + p_f - 2rp_f$	$r + p_t - 2rp_t$	$ (p_f - p_t)(1 - 2r) $	Minimum 0 achieved at $r = 0.5$
False-only random	$\frac{r}{2} + p_f - rp_f$	p_t	$ p_f - p_t + (\frac{1}{2} - p_f)r $	Decrease as r increases when $p_f < 0.5$
False-only Reversal	$r + p_f - 2rp_f$	p_t	$ p_f - p_t + (1 - 2p_f)r $	Decrease as r increases when $p_f < 0.5$
True-only random	p_f	$\frac{r}{2} + p_t - rp_t$	$ p_f - p_t - (\frac{1}{2} - p_t)r $	Minimum 0 achieved at $r = \frac{p_f - p_t}{0.5 - p_t}$ when $p_t < 0.5$
True-only reversal	p_f	$r + p_t - 2rp_t$	$ p_f - p_t - (1 - 2p_t)r $	Minimum 0 achieved at $r = \frac{p_f - p_t}{1 - 2p_t}$ when $p_t < 0.5$

In summary, our analysis shows that our probabilistic framework is robust to several types of adversarial attacks, and integrating crowd responses will contribute to, or at least not jeopardize, the false news detection task.

Finally, in the era of deep fakes and adversarial methods, we think it is important and interesting to further explore this issue in the future. In the current analysis, we simulated several types of malicious users, which might not cover all actual users in real-world social media platforms. It would deliver significant value if we could fight false news even when malicious users manipulate responses or reports of news one by one.

D5. Debunking Response Detection as a Multi-Class Classification Problem

In this section, we explore whether formulating the debunking response detection as a multi-class classification problem will contribute to our proposed framework. Under such formulation, the responses are classified into four categories: supporting, denying, commenting, and querying (Kochkina et al. 2018; Ma et al. 2018; Sharma et al. 2019). In this scenario, we are unable to directly extend the proposed CAND framework to the multi-class setting to utilize the multi-class labels because the logistic-normal distribution cannot be naturally generalized to its multivariate form. Actually, its natural generalization is softmax-normal distribution, which yet does not make sense. Hence, we developed two alternative models that tap into the multi-class information. As these two models share many generative processes with CLNAM, we only introduce the parts that differ.

Using the same set of notations as CLNAM (see Figure 4 in the manuscript), in the multi-class scenario, $d_{n,k}$, i.e., stance of response k to news n , takes the value of 1 (supporting), 2

(denying), 3 (commenting), or 4 (querying); and $\mathbf{Y}_{n,k}^{(2)} \in [0,1]^4$ is a 3-simplex⁴, meaning the predicted probability of $d_{n,k}$ equaling each possible value.

The first model is called Categorical Aggregation Model (CAM). We transform $\mathbf{Y}_{n,k}^{(2)}$ into integers and assume the transformed values follow a categorical distribution. Specifically, given the ground truth of news n , the stance of response k follows

$$d_{n,k} \sim \text{Categorical}(z_n \eta + (1 - z_n) \bar{\eta}) \quad . \quad (\text{D1})$$

Then, the transformed machine prediction is generated based on $d_{n,k}$:

$$\arg \max(\mathbf{Y}_{n,k}^{(2)}) \mid d_{n,k}, \boldsymbol{\mu}^{(2)} \sim \text{Categorical}\left(\sum_{i=1}^4 \mathbb{1}(d_{n,k} = i) \boldsymbol{\mu}_i^{(2)}\right) \quad , \quad (\text{D2})$$

where $\arg \max(\mathbf{x})$ equals the index of the largest value in $\mathbf{x}$ wherein $\mathbf{x}$ is a 3-simplex. Finally, we impose conjugate priors, i.e., Dirichlet distribution, over η , $\bar{\eta}$, and $\boldsymbol{\mu}_i^{(2)}$.

In the CAM model, we did not take advantage of the logistic-normal assumption, which has shown to be effective in the previous experiments. Further, to simultaneously maintain the benefits of the logistic-normal assumption and the multi-class information, we propose another model called Extended CAM (E-CAM). Specifically, given the ground truth of news n , the stance of responses k follows the same distribution as CAM (see Equation E1). Unlike CAM, where $\mathbf{Y}_{n,k}^{(2)}$ is transformed into four integer values, E-CAM transforms it to a continuous value between 0 and 1, which indicates debunking probability. To this end, we treat denying responses

⁴ An n -simplex is a vector of $n+1$ positive continuous values that sum up to 1.

and querying responses as debunking responses. Then, we pose a logistic-normal distribution on the debunking probability:

$$Y_{n,k,2}^{(2)} + Y_{n,k,4}^{(2)} \mid d_{n,k}, \boldsymbol{\mu}^{(2)}, \boldsymbol{\sigma}^{(2)} \sim \text{LogisticNormal}\left(\sum_{i=1}^4 \mathbb{1}(d_{n,k} = i) \mu_i^{(2)}, \sum_{i=1}^4 \mathbb{1}(d_{n,k} = i) (\sigma_i^{(2)})^2\right). \quad (\text{D3})$$

In the E-CAM model, we see that in order to take advantage of the logistic-normal assumption, the multi-class information is only partially exploited.

Two authors relabeled the responses with labels supporting, denying, commenting, and querying in both datasets (i.e., Weibo and Twitter). Following the same experimental procedure as CAND, we ran the CAM and E-CAM methods and compared their performance with CAND. The plots are shown in Figure D8. The results in both datasets indicate that CAND significantly outperforms CAM. This is because in CAM, $\mathbf{Y}_{n,k}^{(2)}$ is transformed into integer values and assumed to follow a categorical distribution, rather than the natural multivariate generalization of logistic-normal distribution (which doesn't exist). Given that categorical distribution is the multivariate generalization of Bernoulli distribution and the poor performance of BAM in our experiment (see the experimental results in the paper), it is as expected that CAND performs better than CAM. By virtue of the logistic-normal distribution, the extended model E-CAM performs comparably to CAND. Such results suggest the importance of the logistic-normal assumption in our task. In addition, although the literature has shown the benefit of multi-class formulation (Kochkina et al. 2018; Ma et al. 2018), the performance gain from treating debunking response detection as a multi-class classification problem in our scenario is quite marginal. There are two potential reasons. First, in order to maintain the logistic-normal assumption, E-CAM only partially exploits the multi-class information. Although this assumption cannot be naturally

generalized to its multivariate form, it would be interesting to explore whether there exist other alternatives in the future. Second, our task formulation is different from the one in the literature. Specifically, in the literature, stance classification and false news detection are jointly trained under a multi-task learning framework, which is able to take advantage of the shared task-invariant features (Ma et al. 2018). The results of the stance classification are not directly utilized by the task of false news detection. However, in our proposed approach, we use the results of stance classification as the input for false news detection, rather than training them jointly. As the most essential and relevant labeling to recognize false news is whether each response is debunking or not, multi-class information might not contribute in our current formulation. Note that, responses with the label “supporting” also have strong predictive power for true news. However, in our datasets, Weibo and Twitter, the rates of “supporting” responses are 2.02% and 4.62% respectively, which are too low to take effect. In the future, we can try to simultaneously take advantage of the joint training (what was done in the literature) and feeding the results of stance detection as the input of false news detection (what was done in this work) to further improve the performance.

(a) Weibo, IR=1:7

(b) Twitter, IR=1:1.5

Figure D8. Performance of CAND in Comparison with CAM and E-CAM (Ref. lines: AUC of CAND)

D6. Dataset Bias

Annotator Bias: Annotator bias refers to the data bias in labels caused by different knowledge, capability, and subjective perception of annotators in regards to the tasks (Wich et al. 2020).

Mitigating the annotator bias usually reduces to the task of collecting high-quality annotations from crowd workers. Two widely-used solutions are active label acquisition and annotation aggregation (Wang et al. 2017; Wei et al. 2020). The first approach actively selects the best instance to be labeled by crowd workers so as to obtain high-quality annotations (Gao and Saar-Tsechansky 2020; Song et al. 2018); the second approach focuses on how to aggregate the collected annotations by considering various factors such as worker reliability and task difficulty (Kim and Ghahramani 2012; Moreno et al. 2015; Wang et al. 2017).

To investigate the annotator bias of the debunking response annotation, we examine the performance of the debunking response detection algorithm for Task T2 and the performance of CAND-12 for false news detection by varying the number of annotators from two to four in the

Weibo dataset. When the number of annotators is two, a consensus is reached for inconsistent annotations after discussion. When the number is three or four, the annotations are aggregated using the iBCC model, which is an effective crowdsourced answer aggregation model, as introduced in the Related Work section. These three annotation schemes are denoted as S1, S2, and S3 respectively. Figure D9a suggests that more annotators will significantly increase the performance of debunking response detection. Note that, although S1 has fewer annotators than S2, the debunking responses detection algorithm under S1 performs better. This is because the annotators reached a consensus for all inconsistent annotations under scheme S1. Given that debunking response detection is an important building block of our CAND framework, we expect CAND-12 under S3 would perform significantly better than the cases of S1 and S2. However, the result in Figure D9b is counter-intuitive—the performance gap is quite marginal. This can be explained by how our CAND framework utilizes the results of Task T2. In our proposed framework, the debunking response detection algorithm in Task T2 will predict a probability as to whether the response is debunking for each post-response pair. With the predicted debunking probability, the CAND framework assumes that the distribution of debunking probability is different for false and true news. It means that CAND relies more on the overall distribution, rather than the specific predicted probability. A less accurate classifier in Task T2 may impact the distribution of debunking probability for both false and true news. However, it will not impact the performance of CAND as long as these two distributions are still distinguishable. This experiment suggests that our CAND framework is robust against the accuracy of Task T2.

(a) Debunking Response Detection

(b) CAND-12

Figure D9. Performance of T2 and CAND-12 with Varying Number of Annotators (Weibo, IR=1:7)

Selection Bias: Selection bias arises from the non-random sampling of subgroups in which case the selected dataset is not representative enough of real-world applications (Mehrabi et al. 2019). In the Weibo dataset, the false posts were collected from the Sina community management center and the true posts were retrieved from general threads, following the method in the previous literature (Ma et al. 2016). The Twitter dataset is less manipulatable as it was constructed based on three reference datasets of the literature. Limited by the resources, we are unable to collect all social media posts or a random sample of them to show that our dataset is an unbiased sample of the real-world posts. However, our research design still mitigates the selection bias in the following ways. First, we collected all false Weibo posts announced in the Sina community management center. Such a set will not be biased as long as the announced false Weibo posts are representative of all false news. In addition, we detect false news at the event level to learn the intrinsic characteristics of false news and prevent the model from learning the news event-specific features. Last, although the true ratio between false and true news is unknown, we varied the Imbalance Ratio to comprehensively evaluate the proposed approach.

References

- Burton-Jones, A., Boh, W. F., Oborn, E., and Padmanabhan, B. 2021. “Editor’s Comments: Advancing Research Transparency at MIS Quarterly: A Pluralistic Approach,” *Management Information Systems Quarterly* (45:2), pp. iii–xviii.
- Gao, R., and Saar-Tsechansky, M. 2020. “Cost-Accuracy Aware Adaptive Labeling for Active Learning,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 2569–2576. (<https://doi.org/10.1609/aaai.v34i03.5640>).
- Kim, H.-C., and Ghahramani, Z. 2012. “Bayesian Classifier Combination,” in *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, pp. 619–627.
- Kochkina, E., Liakata, M., and Zubiaga, A. 2018. “All-in-One: Multi-Task Learning for Rumour Verification,” in *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 3402–3413.
- Kumar, S., and Shah, N. 2018. “False Information on Web and Social Media: A Survey,” *ArXiv:1804.08559 [Cs]*. (<http://arxiv.org/abs/1804.08559>).
- Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B. J., Wong, K.-F., and Cha, M. 2016. “Detecting Rumors from Microblogs with Recurrent Neural Networks,” in *25th International Joint Conference on Artificial Intelligence*, pp. 3818–3824.
- Ma, J., Gao, W., and Wong, K.-F. 2018. “Detect Rumor and Stance Jointly by Neural Multi-Task Learning,” in *Companion Proceedings of the The Web Conference 2018*, Switzerland: International World Wide Web Conferences Steering Committee, pp. 585–593. (<https://doi.org/10.1145/3184558.3188729>).
- Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. 2019. “A Survey on Bias and Fairness in Machine Learning,” *ArXiv:1908.09635 [Cs]*. (<http://arxiv.org/abs/1908.09635>).
- Moreno, P. G., Artés-Rodríguez, A., Teh, Y. W., and Perez-Cruz, F. 2015. “Bayesian Nonparametric Crowdsourcing,” *Journal of Machine Learning Research* (16), pp. 1607–1627.
- Sharma, K., Qian, F., Jiang, H., Ruchansky, N., Zhang, M., and Liu, Y. 2019. “Combating Fake News: A Survey on Identification and Mitigation Techniques,” *ACM Transactions on Intelligent Systems and Technology* (10:3), pp. 1–42. (<https://doi.org/10.1145/3305260>).
- Song, J., Wang, H., Gao, Y., and An, B. 2018. “Active Learning with Confidence-Based Answers for Crowdsourcing Labeling Tasks,” *Knowledge-Based Systems* (159), pp. 244–258. (<https://doi.org/10.1016/j.knosys.2018.07.010>).
- Vosoughi, S., Roy, D., and Aral, S. 2018. “The Spread of True and False News Online,” *Science* (359:6380), pp. 1146–1151. (<https://doi.org/10.1126/science.aap9559>).

- Wang, J., Ipeirotis, P. G., and Provost, F. 2017. “Cost-Effective Quality Assurance in Crowd Labeling,” *Information Systems Research* (28:1), pp. 137–158. (<https://doi.org/10.1287/isre.2016.0661>).
- Wardle, C. 2017. *Fake News. It’s Complicated*. (<https://firstdraftnews.org/latest/fake-news-complicated/>).
- Wei, X., Zhang, M., and Zeng, D. D. 2020. “Learning from Crowd Labeling with Semi-Crowdsourced Deep Generative Models,” in *CCF Conference on Computer Supported Cooperative Work and Social Computing*, Springer, pp. 101–114.
- Wei, X., Zhang, Z., Zhang, M., Chen, W., and Zeng, D. D. 2022. “Combining Crowd and Machine Intelligence to Detect False News on Social Media,” *Management Information Systems Quarterly* (46:2), pp. 977-1008.
- Wich, M., Al Kuwatly, H., and Groh, G. 2020. “Investigating Annotator Bias with a Graph-Based Approach,” in *Proceedings of the Fourth Workshop on Online Abuse and Harms*, Online: Association for Computational Linguistics, November, pp. 191–199. (<https://doi.org/10.18653/v1/2020.alw-1.22>).
- Zhou, X., and Zafarani, R. 2020. “A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities,” *ACM Computing Surveys* (53:5), 109:1-109:40. (<https://doi.org/10.1145/3395046>).