Gleaning Wisdom from the Past: Early Detection of Emerging Rumors in Social Media

Liang Wu* Jundong Li* Xia Hu† Huan Liu*

Abstract

The explosive use of social media, in information dissemination and communication, has also made it a popular platform for the spread of rumors. Rumors could be easily propagated and received by a large number of users in social media, resulting in catastrophic effects in the physical world in a very short period. It is a challenging task, if not impossible, to apply classical supervised learning methods to the early detection of rumors, since the labeling process is time-consuming and labor-intensive. Motivated by the fact that abundant label information of historical rumors is publicly available, in this paper, we propose to investigate whether knowledge learned from historical data could potentially help identify newly emerging rumors. In particular, since a disputed factual claim arouses certain reactions such as curiosity, skepticism, and astonishment, we identify and utilize patterns from prior labeled data to help reveal emergent rumors. Experimental results on real-world data sets demonstrate the effectiveness. Further experiments are conducted to show how much earlier it can detect an emerging rumor than traditional approaches.

1 Introduction

The prevalence of social media has revolutionized the way of information dissemination and communication. The openness of social media platforms enables timely information to be spread at a high rate. Meanwhile, it also allows for the rapid creation and dissemination of rumors, which could cause catastrophic effects in the real world within a short period. For example, on April 23rd 2013, the hacked Twitter account of Associate Press posted a false claim of an attack on the White House, which was soon covered by news agencies, and wiped out $136 billion in the stock market within two minutes[1]. It would be appealing if emerging rumors could be automatically detected in its early stage.

Classical rumor detection methods highly depend on learning patterns from manually labeled data. A straightforward way is to learn a classifier or regressor based on labeled rumors, and then the built model can be employed to determine the credibility of a new message or user. However, in real-world applications, annotating a rumor dataset could be time-consuming and labor-intensive, sometimes even impractical. The labeling bottleneck brings in an unavoidable delay for existing systems, resulting in significant challenges to enable the system to detect new rumors in a timely manner. Therefore, it would be desirable to develop a way for rumor detection without the labeling process.

While the problem of detecting rumors on social media is relatively new, rumors have been extensively investigated for years in social and psychological studies. The literature can be traced back to [2]. A conventional methodology of studying rumors is to analyze the testimonies. The origins, consequences and potential impact of a rumor can be well estimated by linking it to a historical rumor through examining the behaviors of social participants who are exposed to it [3, 20], since *similar* rumors usually trigger similar reactions, such as curiosity, inquiry, and anxiety. Although the content on social networks is informal, its significant role in understanding a rumor has been found [17]. Motivated by the previous findings, we explore the possibility of using the abundant labeled data from prior rumors to facilitate the detection of an emerging rumor.

However, it is particularly difficult and challenging to directly use labeled data from one rumor to build a detection model for the other, *a.k.a.* cross-training. Cross-training can be successfully applied to problems of which different tasks are similar. Since rumor data is highly topic-sensitive, the vocabulary and word choice may vary substantially between different rumors. Therefore, directly applying an existing dataset would lead to the inclusion of noisy features and thus may negatively inhibit the prediction accuracy. In addition, since a certain category of rumors may trigger specific reactions, *e.g.,* wedge-driving rumors cause hatred and atrocity rumors arouse astonishment, it is ideal to find useful patterns within a category. Due to the lack of availability of the category information, it is difficult to find the scarce patterns out of miscellaneous labeled data. Also, since social media users tend to communicate concisely and casually [10], the short content may further exacerbate the scarcity problem.

*Computer Science and Engineering, Arizona State University, Tempe, AZ, USA. {wuliang, jundongl, huan.liu}@asu.edu

†Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA. xiahu@tamu.edu

[1] http://www.bloomberg.com/news/articles/2013-04-23/dow-jones-drops-recovers-after-false-report-on-ap-twitter-page

Table 1: Two real-world examples of social media rumors.

(a) An example rumor about the Presidential election and the corresponding social media posts.

Rumor	Rightwing Christian says elect #trump or face #deathcamps run by #liberals http://bit.ly/2as5MJ5 .
Post #1	Christian conservative gets political. Can't fix stupid but it can be blocked.
Post #2	So, when did bearing false witness become a Christian value?
Post #3	Graham Says Christians Must Support Trump or Face Death Camps. Does he still claim to be a Christian?

(b) An example rumor about the Ferguson protests and the corresponding social media posts.

Rumor	A Ferguson protesting sign reads 'No Mother Should Fear for Her Son's Life Every Time He Robs a Store.'
Post #1	i've just seen the sign on fb. you can't fix stupid.
Post #2	THIS IS PURE INSANITY.. HOW ABOUT THIS STATEMENT.
Post #3	No Mother Should Have To Fear For Her Son's Life Every Time He Robs A Store #AllLivesMatt

In order to tackle the aforementioned challenges, we present a novel learning framework to detect emerging rumors with existing labeled data from prior rumors. The proposed framework is built upon a sparse representation model, and it jointly selects descriptive features from prior labeled data and trains the topic-independent classifier with selected features. The proposed framework extends the earliness bottleneck of current rumor detection methods. Our contributions are summarized below:

- Providing a principled way to leverage prior labeled data to detect emerging rumors;
- Proposing a novel sparse learning method to jointly select features and train the classifier for rumors;
- Evaluating the proposed framework extensively using real-world social media data.

The rest of the paper is organized as follows. We introduce the main intuition and formulate the problem in Section 2. The proposed framework is presented in Section 3. Section 4 discusses the experiments and the results on effectiveness and earliness. Section 5 introduces the relationship to existing work. Section 6 concludes the paper and points out future directions.

2 Emerging Rumor Detection Framework

In this section, we introduce our motivation of leveraging labeled data from prior rumors. We also briefly introduce the proposed framework, and formally define the computational problem.

2.1 Motivation In Table 1, we display rumors about two topics. The first rumor in Table 1(a) is about endorsements for the presidential candidate. The rumor says a famous evangelist urged Christians to vote for Donald Trump, otherwise they will face death camps. The following three sentences are posts of the rumor. The second rumor in Table 1(b) is about a Ferguson protester. The rumor says the sign that the protesters are holding reads "No Mother Should Fear for Her Son's Life Every Time He Robs a Store". The bias of word choice of different rumor topics makes it difficult for cross-training. For example, the classifiers trained on the first rumor, which use features such as "political" and "Christian" would be useless in identifying posts of the second rumor.

In the literature of social and psychological studies, both rumors can be categorized as wedge-driving rumors [2] that feed on hate. In the user posts, we find contents that express hostility similarly, such as "fix stupid" and "pure insanity". These similar expressions are useful in identifying future wedge-driving rumors, which may or may not be related to the two topics. Therefore, we aim to discover the topic-independent patterns in user posts.

2.2 Working of the Framework In order to build the framework that can exploit prior labeled data, two main issues remain to be solved. An ideal case for selecting topic-independent features is that we group rumors by their categories and find discriminative features for each category, such as hatred features for wedge-driving rumors, worrying features for anxiety-arising rumors, and astonishment features for atrocity rumors. However, rumor categories are unavailable. In order to solve the problem, we adopt structure learning-based feature selection in this work. Motivated by recent research on unsupervised feature selection [11], for an unlabeled dataset, we can effectively select features by preserving the intrinsic structure of data. In our work, the structure is the rumor category, and within the same category, rumors trigger similar contents.

As conventional practices in unsupervised feature selection approaches, the selected features can then be used for training a classifier. However, the supervised information, *i.e.,* the rumor labels, has not been considered in the feature selection process, which leads to

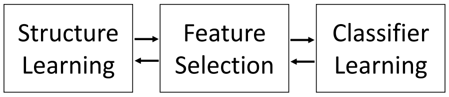

Figure 1: An illustration of the learning procedure of the proposed framework. The framework consists of three components: inferring rumor categories (structure learning), selecting discriminative features, and learning the rumor classifier.

the issue that the selected features may fail to capture the key knowledge of rumors. A more coherent method is to integrate the feature selection and classification processes into a unified framework.

Figure 1 illustrates the three components of the proposed framework. The framework is built upon sparse representation learning methods, which simultaneously infers the category structure of rumor data and selects discriminative features. The rumor label is also jointly utilized by supervising the feature selection process which results in an optimal rumor classifier.

2.3 Problem Definition $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_m\} \in \mathbb{R}^{m \times n}$ is the data matrix with each row $\mathbf{d}_i \in \mathbb{R}^n$ being a data instance and each column $\mathbf{f}_i \in \mathbb{R}^m$ being a vector of each feature. $\mathbf{y} \in \{-1, 1\}^m$ is the label vector for training data. $\mathbf{y}_i = 1$ if i refers to a rumor, and otherwise, $\mathbf{y}_i = -1$. Given the data matrix $\mathbf{D}$, label vector $\mathbf{y}$, we aim to learn a predictor that accurately classifies rumors and non-rumors based on the social media posts.

3 Cross-Training for Emerging Rumor Detection

In this section, we first introduce how we select features to allow for the cross-training. As mentioned earlier, the two subtasks are (1) clustering data according to rumor categories and (2) selecting features that are effective across topics within a category.

3.1 Proposed Framework Motivated by recent research on feature selection [11], we start with a matrix factorization formulation:

$$\min_{\mathbf{U},\mathbf{V}} \frac{1}{2}||\mathbf{D} - \mathbf{U}\mathbf{V}^T||_F^2,$$

where $||\cdot||_F$ denotes the *Frobenius* norm. The original data matrix is decomposed into two factors, $\mathbf{U} \in \mathbb{R}^{m \times k}$ is the low-rank representation of users, and $\mathbf{V} \in \mathbb{R}^{n \times k}$ is the low-rank representation of features with $k \ll n$. The factorization separates data from feature by k latent factors, which enables the clustering and feature selection to be jointly performed. In order to force the user factor $\mathbf{U}$ to be cluster indicators instead of latent factors, we impose a constraint on $\mathbf{U}$:

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V}} \frac{1}{2}||\mathbf{D} - \mathbf{U}\mathbf{V}^T||_F^2, \\ &s.t. \quad \mathbf{U} \in \{0,1\}^{m \times k}, \mathbf{U}\mathbf{1} = \mathbf{1} \end{aligned} \tag{3.1}$$

where $\mathbf{1}$ is a vector with all elements equal to 1. The m rows are then clustered into k clusters. However, due to the constraint on $\mathbf{U}$, it is difficult to solve the problem in Eq.(3.1). Motivated by research on spectral clustering [27], we introduce an orthogonal constraint on the rows to relax it. Eq.(3.1) can then be rewritten as follows:

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V}} \frac{1}{2}||\mathbf{D} - \mathbf{U}\mathbf{V}^T||_F^2, \\ &s.t. \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{U} \geq \mathbf{0} \end{aligned} \tag{3.2}$$

where $\mathbf{I}$ is an identity matrix and thus rows in $\mathbf{U}$ are orthogonal to each other.

The orthogonal constraint ensures that data instances are clustered into different rumor categories. For each rumor category, we aim to select descriptive features. To this end, we try to select key features while force the unselected features to be zero. In the literature of sparse learning and feature selection, it can be done by imposing an $\ell_{2,1}$-norm [9]. Motivated by recent studies on embedded feature selection [28], we rewrite Eq.(3.2) as follows:

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V}} \frac{1}{2}||\mathbf{D} - \mathbf{U}\mathbf{V}^T||_F^2 + \alpha||\mathbf{V}||_{2,1}, \\ &s.t. \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{U} \geq \mathbf{0} \end{aligned} \tag{3.3}$$

where the $\ell_{2,1}$-norm regularizer selects features that best preserve the structure of clustering $\mathbf{U}$. α controls the extent of sparsity.

Through solving Eq.(3.3), we can obtain the low-rank representations. However, the labeled data that are available for distinguishing rumor and non-rumor content has not been exploited. The resultant representation would fail to capture the key signal that reveals the appearance of rumors of a category. Motivated by Collective Matrix Factorization-based relational learning [24, 30], we introduce a classification loss term in the objective function. We adopt the hinge loss used in Support Vector Machines (SVMs), and Eq.(3.3) is reformulated as:

$$\begin{aligned} &\min_{\mathbf{U},\mathbf{V},\mathbf{w}} \frac{1}{2}||\mathbf{D} - \mathbf{U}\mathbf{V}^T||_F^2 + \alpha||\mathbf{V}||_{2,1} + \beta\sum_{i=1}^{m} h(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i), \\ &s.t. \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{U} \geq \mathbf{0} \end{aligned} \tag{3.4}$$

where $h(\cdot)$ is the hinge loss and β controls the extent that the training information influences the feature selection and structure learning processes. $\mathbf{u}_i\mathbf{V}^T$ is the reconstructed formulation of a data instance. $\mathbf{w}$ is the model parameter of the SVMs, and $\mathbf{u}_i\mathbf{V}^T\mathbf{w}$ denotes the prediction with given low-rank representations. To make it convenient for optimization, we adopt the smoothed hinge loss [19] for $h(\cdot)$ as follows:

$$h(\theta) = \begin{cases} \frac{1}{2} - \theta & \theta \leq 0 \\ \frac{1}{2}(1-\theta)^2 & 0 < \theta < 1 \\ 0 & \theta \geq 1 \end{cases}$$

where the loss function is smoothed when $\theta = 1$, and the corresponding optimization task of computing its gradient is more tractable. The gradient of the smoothed hinge loss is

$$h'(\theta) = \begin{cases} -1 & \theta \leq 0 \\ \theta - 1 & 0 < \theta < 1 \\ 0 & \theta \geq 1 \end{cases} \tag{3.5}$$

Next, we will introduce how to optimize the objective function in Eq.(3.4) efficiently.

3.2 Optimization

The objective function in Eq.(3.4) is not convex *w.r.t.* all three variables, *i.e.,* $\mathbf{U}, \mathbf{V}$, and $\mathbf{w}$. However, Eq.(3.4) is convex in each of the three variables separately. Hence, we update each of them by fixing the other two iteratively.

3.2.1 Modeling Rumor Category

First, we introduce how $\mathbf{U}$ can be updated by fixing $\mathbf{V}$ and $\mathbf{w}$. By removing terms that are irrelevant to $\mathbf{U}$, Eq.(3.4) can be reformulated as follows:

$$\begin{aligned} &\min_{\mathbf{U}} \frac{1}{2}||\mathbf{D} - \mathbf{U}\mathbf{V}^T||_F^2 + \beta \sum_{i=1}^{m} h(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i). \\ &s.t. \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{U} \geq \mathbf{0} \end{aligned} \tag{3.6}$$

The problem in Eq.(3.6) is an orthogonality constrained optimization problem. The problem can be solved in the Crank-Nicolson scheme. Following [29], $\mathbf{U}$ can be efficiently updated as follows:

$$\mathbf{U} \leftarrow (\mathbf{I} + \frac{\tau}{2}\mathbf{Q})^{-1}(\mathbf{I} - \frac{\tau}{2}\mathbf{Q})\mathbf{U}, \tag{3.7}$$

where τ is the step size and $\mathbf{Q}$ is a skew-symmetric matrix, which leads to the descent along geodesics and inside the feasible set. $\mathbf{Q}$ can be constructed as

$$\begin{aligned} \mathbf{Q} &= [\mathbf{U}, \mathbf{G}][\mathbf{G}, -\mathbf{U}]^T \\ &= \mathbf{U}\mathbf{G}^T - \mathbf{G}\mathbf{U}^T, \end{aligned} \tag{3.8}$$

where $\mathbf{G}$ is the gradient of the optimization objective in Eq.(3.6). Since both terms in Eq.(3.6) are convex, the gradient can be obtained with Eq.(3.5) as

$$\mathbf{G}_{i,j} = [\mathbf{U}\mathbf{V}^T\mathbf{V} - \mathbf{D}\mathbf{V}]_{i,j} + \beta[h'(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i)\mathbf{y}_i\mathbf{w}^T\mathbf{V}]_j,$$

where $[\cdot]_{i,j}$ is the (i,j) entry of the matrix and $[\cdot]_j$ is the j^{th} entry of the vector. A problem of directly updating Eq.(3.7) is that the time complexity is high, since the inverse operation dominates the calculation when m is large. In order to solve the problem, we rewrite the objective function as follows by applying the SMW formula [23, 29]:

$$\begin{aligned} &(\mathbf{I} + \frac{\tau}{2}\mathbf{Q})^{-1}(\mathbf{I} - \frac{\tau}{2}\mathbf{Q})\mathbf{U} \\ &= (\mathbf{I} + \frac{\tau}{2}[\mathbf{U}, \mathbf{G}][\mathbf{G}, -\mathbf{U}]^T)^{-1}(\mathbf{I} - \frac{\tau}{2}[\mathbf{U}, \mathbf{G}][\mathbf{G}, -\mathbf{U}]^T)\mathbf{U} \\ &= \mathbf{U} - \tau[\mathbf{U}, \mathbf{G}](\mathbf{I} + \frac{\tau}{2}[\mathbf{G}, -\mathbf{U}]^T[\mathbf{U}, \mathbf{G}])^{-1}[\mathbf{G}, -\mathbf{U}]^T\mathbf{U} \end{aligned} \tag{3.9}$$

By reformulating the objective function in Eq.(3.7), only the inverse of $(\mathbf{I} + \frac{\tau}{2}[\mathbf{G}, -\mathbf{U}]^T[\mathbf{U}, \mathbf{G}])$ needs to be calculated, which takes $O(k^3)$. Since k is the number of clusters and normally $k \ll n$ and $k \ll m$, the inverse operation is much easier to solve and no longer dominates the computation.

In order to find the optimal step size τ in Eq.(3.7), we first introduce the Armijo-Wolfe condition [8]

$$\begin{aligned} \mathcal{L}(\mathbf{U}_\tau) &\leq \mathcal{L}(\mathbf{U}_{\tau=0}) + \rho_1\tau\mathcal{L}'(\mathbf{U}_\tau), \\ \mathcal{L}'(\mathbf{U}_\tau) &\geq \rho_2\mathcal{L}'(\mathbf{U}_{\tau=0}), \end{aligned} \tag{3.10}$$

where $\mathbf{U}_\tau$ is the trial point of gradient descent given a specific τ, and $\mathbf{U}_{\tau=0}$ is the value by setting τ to zero. ρ_1 and ρ_2 are two parameters satisfying that $0 < \rho_1 < \rho_2 < 1$ [13]. $\mathcal{L}(\cdot)$ is the loss function in Eq.(3.6), and $\mathcal{L}'(\cdot)$ is its gradient.

The optimal value of τ can be obtained through curvilinear search [4] with Armijo-Wolfe condition in Eq.(3.10), and details are presented in Algorithm 1.

Algorithm 1 Curvilinear Search for τ

```
1: Initialize $\tau > 0$
2: Until Eq.(3.10) is satisfied
3:     Set $\tau \leftarrow \frac{\tau}{2}$
4: Return $\tau$
```

3.2.2 Selecting Features

Now we are introducing how $\mathbf{V}$ can be updated given fixed $\mathbf{U}$ and $\mathbf{w}$. The optimization function of $\mathbf{V}$ can be formulated based on Eq.(3.4) as

$$\min_{\mathbf{V}} \frac{1}{2}||\mathbf{D} - \mathbf{U}\mathbf{V}^T||_F^2 + \alpha||\mathbf{V}||_{2,1} + \beta \sum_{i=1}^{m} h(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i), \tag{3.11}$$

where constraints on $\mathbf{U}$ are removed. The objective function in Eq.(3.11) is similar to that of multi-task feature selection [16]. The update rule for $\mathbf{V}$ can be obtained by taking the derivative and setting it to zero. The derivative can be formulated as

$$\mathbf{V} - \mathbf{D}^T\mathbf{U} + \alpha\mathbf{C}\mathbf{V} + \beta\sum_{i=1}^{m}(\mathbf{y}_i h'(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i))\mathbf{w}\mathbf{u}_i, \tag{3.12}$$

where $\mathbf{C}$ is a diagonal matrix where $\mathbf{C}_{i,i} = \frac{1}{2||\mathbf{v}_i||_2}$. $\mathbf{C}$ is constructed to obtain the derivative of the $\ell_{2,1}$ regularization term of $\mathbf{V}$ [25]. By setting Eq.(3.12) to zero, the update rule of $\mathbf{V}$ can be written as:

$$\mathbf{V} \leftarrow (\mathbf{I} + \alpha\mathbf{C})^{-1}(\mathbf{D}^T\mathbf{U} - \beta\sum_{i=1}^{m}(\mathbf{y}_i h'(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i))\mathbf{w}\mathbf{u}_i). \tag{3.13}$$

3.2.3 Learning Rumor Classifier Finally, we will introduce how the rumor classifier can be obtained given fixed $\mathbf{U}$ and $\mathbf{V}$. By removing terms that are irrelevant to $\mathbf{w}$, Eq.(3.4) can be rewritten as

$$\min_{\mathbf{w}} \beta\sum_{i=1}^{m} h(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i) + \frac{\gamma}{2}||\mathbf{w}||_2^2,$$

where we add a regularization term to avoid over-fitting, and γ to control the complexity of $\mathbf{w}$. Since both terms are smooth and convex, the update rule of $\mathbf{w}$ can be written as

$$\mathbf{w} \leftarrow \mathbf{w} - \eta\big(\beta(\mathbf{y}_i h'(\mathbf{u}_i\mathbf{V}^T\mathbf{w}\mathbf{y}_i))\mathbf{V}\mathbf{u}_i^T\big), \tag{3.14}$$

where η is the step size and can be efficiently estimated with backtracking line search [15].

3.2.4 Analysis Given update rules of $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{w}$, the problem can be efficiently solved by a Stochastic Gradient Descent algorithm (SGD). SGD solves the optimization problem in the hill-climbing scheme by seeking the stationary point. The optimization process can be found in Algorithm 2. $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{w}$ are updated alternatively from line 3 to line 5. Since the objective function decreases for each of the subproblems, and Eq.(3.4) has lower bounds such as zero, Algorithm 2 converges. As mentioned earlier, the inverse operation in Eq.(3.9) can be quickly done in $O(k^3)$. The inverse operation in Eq.(3.13) can be solved in $O(n)$ since $(\mathbf{I}+\alpha\mathbf{C})$ is a diagonal matrix. Therefore, the complexity of one iteration (lines 3-5) is dominated by the matrix multiplication, which can be efficiently solved since the data matrix $\mathbf{D}$ obtained from social media contents is usually sparse. In addition, the experimental results on our datasets show that the algorithm often converges in less than 20 iterations.

Algorithm 2 Early Detection of Emerging Rumors

```
Input: Data matrix $\mathbf{D}$, label vector $\mathbf{y}$, maximal
  number of iterations $I$
Output: $\mathbf{U}$, $\mathbf{V}$, $\mathbf{w}$
 1: Generate $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{w}$ randomly
 2: For $i$=1 to $I$ do
 3:     Update $\mathbf{U}$ by Eq.(3.9)
 4:     Update $\mathbf{V}$ by Eq.(3.13)
 5:     Update $\mathbf{w}$ by Eq.(3.14)
 6:     If convergence Break
 7: End For
 8: Return $\mathbf{U}$, $\mathbf{V}$, $\mathbf{w}$
```

4 Experiments

In this section, we conduct experiments to assess the performance of the proposed framework, namely Cross-topic Emerging Rumor deTection (CERT), with real world social media data. In particular, we aim to answer the following two questions through experiments:

- How effective is CERT in detecting emerging rumors in social media by leveraging prior labeled data of rumors?
- How quickly can CERT detect emerging rumors after rumors start being spread with only prior labeled data of rumors?

We begin by introducing how we obtain the real-world social media data and the corresponding ground truth. Then we introduce the experimental setup and baselines for comparison. Based on the experimental results, we finally investigate the effectiveness and the earliness of CERT on rumor detection.

4.1 Datasets Over 200 million posts are posted per day on Twitter[2] and the popularity has made Twitter a testbed for rumor detection research [18, 21, 33]. In this work, we aim to collect a large dataset that includes tweets about all prior rumors within a certain period. Following [18], we leverage Twitter Search API[3] to retrieve tweets of interests by compiling queries with a fact-checking website.

In order to validate and debunk unverified information, several fact-checking websites have been developed. Verification of rumors on fact-checking websites is mainly run by professional editors and trusted information sources. Though fact-checking sites may cover only a small portion of rumors in social media, the identified rumors offers us valuable resources

[2] https://blog.twitter.com/2011/200-million-tweets-per-day

[3] https://dev.twitter.com/rest/public/search

to evaluate rumor detection algorithms. In this work, we choose Snopes[4] to obtain ground truth, which is the top rumor reference site according to Alexa[5]. In order to obtain non-rumor posts pertaining to the same topic, we extract keywords in regular expressions as queries to retrieve posts.

With queries generated from 252 rumors from June 30^{th} to July 11^{th}, we collect 9,918 tweets and hire two human annotators to manually verify that they are rumors. The annotators classify a tweet by reading the content and referring to the Snopes article. The inter-judge agreement over all data instances achieves a high Cohen's κ score 0.93, which demonstrates the annotation accuracy. An expert makes the final judge when annotators disagree with each other. The resultant dataset contains 1,618 rumor instances and 8,300 non-rumor instances.

4.2 Experimental Settings We follow conventional settings [18] to evaluate the performance with *Precision*, *Recall*, and *F-measure*. All other parameters are set with cross-validation based on a holdout dataset. Next, we will introduce methods that we use to compare with CERT. First, we aim to investigate how effective is CERT in detecting emerging rumors with the historical rumors. Since CERT jointly clusters rumors, selects features and trains a classifier, first, we introduce three variants of the proposed method to validate different aspects of CERT:

- **Pooling** trains the classifier on the prior training data directly without clustering data or selecting features, and we adopt the linear-SVM as the classifier. As shown in Figure 1, the way of CERT to model prior labeled data is to cluster them into different rumor categories. On the contrary, Pooling directly learns a classifier with all prior labeled data. Hence, Pooling is used to validate the necessity of structure learning and feature selection.

- **Elastic Net** trains the classifier by imposing a s-parsity regularization term to select features. Pooling aims to evaluate structure learning and feature selection as a whole, while Elastic Net only tests the effectiveness of feature selection. Elastic Net aims to learn a sparse classifier with fewer selected features without clustering data instances into rumor categories. So the result can be used to validate the necessity of structure learning.

- **KM_SVM** first clusters data instances and trains a classifier for each cluster. KM_SVM is designed to evaluate the method that separately clusters data and trains classifiers. Since we propose to unify the data clustering and classifier learning processes, the result of KM_SVM can be used to validate the necessity of the joint learning framework. Given a test instance, we first find the closest cluster center and apply the corresponding classifier of the cluster to determine the label of the test instance.

Several methods have been proposed to identify unverified information from social media. In order to compare with the state-of-the-art approaches, we include the following methods:

- **FE_LL** [18]: Rumors that are widespread in social media usually share similar patterns in terms of content and diffusion. In order to capture the patterns of rumors, Qazvinian *et al.* implement a method to extract relevant features that capture the patterns of rumors. Based on the extracted features and labeled instances, classifiers are trained to predict rumors. The adopted classifier is a ℓ_1-regularized log-linear model.

- **LK_RBF** [21]: A problem that hinders the early detection of rumors is the data scarcity: only few comments are available and they are scattered in different discussion threads. In order to relieve the scarcity, a possible way is to combine these individual tweets from different threads together as a "conversation". Sampson *et al.* propose several methods to combine tweets and try different supervised learning methods to classify rumors. We choose the URL-based method to combine tweets and the RBF kernel method as the classifier, which achieve the best performance in that work and also on our dataset. LK_RBF is effective for detecting rumors in the early stage, and the comparison can be used to evaluate the earliness of CERT.

We design two experiments to show the performance of CERT. In the first experiment for studying the effectiveness, we arrange rumors in the chronological order by the starting time, and we take the first 50% for training and the rest for testing. Therefore, all methods predict new rumors with historical training data and the experiment shows the performance on cross-training. In the second one, baseline methods are trained on the rumors for evaluation, and the training data is added in the chronological order by the generation time. The second experiment shows the minimum time that could be saved by CERT regardless of the annotation.

[4] http://www.snopes.com/

[5] http://www.alexa.com/topsites/category/Society/Folklore/Literature/Urban_Legends

Table 2: Performance on detecting emerging rumors.

Approaches	Precision	Recall	F-score
Pooling	76.13%	60.20%	67.23%
Elastic Net	79.56%	65.62%	71.92%
KM_SVM	70.12%	72.55%	71.31%
FE_LL	86.29%	85.33%	85.81%
LK_RBF	80.16%	64.62%	71.56%
CERT	**92.18%**	**88.15%**	**90.12%**

4.3 Effectiveness Analysis The comparison of the performance is shown in Table 2. Precision shows how accurate rumors can be detected, recall shows how sensitive the models are to rumors, and F-score (F-1 measure) is the harmonic mean of precision and recall. Based on the results shown in Table 2, we draw the following observations. The three variants, *i.e.,* Pooling, Elastic Net, and KM_SVM, cannot effectively detect emerging rumors with historical training data. Imposing a feature selection is useful since Elastic Net outperforms Pooling. Disjointly clustering and detecting rumors with KM_SVM does not achieve comparable results, which proves the necessity of a coherent method.

Among the two rumor detection methods, *i.e.,* FE_LL and LK_RBF, FE_LL achieves the better results and is the runner-up among all methods, showing that feature engineering helps detect rumors better. The feature engineering process can be integrated into CERT easily. CERT outperforms existing methods by jointly grouping data instances, selecting features and learning classifiers. The result empirically demonstrates that CERT is effective in exploiting knowledge in historical training data.

4.4 Earliness Analysis In the second experiment, we allow existing rumor detection methods to be trained on rumors that are for evaluation. Through incrementally adding training data in the chronological order, we will be able to estimate the time that can be saved by utilizing historical data. The results on earliness are shown in Figure 2. Note that, CERT is trained **only** with historical data, meaning that when the other two methods are trained on more labeled data of the emerging rumor, CERT is not retrained and only exploits the prior labeled data.

At an early stage with 10% to 50% training data, LK_RBF outperforms FE_LL regarding F-score, showing that linking and combining posts with the same URLs alleviates the data scarcity problem. With more data being generated, the advantages of linking data become diminishing, and FE_LL outperforms LK_RBF. The result shows that FE_LL is more effective with abundant training data, while LK_RBF is more useful for an emerging rumor. However, the best baseline achieves the result of CERT with 70% training data, which has an average time lag of 22 hours. Therefore, we empirically prove that the use of CERT not only yields effective classifiers but also finds emerging rumors faster than existing approaches.

4.5 Rumor Categories An intermediate task is to cluster rumors into categories, which is helpful for the detection since rumors of the same category trigger similar reactions [6]. To help understand the clustering results, we show three example categories and the corresponding top rumors in the category. The results are illustrated in Table 3, including wedge-driving rumors, dread rumors and curiosity rumors. The name of the three clusters is acquired through manual checking. We see that rumors are clustered cohesively, and the cohesiveness explains how it facilitates selecting key features from sparse data.

5 Related Work

The study of rumor can be traced back to 1940s [1], when psychological studies try to discover the driving forces of rumors [2]. Since then, story importance and information ambiguity have been intensively investigated for understanding rumor spreading [3, 20]. For example, Anthony first introduces to use anxiety to approximate the rumor importance, since it is easier to model the sentiment of readers. Rosnow introduces the concept of *personal involvement* as an additional variable related to user anxiety. However, the efforts on studying anxiety of social media users for detecting rumors are very few. Oh *et al.* utilize anxiety on social media sites for understanding the diffusion of rumors [17], and our work first introduces to exploit it as a topic-independent feature, which paves the way for effectively reusing historical data.

In order to combat the widespread misinformation in social media, supervised learning approaches have been used [31] to detect rumors and the spreaders [14]. The creation and spread of rumors have first been studied on Twitter. Since it is crucial to detect rumors as early as possible, Qazvinian *et al.* employ a feature engineering approach to distinguish misinformation from Twitter's content stream [18]. Observing rumor posts usually arise inquiries, Zhao *et al.* compile regular expressions to detect topic with concentrated questions [33]. Our recent work studies linking distributed discussion snippets to alleviate the cold-start problem [21]. All aforementioned approaches depend on a labeled dataset, which unavoidably introduces a time lag for annotation. Meanwhile, systems have also been

Table 3: Three example categories of rumors detected by CERT.

Wedge-Driving Rumors	Dread Rumors	Curiosity Rumors
President Obama claimed that Americans would be better off under the martial law during an interview with Washington Post.	Police in assessed that an encounter with three men at Silver Lake Park was an attempted human trafficking incident.	A North Carolina provider of mental health services is named "Nutz R Us."
A Black Lives Matter protest in Memphis obstructed I-40, leading to the death of a critically ill child transplant patient.	A "purge" event is planned for 9 July 2016 in Baton Rouge kill all police officers.	A fisherman captured a 3,000 lb. great white shark out of the waters in the Great Lakes Michigan.
A police officer shot two-year-old Malik Gibson after mistaking his pacifier for a gun.	NASA has warned of imminent disaster due to the trajectory of Nibiru.	Researchers sequenced octopus genomes and discovered alien DNA.

developed to visualize and track known rumors [12, 22]. Through representing data via intuitive visualization, experts can observe and understand how rumors spread from node to node, so that they are enabled to supervise the learning procedure of rumor classifiers with their domain knowledge and expertise [5, 32].

Our work is also related to feature selection. Feature selection aims to find the subset of features that are optimal for a learning task [11]. A common method is to impose a sparsity regularization term. For example, ℓ_1-norm leads to a sparse representation of variables in a regression task [26] and $\ell_{2,1}$-norm jointly leads to sparse models and preserves the intrinsic structure of data [34]. Since the intrinsic data structures may remain to be discovered, such as the prior rumor data, recent studies have investigated simultaneously characterizing the structure and selecting features. Du *et al.* propose to preserve global and local structure [7], and Wang *et al.* integrate feature selection into a clustering process [28]. However, since our ultimate goal is to learn a classifier, these methods are not applicable.

6 Conclusion

Circulating online rumors have become a key issue for today's social media sites. They may result in catastrophic effect both online and offline quickly. After they go viral, it is extremely difficult to eliminate their existence. In order to detect rumors at an early stage, we propose to directly train a classifier based on readily available labeled data from prior rumors. Motivated by traditional studies on rumors, we introduce a novel framework that jointly clusters data, selects features, and trains classifiers. An optimization approach is also presented to solve the problem efficiently. The proposed framework, CERT, largely breaks the bottleneck of the time lag from annotating datasets. Experimental results illustrate the effectiveness and earliness of CERT on real-world data.

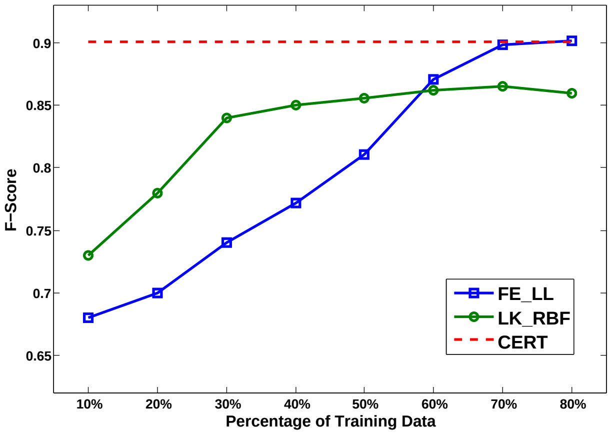

Figure 2: Performance of traditional approaches with chronologically additional training data, while CERT uses the historical data.

The same rumor may outbreak in miscellaneous websites. For example, after being created, a political rumor may be spread on Twitter, Facebook, and even news media simultaneously. It would be interesting to investigate the potential impact of jointly modeling cross-network information. Besides, a social media rumor may consist of contents other than text, such as short videos and images. In the future, we would like to utilize cross-modal information to further facilitate the detection of rumors of all sorts.

Acknowledgments

We would like to thank anonymous reviewers for their constructive comments. The work is funded, in part, by ONR N00014-16-1-2257 and the Department of Defense under the MINERVA initiative through the ONR N000141310835.

References

[1] Gordon W Allport and Leo Postman. The basic psychology of rumor. *Transactions of the New York Academy of Sciences*, 1945.

[2] Gordon W Allport and Leo Postman. *The Psychology of Rumor.* Henry Holt, 1947.

[3] Susan Anthony. Anxiety and rumor. *The Journal of Social Psychology*, 1973.

[4] MJ Box, William Henry Swann, and D Davies. Nonlinear optimization techniques. 1969.

[5] Nan Cao, Conglei Shi, Sabrina Lin, Jie Lu, Yu-Ru Lin, and Ching-Yung Lin. Targetvue: Visual analysis of anomalous user behaviors in online communication systems. *IEEE Transactions on Visualization and Computer Graphics*, 22(1), 2016.

[6] Nicholas DiFonzo and Prashant Bordia. *Rumor Psychology: Social and Organizational Approaches.* American Psychological Association, 2007.

[7] Liang Du and Yi-Dong Shen. Unsupervised feature selection with adaptive structure learning. In *International Conference on Knowledge Discovery and Data Mining*, pages 209–218, 2015.

[8] Roger Fletcher. *Practical Methods of Optimization.* John Wiley & Sons, 2013.

[9] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations.* CRC Press, 2015.

[10] Jan H Kietzmann, Kristopher Hermkens, Ian P McCarthy, and Bruno S Silvestre. Social media? get serious! understanding the functional building blocks of social media. *Business horizons*, 2011.

[11] Jundong Li, Xia Hu, Liang Wu, and Huan Liu. Robust unsupervised feature selection on networked data. In *2016 SIAM International Conference on Data Mining (SDM)*, pages 387–395, 2016.

[12] Karissa Rae McKelvey and Filippo Menczer. Truthy: Enabling the study of online social networks. In *CSCW*, pages 23–26. ACM, 2013.

[13] Jorge J Moré and David J Thuente. Line search algorithms with guaranteed sufficient decrease. *ACM Transactions on Mathematical Software (TOMS)*, 20(3):286–307, 1994.

[14] Fred Morstatter, Liang Wu, Tahora H Nazer, Kathleen M Carley, and Huan Liu. A new approach to bot detection: Striking the balance between precision and recall. In *ASONAM*, pages 533–540, 2016.

[15] Jorge Nocedal and Stephen Wright. *Numerical Optimization.* 2006.

[16] Guillaume Obozinski, Ben Taskar, and Michael Jordan. Multi-task feature selection. *Statistics Department, UC Berkeley, Tech. Rep*, 2, 2006.

[17] Onook Oh, Manish Agrawal, and H Raghav Rao. Community intelligence and social media services: A rumor theoretic analysis of tweets during social crises. *Mis Quarterly*, 37(2):407–426, 2013.

[18] Vahed Qazvinian, Emily Rosengren, Dragomir R Radev, and Qiaozhu Mei. Rumor has it: Identifying misinformation in microblogs. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1589–1599. Association for Computational Linguistics, 2011.

[19] Jason DM Rennie. Smooth hinge classification. *Proceeding of Massachusetts Institute of Technology*, 2005.

[20] Ralph L Rosnow. Inside rumor: A personal journey. *American Psychologist*, 46(5):484, 1991.

[21] Justin Sampson, Fred Morstatter, Liang Wu, and Huan Liu. Leveraging the implicit structure within social media for emergent rumor detection. *ACM Conference on Information and Knowledge Management (CIKM)*, 2016.

[22] Chengcheng Shao, Giovanni Luca Ciampaglia, Alessandro Flammini, and Filippo Menczer. Hoaxy: A platform for tracking online misinformation. In *WWW*, 2016.

[23] Jack Sherman and Winifred J Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *The Annals of Mathematical Statistics*, 21(1):124–127, 1950.

[24] Ajit P Singh and Geoffrey J Gordon. Relational learning via collective matrix factorization. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 650–658, 2008.

[25] Jiliang Tang and Huan Liu. Feature selection with linked data in social media. 2012.

[26] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[27] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[28] Suhang Wang, Jiliang Tang, and Huan Liu. Embedded unsupervised feature selection. In *AAAI*, pages 470–476, 2015.

[29] Zaiwen Wen and Wotao Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434, 2013.

[30] Liang Wu, Xia Hu, and Huan Liu. Relational learning with social status analysis. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 513–522. ACM, 2016.

[31] Liang Wu, Fred Morstatter, Xia Hu, and Huan Liu. Mining misinformation in social media. In *Big Data in Complex and Social Networks*, pages 123–152. CRC Press, 2016.

[32] Jian Zhao, Nan Cao, Zhen Wen, Yale Song, Yu-Ru Lin, and Christopher Collins. # fluxflow: Visual analysis of anomalous information spreading on social media. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1773–1782, 2014.

[33] Zhe Zhao, Paul Resnick, and Qiaozhu Mei. Enquiring minds: Early detection of rumors in social media from enquiry posts. In *WWW*, pages 1395–1405, 2015.

[34] Yang Zhou, Rong Jin, and Steven CH Hoi. Exclusive lasso for multi-task feature selection. In *AISTATS*, volume 9, pages 988–995, 2010.