Semi-supervised Multi-instance Interpretable Models for Flu Shot Adverse Event Detection

Junxiang Wang, Liang Zhao
Department of Information Sciences and Technology
George Mason University
{jwang40,lzhao9}@gmu.edu

Yanfang Ye
Lane Department of Computer Science and Electrical Engineering
West Virginia University
yanfang.ye@mail.wvu.edu

***Abstract*—It is important to track adverse events that occur due to flu shots as those could pose a serious threat to public health. Traditional adverse event reporting systems suffer from poor timeliness and a severe lack of data. In contrast, social media like Twitter and Facebook have become ubiquitous real-time social sensors where user states are indicated swiftly and extensively. However, little work has focused on adverse event detection using social media because of several challenges that have not been jointly solved: 1) message sparsity with irrelevant topics, 2) the difficulty of labeling health states, and 3) scalability in parameter optimization. To address these problems simultaneously, this paper presents a new semi-supervised multi-instance learning model to detect potential adverse events reflected by social media, which will facilitate the further clinical verification and prompt intervention. Specifically, given only user-level labels, this model interpretably identifies the user's adverse-event-indicative messages by employing a multi-instance learning strategy; unlabeled users' messages are also utilized to improve classifier performance by a semi-supervised term. Two models and corresponding algorithms, namely the non-smooth Semi-Supervised Multi-instance (nSSM) algorithm and the smooth Semi-Supervised Multi-instance (sSSM) algorithm, have been developed to optimize parameters accurately and efficiently. Experiments on a synthetic dataset and a real Twitter dataset confirm that our model outperforms other baseline models. Case studies show interesting interpretable patterns including key messages, keywords, and several common symptoms found in adverse-relevant tweets extracted by our methods.**

I. Introduction

A wide range of vaccinations are now available worldwide such as influenza and hepatitis B, making a significant contribution to global health. Many people are reached by vaccination programs; for example, flu vaccination coverage during the 2014-15 flu season was 47.1 percent of the whole U.S. population, according to a report released by the CDC[1]. However, although it does prevent people from becoming infected, sometimes vaccination itself can cause adverse reactions in large populations, which is now one of the most significant issues in healthcare. For example, 25.8 percent of adverse events, including one diagnosed as pneumonia, were reported from a recent Influenza A (H1N1) vaccination program in Korea[18]. Severe adverse reactions may even lead to death. For instance, a woman died of multiorgan failure and respiratory distress, which is clinically verified to be caused by a yellow fever vaccination in Spain on October 24, 2004[14]. Therefore, considering the immense influence and potentially severe consequences of these vaccination adverse reactions, a system that can promptly and accurately identify adverse events is imperative (e.g. FDA's Sentinel Initiative[4]).

Traditionally, adverse events are gathered by reporting systems where users submit long descriptions with complicated forms after the victims recover from the adverse reactions. Here are two major drawbacks to this system: First, only a few people will actually submit a formal adverse report due to the complex procedures involved, and second, there is a serious time delay for the submission of such formal reports due to administrative processing. For example, the FDA's[2] adverse event reporting system typically only releases data every three months. In contrast, social media like Twitter and Facebook, which have rapidly become new information dissemination platforms, have begun to be used in several applications in health care[22][19][27] because social media can capture timely and ubiquitous disease information from social sensors. These advantages effectively address the drawbacks in traditional adverse event reporting systems. However, until now little work has focused on their use for adverse event detection, despite the immense potential of this approach.

Flu shots adverse event detection suffers from several challenges. **1. The sparsity of indicative messages**. Side-effect descriptions in the messages regarding flu shots are indicators of flu shot adverse events. However, these messages are very sparse, which makes a classifier difficult to catch such message indicators. According to our dataset of 300 users labeled as positive, only 7.22% were indicative of adverse reactions. **2. Cost of labeling health states.** Even though it is mandatory to check message by message in order to label health states, labeling sufficient users is prohibitively labor-intensive. For example, a user on average has at least 100 messages every month, so in order to collect a label set with a modest size of ten thousand, it is required to check a million messages, which is difficult to accomplish manually. Considering millions of users, most of them remain unlabeled. However, little existing work for health states has typically utilized them to improve model performance, which leads to a huge information loss. **3. Scalability in parameter optimization.** Flu shot adverse

[1]CDC: U.S. Centers for Disease Control and Prevention. https://www.cdc.gov

[2]FDA: U.S. Food and Drug Administration. http://www.fda.gov

978-1-5386-5035-6/18/$31.00 ©2018 IEEE

event detection task entails the use of a real-time (or near real-time) framework and hence computational scalability is critical. However, this is challenging for several reasons, including (1) high-dimensional features that characterize thousands of enriched keywords; (2) the large user sets consisting of large-scale social networks; and (3) a large message set for each user. This implies that even a medium dataset with 1,000 keywords and 10,000 users, each of whom has 100 messages, will yield 1 billion data points, which entails massive storage and time-consuming computation. As a result, flu shot adverse event detection requires large-scalable optimization algorithms.

In order to simultaneously address all these technical problems, we propose a novel semi-supervised multi-instance learning model. Specifically, given only user-level labels, this model automatically selects representative messages by a multi-instance learning strategy which combines user levels with message levels directly. In the meantime, unlabeled users' messages are also utilized to stabilize the decision boundaries of the classifier and reinforce its generalization abilities by a semi-supervised term. To handle the challenge of scalability, we have developed two optimization algorithms that we have named the non-smooth Semi-Supervised Multi-instance (nSSM) algorithm and the smooth Semi-Supervised Multi-instance (sSSM) algorithm based on the Alternating Direction Method of Multipliers (ADMM)[7] to process large-scale user and message data in a decentralized fashion.

The main contributions of our research are summarized as follows:

- **Design a framework to address the flu shot adverse event detection problem.** A general framework for detecting adverse events for flu shots on Twitter is formulated. Some classic models are proved to be special cases of our generalized model.
- **Develop an effective nSSM algorithm based on the ADMM and a linear search method.** The objective function proposed by the model contains the non-smooth max function, the nSSM algorithm based on the ADMM and a linear search method is developed to deal with the max function directly.
- **Propose an efficient sSSM algorithm based on the ADMM and a smooth approximation approach.** The nSSM algorithm cannot ensure convergence. Therefore, the efficient sSSM algorithm based on the ADMM and a smooth approximation approach substitutes a softmax operator for the max function and guarantees convergence.
- **Conduct extensive experiments for performance evaluations.** Experiments on a synthetic dataset and a real Twitter dataset show that our nSSM and sSSM outperform other models. Key parameters of the proposed model and scalability are explored on the Twitter data. Furthermore, case studies show interesting keyword patterns and several common symptoms found in adverse-relevant tweets extracted by our methods.

The rest of the paper is organized as follows. In Section II, we summarize recent research work related to this paper. In Section III, we present the problem formulation. In Section IV, we propose the learning framework and two models. In Section V, we develop two effective ADMM-based optimization algorithms to solve two optimization problems. In Section VI, extensive experiments are conducted to validate the effectiveness of our model. Section VII concludes by summarizing the whole paper.

II. Related Work

This section introduces the related work in several research fields.

Multi-instance learning. Multi-instance classifiers are categorized as either *instance-level* and *bag-level*[2]. *Instance-level* classifiers score each instance without considering the characteristic of the whole bag. For example, the image classification of beaches and non-beaches is determined by their visual content[2], and Kumar and Raj detected audio events based on a collection of audio recordings[21]. *Bag-level* is more common than *instance-level*. Dietterich et al. evaluate drugs as being good if at least one of the three-dimensional shapes binds well with the target binding site[2][13]. Andrews et al. gave instance-level and bag-level formulations as a maximum margin problem in their Support Vector Machines (SVM) settings[3]. Zhou et al. developed two methods to discriminate bag labels by graph theories[43]. However, few of these methods are performed in the social media setting and on text data types. As an example, Wang et al. utilized formal reports to detect vaccine adverse events in the multi-instance learning framework[33].

Semi-supervised learning. This technique has been employed extensively in the data mining field. The main idea of semi-supervised learning is to utilize a large number of unlabeled data to enhance classification ability[9]. Some researchers have applied it to improve training performance. For example, Cheng et al. combined parallel corpora and monolingual corpora to improve a neural machine translation (NMT) system[11]. Zhang et al. re-evaluated mislabeled data, exploiting the complementarity between audio-visual features[40]. Wang et al. employed the graph-based semi-supervised learning model to perform object detection and segmentation from multi-view images[32]. Guillaumin et al. combined image information with keywords from labeled and unlabeled images to improve image categorization performance[17]. Others have applied it to the unsupervised learning or clustering process. For example, Cohn et al. iterated the clustering process with user feedback[12] and Wang et al. guided the co-clustering process with inter-type information and constraints[31]. However, little work has been done on utilizing social media data to improve classification ability.

Adverse event surveillance and detection. Previously, much work attempted to analyze adverse events in traditional adverse event reporting systems. For example, Cai et al. proposed a random effects model to test the heterogeneity of reporting rates across reporting years[8]. Shimabukuro et al. provided an overview of the Vaccine Adverse Event Reporting System (VAERS) and described strengths and limitations about

TABLE I
NOTATIONS AND DESCRIPTIONS

Notations	Descriptions
X_u	The tweet set from user u
Y_u	The predefined label from user u
K	The keyword set
U	The user set
U_1	The set of labeled users
U_2	The set of unlabeled users
β	The coefficient vector of the keyword set
β_0	The constant intercept

VAERS[29]. Recently, health care topics on social media have begun to attract considerable attention from researchers. Some work focuses on flu surveillance. For instance, Lee et al. conducted a real-time analysis of Twitter data to detect seasonal flu[20] and Chen et al. inferred the hidden state of a user from his tweets during flu outbreaks and aggregated state statistics by geographic region[10]. Signorini et al. tracked public concerns regarding H1N1 flu and measured flu activities[30], while Lampos et al. monitored the incidence of flu-like illness in several areas of the United Kingdom using a Twitter microblogging service[23]. Some work detects sentiments related to vaccinations in social media. For example, Marcel Salathe and Shashank Khandelwal measured the spatiotemporal sentiment towards a new vaccine using public social media data collected over six months[28]. Mitra et al. identified users who persistently hold pro and anti attitudes to vaccination on Twitter and analyzed the drivers of attitudes[26]. Others are drug-related adverse event detection. For example, Metke et al. evaluated the impact of the text-processing step on the extraction results[25]. Yomtov and Gabrilovich proposed a low-cost method to monitor adverse drug events continuously[38]. White et al. mined adverse drug events on the Internet user search logs[36]. Freifeld et al. evaluated the level of concordance between adverse-relevant tweets and spontaneous reports received by a regulatory agency[16]. However, to the best of our knowledge, little work has aimed to detect adverse events caused by flu shots.

III. PROBLEM SETUP

In this section, the problem addressed by this research is formulated in the Twitter setting.

A. Problem Formulation

Important notations used in this paper are described in Table I. Suppose a tweet set is denoted as $X = \{X_u\}_{u \in U}$, where a user set is denoted as U and $X_u \in \mathbb{Z}^{n_u \times |K|}$ stands for the tweets from user u. K denotes a keyword set that represents symptom descriptions of flu shots and n_u refers to the number of tweets from user u. $X_{u,i}$ stands for the ith tweet from user u. The jth entry of $X_{u,i}$, denoted by $X_{u,i,j}$, is the count of the jth keyword in the ith tweet from user u. The user set U is then divided into two disjoint parts: $U = U_1 \cup U_2$, where U_1 and U_2 denote the set of labeled users and unlabeled users, respectively. A labeled user $u \in U_1$ has a predefined label Y_u while an unlabeled user does not. $Y_u \in \{0, 1\}$ denotes the health state of user u, $Y_u = 1$ implies that user u is regarded as a positive (i.e., affected by adverse events) user while $Y_u = 0$ shows user u is negative. $Y = \{Y_u\}_{u \in U}$ denotes the health states of all users. $X_1 = \{X_u\}_{u \in U_1}$ and $X_2 = \{X_u\}_{u \in U_2}$ denote tweet sets of labeled users and unlabeled users, respectively. Then the problem of flu shots adverse event detection can be formulated as follows:

Problem Formulation: Given a tweet set $X = \{X_u\}_{u \in U}$, the goal of the problem is to detect the health state of a user $u \in U$ by learning the mapping f:

$$f : \{X_{u,1}, X_{u,2}, \cdots, X_{u,n_u}\} \rightarrow Y_u \tag{1}$$

B. Challenges

To address the formulated problem in Equation (1), several specific challenges remain unsolved: 1) Only a small proportion of tweets indicate adverse events, but it is difficult to make them stand out. 2) The user set U consists of the labeled user set U_1 and the unlabeled user set U_2, so the utilization of U_2 is important because in practice $|U_1| \ll |U_2|$. 3) An optimization algorithm to solve this problem meets the difficulty of a growing amount of computation as the user set $|U|$, the keyword set $|K|$ and the tweet set $|X|$ increase substantially. Thus in the next two sections, we propose a novel semi-supervised multi-instance learning model to address these problems in turn.

IV. SEMI-SUPERVISED MULTI-INSTANCE LEARNING MODEL

A. Automatic Representative Tweet Selection

We begin by considering how to select representative tweets automatically for a user. Although user $u \in U$ may have a large number of tweets, only few of them will be relevant to flu shots. As for positive users, we only need a tweet indicating abnormal symptoms, but for negative users none of tweets should imply any abnormal information. In order to dig positive tweets out, **the max rule**, a multi-instance learning strategy, is applied to flu shot adverse event detection task, provided that every user and their tweets are considered as a bag and instances, respectively. The max rule means that a bag (i.e., user) label is judged as positive if at least an instance (i.e., tweet) indicates a positive label and negative if none of the instances are indicative of a positive label. Suppose a logistic regression model is used to predict the probability of instance labels, the max rule summarizes the relation between bags and instances mathematically as follows:

$$p_u = \max_{i=1,\cdots,n_u} logit(X_{u,i}; \beta; \beta_0) \tag{2}$$

where $logit(X_{u,i}; \beta; \beta_0) = 1/(1 + \exp(-\beta^T X_{u,i} - \beta_0))$ is a logit function[15] which predicts the probability of an instance label from a input vector $X_{u,i}$, p_u is a predicted probability of a positive label for user u, β is a coefficient vector from the keyword set K, of which each element indicates the weight of a keyword and β_0 is a constant intercept.

The max rule biases for positive users and hence offsets

the bias induced by the problem of highly imbalanced classes where the majority of users are labeled as negative, as shown in empirical results (in Section VI-B). The max rule is also very resistant to noisy tweet data because the representative tweets for a user depend on not a single tweet but on all candidate tweets. For example, suppose 1000 candidate tweets are posted by a user, the noiseless representative tweet selected by the max rule prevents the introduction of noise from other 999 tweets.

B. Utilization of Unlabeled Users

In practice, the number of users who have been labeled is very limited and labeling users is a laborious task. As a result, $|U_1| \ll |U_2|$, i.e. unlabeled users greatly outnumber labeled users. Instead of learning with labeled users only, we introduce unlabeled users based on the following two considerations: first, while the decision boundary for the logistic regression varies widely based on the limited labeled users, these extra unlabeled users restrict the range of the decision boundary and hence improve its generalization ability; second, the introduction of unlabeled users increases the size of the training samples and reduces the noise induced by labeled users. Therefore, we introduce a semi-supervised loss $L_u(\beta;\beta_0)$ for an unlabeled user $u \in U_2$ as follows:

$$L_u(\beta;\beta_0) = -\min(\log(p_u), \log(1-p_u)) \tag{3}$$

where p_u determined by β and β_0 is defined in Equation (2). The $L_u(\beta;\beta_0)$ forces p_u to approach toward either 0 or 1 so that probabilities p_u for all unlabeled users $u \in U_2$ are separable from a clear decision boundary determined by β and β_0.

C. Overall Model

We combine the max rule for multi-instance learning with unlabeled data for semi-supervised learning to a unified learning model. The proposed learning model aims to minimize the empirical risk.

$$\begin{aligned}(\beta^*, \beta_0^*) = &\arg\min_{\beta,\beta_0} \sum\nolimits_{u\in U_1} H_u(\beta;\beta_0) \\ &+\nu \sum\nolimits_{u\in U_2} L_u(\beta;\beta_0) + \Omega(\beta)\end{aligned} \tag{4}$$

where $H_u(\beta;\beta_0) = -Y_u \log p_u - (1-Y_u)\log(1-p_u)$ is a log-loss for user $u \in U_1$, $L_u(\beta;\beta_0)$ is given in Equation (3), $\Omega(\beta)$ is a regularization term and $\nu > 0$ is a parameter. $\Omega(\beta) = \lambda\|\beta\|_1$ due to the nature of high dimension and high sparsity of the feature set, the sparsity of β is enforced conventionally by ℓ_1-norm, where $\lambda > 0$ is a regularization parameter.

D. Objective Functions and Approximation

In this subsection, two transformations of the original model are conducted to form two responding models: the non-smooth Semi-supervised Multi-instance (nSSM) model and the smooth Semi-supervised Multi-instance (sSSM) model. The nSSM model is equivalent of the original model; the sSSM model replaces the non-smooth terms $\max(\cdot)$ and $\min(\cdot)$ in the original model with the smooth approximation.

1) the nSSM model: We transform the nSSM model directly by integrating Equation (2) into Equation (4) and using the fact that $\max(\log(\cdot)) = \log(\max(\cdot))$.

$$\begin{aligned}(\beta^*,\beta_0^*) = &\arg\min_{\beta,\beta_0} \sum\nolimits_{u\in U_1} (\log(1+\exp(\max\nolimits_{i=1,\cdots,n_u}(X_{u,i}\beta+\beta_0))) \\ &- Y_u \max\nolimits_{i=1,\cdots,n_u}(X_{u,i}\beta+\beta_0) + \lambda\|\beta\|_1 \\ &+ \nu \sum\nolimits_{u\in U_2} (\log(1+\exp(\max\nolimits_{i=1,\cdots,n_u}(X_{u,i}\beta+\beta_0))) \\ &- \min(\max\nolimits_{i=1,\cdots,n_u}(X_{u,i}\beta+\beta_0, 0)))\end{aligned} \tag{5}$$

2) the sSSM model: The $\max(\cdot)$ and $\min(\cdot)$ in the Equation (5) are non-smooth, which make the problem non-convex and parameter optimization difficult. To address this problem, the sSSM model is proposed below to approximate these non-smooth terms, which is simply to replace the non-smooth $\max(\cdot)$ and $\min(\cdot)$ with smooth softmax operators[6]. Softmax operators approximate $\max(\cdot)$ and $\min(\cdot)$ asymptotically while preserving the smoothness, which are are formulated as follows:

$$\begin{aligned}&\max\nolimits_{i=1,\cdots,n_u}(\beta^T X_{u,i} + \beta_0) \approx (\log \sum\nolimits_{i=1}^{n_u} e^{A(\beta X_{u,i}+\beta_0)})/A \\ &-\min(\log(p_u), \log(1-p_u)) = \max(-\log(p_u), -\log(1-p_u)) \\ &\qquad\qquad \approx \log(e^{-B\log(p_u)} + e^{-B\log(1-p_u)})/B\end{aligned}$$

where $A > 0$ and $B > 0$ control the approximation degree of the non-smooth $\max(\cdot)$ and $\min(\cdot)$. Then the objective function is approximated from (4) as follows:

$$\begin{aligned}(\beta^*, \beta_0^*) = &\arg\min_{\beta,\beta_0} \sum\nolimits_{u\in U_1} (-Y_u \log p_u - (1-Y_u)\log(1-p_u)) \\ &+\nu \sum\nolimits_{u\in U_2} \log(e^{-B\log(p_u)} + e^{-B\log(1-p_u)})/B + \lambda\|\beta\|_1 \\ &s.t.\ p_u = 1/(1+(\sum\nolimits_{i=1}^{n_u} e^{A(\beta X_{u,i}+\beta_0)})^{-1/A})\end{aligned} \tag{6}$$

where p_u is the probability of a positive label for user $u \in U$ based on the softmax approximation.

E. Relationship to Previous Related Approaches

In this section, we show that several classic methods are special cases of our model.

1. Generalization of logistic regression. Let $n_u = 1$ for $u \in U_1$ and $U_2 = \emptyset$. The model then is reduced to a logistic regression with ℓ_1-norm regularization[7]:

$$\begin{aligned}(\beta^*, \beta_0^*) = &\arg\min_{\beta,\beta_0} \sum\nolimits_{u\in U} -Y_u \log p_u - (1-Y_u)\log(1-p_u) \\ &+ \lambda\|\beta\|_1 \\ &s.t.\ p_u = logit(X_u;\beta;\beta_0)\end{aligned}$$

where U is the user set, since $U = U_1$.

2. Generalization of logistic regression combined with semi-supervised learning. Let $n_u = 1$. The model is then reduced to a logistic regression combined with semi-supervised learning with ℓ_1-norm regularization[1].

$$\begin{aligned}(\beta^*, \beta_0^*) = &\arg\min_{\beta} \sum\nolimits_{u\in U_1} -Y_u \log p_u - (1-Y_u)\log(1-p_u) \\ &- \nu \sum\nolimits_{u\in U_2} \min(\log p_u, \log(1-p_u)) + \lambda\|\beta\|_1 \\ &s.t.\ p_u = logit(X_u;\beta;\beta_0)\end{aligned}$$

3. Generalization of logistic regression combined with multi-instance learning. Let $U_2 = \emptyset$. The model is then reduced to a logistic regression combined with multi-instance learning[37].

$$(\beta^*, \beta_0^*) = \arg\min_\beta \sum\nolimits_{u \in U_1} -Y_u \log p_u - (1-Y_u)\log(1-p_u) + \lambda\|\beta\|_1$$
$$s.t.\ p_u = \max_{i=1,\cdots,n_u} logit(X_{u,i}; \beta; \beta_0)$$

V. Model Optimization

In this section, two optimization algorithms based on Alternating Direction Method of Multipliers (ADMM)[7], are designed to solve the nSSM model and the sSSM model, respectively.

A. the nSSM Algorithm

Equation (5) remains difficult to solve because the non-smooth $\max(\cdot)$ appears in the objective function. By introducing auxiliary variables S and ε, Equation (5) can be transformed into the following problem.

$$\beta^* = \arg\min_\beta \sum\nolimits_{u \in U_1} (\log(1+\exp(\varepsilon_u)) - Y_u \varepsilon_u) + \nu \sum\nolimits_{u \in U_2} (\log(1+\exp(\varepsilon_u)) - \min(\varepsilon_u, 0)) + \lambda\|\beta\|_1 \quad (7)$$
$$s.t.\ S_{u,i} = X_{u,i}\beta + \beta_0, \varepsilon_u = \max_{i=1,\cdots,n_u}(S_{u,i})$$

The Alternating Direction Method of Multipliers (ADMM)[7] is therefore utilized to decompose it into subproblems via the following Augmented Lagrangian function of Equation (7):

$$L_\rho(\beta,\beta_0,\varepsilon,S,y) = F(\varepsilon) + G(\beta) + y_1^T(\varepsilon - \max S) + y_2^T(S - X\beta - \beta_0) + \rho/2(\|\varepsilon - \max S\|_2^2 + \|S - X\beta - \beta_0\|_2^2)$$

where $\rho > 0$ is a penalty parameter, $F(\varepsilon) = \sum_{u \in U_1}(\log(1+\exp(\varepsilon_u)) - Y_u\varepsilon_u) + \nu\sum_{u \in U_2}(\log(1+\exp(\varepsilon_u)) - \min(\varepsilon_u, 0))$ and $G(\beta) = \lambda\|\beta\|_1$.

Define scale variables $v_1 = \frac{y_1}{\rho}$ and $v_2 = \frac{y_2}{\rho}$. The nSSM algorithm is shown in Algorithm 1. Concretely, Lines 9-14 calculates residuals and Lines 4-8 update each parameter alternately by solving the sub-problems described below.

1. Update ε.

The auxiliary variable ε is updated as follows:

$$\varepsilon^{k+1} \leftarrow \arg\min_\varepsilon F(\varepsilon) + (\rho^{k+1}/2)\|\varepsilon - \max S^k + v_1^k\|_2^2$$

which is a logistic regression with an ℓ_2-penalty term. A fast iterative shrinkage-thresholding algorithm (FISTA)[5] is applied to solve this problem because it converges much faster than general gradient descent methods.

2. Update (β, β_0).

The variables β and β_0 are updated as follows:

$$(\beta^{k+1}, \beta_0^{k+1}) \leftarrow \arg\min_{\beta,\beta_0} G(\beta) + (\rho^{k+1}/2)\|S^k - X\beta - \beta_0 + v_2^k\|_2^2$$

Updating (β, β_0) is a square loss function with ℓ_1-regularization, which is easily solved by FISTA[5].

Algorithm 1 the nSSM Algorithm

Require: X, Y, λ.
Ensure: β, β_0
1: Initialize β, β_0, ε, S, $\rho = 1$, $r = 0$, $s = 0$, $k = 0$.
2: **repeat**
3: Update ρ^{k+1} if necessary.
4: $\varepsilon^{k+1} \leftarrow \arg\min_\varepsilon F(\varepsilon) + (\rho^{k+1}/2)\|\varepsilon - \max S^k + v_1^k\|_2^2$.
5: $(\beta^{k+1}, \beta_0^{k+1}) \leftarrow \arg\min_{\beta,\beta_0} G(\beta) + (\rho^{k+1}/2)\|S^k - X\beta - \beta_0 + v_2^k\|_2^2$.
6: $S^{k+1} \leftarrow \arg\min_S \| \varepsilon^{k+1} - \max S + v_1^k\|_2^2 + \|S - X\beta^{k+1} - \beta_0^{k+1} + v_2^k\|_2^2$.
7: $v_1^{k+1} \leftarrow v_1^k + \rho^{k+1}(\varepsilon^{k+1} - \max S^{k+1})$.
8: $v_2^{k+1} \leftarrow v_2^k + \rho^{k+1}(S^{k+1} - X\beta^{k+1} - \beta_0^{k+1})$.
9: $r_1 = \|\varepsilon^{k+1} - \max S^{k+1}\|_2$.
10: $r_2 = \|S^{k+1} - X\beta^{k+1} - \beta_0^{k+1}\|_2$.
11: $s_1 = \|\rho^{k+1}(\max S^k - \max S^{k+1})\|_2$.
12: $s_2 = \|\rho^{k+1}X^T(S^{k+1} - S^k)\|_2$.
13: $r = \sqrt{r_1^2 + r_2^2}$. #Calculate prime residual
14: $s = \sqrt{s_1^2 + s_2^2}$. #Calculate dual residual
15: $k = k + 1$.
16: **until** convergence.
17: Output β.

3. Update S.

The auxiliary variable S is updated as follows:

$$S^{k+1} \leftarrow \arg\min_S \|\varepsilon^{k+1} - \max S + v_1^k\|_2^2 + \|S - X\beta^{k+1} - \beta_0^{k+1} + v_2^k\|_2^2$$

This is the most difficult among all sub-problems because it contains a non-smooth max function. We propose a linear search method to solve this problem, which is similar to updating D in the MREF-II optimization algorithm[41] and updating Q in the DHML algorithm[42].

4. Update v_1, v_2.

The Lagrangian multipliers v_1, v_2 are updated as follows:

$$v_1^{k+1} \leftarrow v_1^k + \rho^{k+1}(\varepsilon^{k+1} - \max S^{k+1}).$$
$$v_2^{k+1} \leftarrow v_2^k + \rho^{k+1}(S^{k+1} - X\beta^{k+1}).$$

B. The sSSM Algorithm

Even though the nSSM algorithm can solve Equation (5), the nonsmooth terms $\max(\cdot)$ and $\min(\cdot)$ may slow down the speed of convergence. This is because the nSSM may oscillate among nondifferentiable points[46]. Therefore, the alternative algorithm sSSM is proposed to deal with this challenge and accelerate convergence.

By introducing an auxiliary variable η, Equation (6) can be transformed into

$$(\beta^*, \beta_0^*) = \arg\min_{\beta,\beta_0} \sum\nolimits_{u \in U_1} -Y_u \log p_u - (1-Y_u)\log(1-p_u) + \nu \sum\nolimits_{u \in U_2} \log(e^{-B\log(p_u)} + e^{-B\log(1-p_u)})/B + \lambda\|\eta\|_1$$
$$s.t.\ p_u = 1/(1 + (\sum\nolimits_{j=1}^{n_u} e^{A(\beta X_{u,i} + \beta_0)})^{-1/A}), \beta - \eta = 0 \quad (8)$$

Similar to the non-smooth problem, the ADMM was utilized to solve this problem. The Augmented Lagrangian of Equation (8) is:

$$L_\rho(\beta, \beta_0, \eta, y) = h(\beta; \beta_0) + g(\eta) + y^T(\beta - \eta) + \rho/2\|\beta - \eta\|^2$$

where $\rho > 0$ is a penalty parameter, $h(\beta; \beta_0) = \sum_{u \in U_1} -Y_u \log p_u - (1 - Y_u)\log(1 - p_u) + \nu\sum_{u \in U_2} \log(e^{-B\log(p_u)} + e^{-B\log(1-p_u)})/B$ and

$g(\eta) = \lambda\|\eta\|_1$.

Let the scale variable $v = \frac{y}{\rho}$. The sSSM algorithm is shown in Algorithm 2. It is proved to converge[7] and contains fewer and easier sub-problems than the nSSM algorithm. Lines 7-8 calculate the residuals and Lines 4-5 optimize each parameter alternately.

Algorithm 2 the sSSM Algorithm

Require: X, Y, λ.
Ensure: β, β_0
1: Initialize β, β_0, η, $\rho = 1$, $r = 0$, $s = 0$, $k = 0$.
2: **repeat**
3: Update ρ^{k+1} if necessary.
4: $(\beta^{k+1}, \beta_0^{k+1}) \leftarrow \arg\min_{\beta,\beta_0} h(\beta;\beta_0) + \rho^{k+1}/2\|\beta - \eta^k + v^k\|^2$.
5: $\eta^{k+1} \leftarrow \arg\min_\eta g(\eta) + \rho^{k+1}/2\|\beta^{k+1} - \eta + v^k\|^2$.
6: $v^{k+1} \leftarrow v^k + \rho^{k+1}(\beta^{k+1} - \eta^{k+1})$.
7: $r = \|\beta^{k+1} - \eta^{k+1}\|_2$. #Calculate prime residual
8: $s = \|\rho(\eta^k - \eta^{k+1})\|_2$. #Calculate dual residual
9: $k = k + 1$.
10: **until** convergence.
11: Output β.

1. Update (β, β_0).
The parameters β and β_0 are updated as follows:

$$(\beta^{k+1}, \beta_0^{k+1}) \leftarrow \arg\min_{\beta,\beta_0} h(\beta;\beta_0) + \rho^{k+1}/2\|\beta - \eta^k + v^k\|^2$$

This sub-problem is a convex function combined with ℓ_2-penalty, which is easily solved by FISTA.

2. Update η.
The auxiliary variable η is updated as follows:

$$\eta^{k+1} \leftarrow \arg\min_\eta g(\eta) + \rho^{k+1}/2\|\beta^{k+1} - \eta + v^k\|^2.$$

This is a square loss function with a ℓ_1-penalty. Fortunately, it has a closed-form solution which is given by:

$$\eta^{k+1} = S_{\lambda/\rho}(\beta^{k+1} + v^k)$$

where $S_\kappa(a)$ is a soft thresholding operator:

$$S_\kappa(a) = (a - \kappa)_+ - (-a - \kappa)_+$$

3. Update v.
The Lagrangian multiplier v is updated as follows:

$$v^{k+1} \leftarrow v^k + \rho^{k+1}(\beta^{k+1} - \eta^{k+1}).$$

VI. Experiment

In this section, we evaluate the nSSM and sSSM using a synthetic dataset and a real Twitter dataset. The effectiveness and the efficiency of the nSSM and the sSSM are assessed against several existing methods for different amounts of unlabeled data. All experiments were analyzed in compliance with the Twitter policies[3]. They were conducted on a 64-bit machine with Intel(R) core(TM) processor (i7-6820HQ CPU@ 2.70GHZ) and 16.0GB memory.

[3]https://dev.twitter.com/overview/terms/agreement-and-policy

A. Experimental Setup

1) Synthetic Dataset: The domain-free synthetic dataset is used to illustrate the effectiveness of the nSSM and the sSSM. The synthetic dataset consisted of $10,000$ bags which contained $55,195$ instances. Each bag was composed of at most 10 instances. These instances were generated either by $\mathcal{N}(\mu_1, I)$ or $\mathcal{N}(\mu_2, I)$. $\mathcal{N}(\mu_1, I)$ and $\mathcal{N}(\mu_2, I)$ both denote the multivariate normal distribution with mean and convariance of a 100-dimensional identity matrix I. However, their means are different: the mean of $\mathcal{N}(\mu_1, I)$ or μ_1 is a 100-dimensional zero vector while that of $\mathcal{N}(\mu_2, I)$ or μ_2 is a 100-dimensional vector where every element is 0.3. One bag was labeled as positive if at least one of its instances was generated by $\mathcal{N}(\mu_2, I)$. Otherwise, if all its instances were generated by $\mathcal{N}(\mu_1, I)$, the bag was negative. We kept the proportion of positive labels as about 20% and hence 8026 bags were negative and the remaining 1974 were positive. They were evaluated by 5-fold cross validation.

2) Twitter Dataset: The Twitter data in this paper were retrieved by the following process[34]. First, we queried the Twitter API to obtain the tweets that were potentially related to the topic of flu shot by the query consisting of 113 keywords including "flu", "h1n1" and "vaccine". A total of 11,993,211,616 tweets for the period between Jan 1, 2011 and Apr 15, 2015 in the United States were retrieved. Second, from the retrieved tweet sets, the Twitter users who had indicated flu vaccination were identified by their tweets using the LibShortText[39] text filter that was trained on 10,000 positive and another 10,000 negative tweets provided by Lamb et.al.[22]. The training accuracy of the LibShortText was 92% by the 3-fold cross validation. The full text representation was used as the features in LibShortText. Then, we queried the Twitter API again for those users identified in the second step to obtain their tweets posted within 60 days since their vaccination were identified. Finally, this tweet set formed the Twitter data in this paper, which contained 3,139 users and their 90,185 tweets in total, including 41,438 tweets from 1,572 labeled users where 566 were positive users and 1,006 were negative. The labels of Twitter users provided by[33] were annotated by domain experts. They were evaluated by 5-fold cross-validation. The remaining 1,567 users with their 48,847 tweets were unlabeled, which were used together with our labeled data for model training based on our semi-supervised strategy.

3) Parameter Settings and Metrics: We considered both the nSSM and the sSSM for comparison. Two tuning variables ν and λ are included in two algorithms, which were both set to 1 based on 5-fold cross validation on the training set. Two parameters A and B in the sSSM algorithm were both set to 10. In addition, we highlighted the choice of ρ, which denotes the step size for each iteration. Here we chose three kinds of ρ: (1) $\rho^k = 1$; (2) if $r^k > 4s^k$, $\rho^{k+1} = 2\rho^k$, if $4r^k < s^k$, $\rho^{k+1} = \frac{\rho^k}{2}$ with $\rho^0 = 1$; (3) $\rho^{k+1} = \rho^k + \frac{2}{MAX\_ITER}$ with $\rho^0 = 0$ where MAX_ITER is the maximal iteration. The maximal iteration for the nSSM and the sSSM were set to

2,000 and 20, respectively.

In the experiments, six metrics were utilized to evaluate model performance: the Accuracy (ACC) is the ratio of accurately labeled users to all users; the Precision (PR) is the ratio of accurately labeled as positive users to all labeled as positive users; the Recall (RE) defines the ratio of accurately labeled as positive users to all positive users; the F-score (FS) is the harmonic mean of precision and recall; the Receiver Operating Characteristic (ROC) curve delineates the classification ability of a model as its discrimination threshold varies; and the Area Under ROC (AUC) is an important measurement of classification ability; aside from ROC curve, the Precision Recall (PR) curve is the other one to measure classification performance in which recall and precision are listed as the X axis and the Y axis, respectively; similar to AUC, The Area Under PR curve (AUPR) evaluates classification performance of a classifier.

4) Comparison Methods: The following methods were utilized as baselines for the performance comparison, the first four of which were multi-instance based learning approaches and the last one was semi-supervised learning method. All parameters were set by 5-fold cross-validation on the training set.

1. Constructive Clustering based Ensemble (CCE) [45]. CCE adapted multi-instance learning problems to single-instance learning ones. Each instance in the bag first was clustered into some groups, then a classifier distinguished a bag from others by group information. Many classifiers were generated due to different group numbers. The final step was to ensemble all classifiers together.

2. Multi-instance learning with graph (miGraph)[44]. The miGraph treated instances in the bags as non-independently and identically distributed. It constructed the graph in an implicit manner by considering affinity matrices and defined a new graph kernel which contained the clique information.

3. Multi-instance Learning based on the Vector of Locally Aggregated Descriptors representation (miVLAD)[35]. Multiple instances were mapped into a high dimensional vector by the Vector of Locally Aggregated Descriptors (VLAD) representation. The SVM was applied to train a classifier.

4. Multi-instance Learning based on Fisher Vector representation (miFV)[35]. The miFV was similar to the miVLAD except that multiple instances were encoded by the Fisher Vector (FV) representation.

5. WEakly LabeLed Support Vector Machines (WELLSVM)[24]. WELLSVM is proposed to solve the problem of weakly labeled data by a label generation strategy. As a convex relaxation of Mixed-Integer Programming (MIP) problem, WELLSVM can be solved by a sequence of subproblems to ensure scalability.

B. Performance

In this section, experimental results for both the nSSM and the sSSM are analyzed for all the comparison methods.

1) Model Performance on the Synthetic Dataset: The first part of Table II summarized prediction results of the nSSM and sSSM compared with other methods on the synthetic dataset. Six performance metrics (ACC, PR, RE, FS, AUC and AUPR) were employed to quantify performance. Three choices of ρ were test: $\rho(1)$, $\rho(2)$ and $\rho(3)$ denote the first, second and third choice of ρ, respectively.

The result demonstrated that the nSSM and the sSSM dominated others in the synthetic dataset. Their AUC and AUPR were around 0.94 and 0.85, respectively, whereas the AUC and the AUPR of the miFV and the miVLAD were in the vicinity of 0.78 and 0.56, respectively. The nSSM and the sSSM were superior to others in ACC, which performed 0.07 better than CCE. When it came to the PR, the nSSM and the sSSM attained scores 0.18 higher than that of miFV. As for RE, the scores of the nSSM and the sSSM were far better than others. This was because the max rule had a preference for positive bags and hence reduced bias from imbalanced labels even though positive bags only occupied 20% of the total. Thanks to excellent performance in the PR and RE metrics, the nSSM and the sSSM scored highly in FS. The choice of ρ had little effect on the performance of the nSSM and the sSSM. This fact showed that our nSSM and sSSM were robust to parameter change. The CCE performed the best among the comparison methods in every metric. The performance of the miGraph was unexpected: it suffered from bias from imbalanced labels and all predictions were negative. The WELLSVM achieved an inferior performance compared with four multi-instance learning methods: its ACC was below 0.8 while all four multi-instance learning methods attained ACCs higher than 0.8.

2) Model Performance on the Twitter Dataset: The second part of Table II summarized prediction results of the nSSM and sSSM compared with other methods on the Twitter dataset. 1,567 unlabeled users were used for those semi-supervised ones including the WELLSVM, the nSSM and the sSSM for this comparison.

Similar to the performance in the synthetic dataset, the nSSM and the sSSM performed better than any of the comparison methods. They both exceeded 0.86 and 0.77 in the AUC and AUPR, respectively, no matter what choice of ρ it was, while none of the baseline methods attained over 0.841 and 0.757 in the AUC and AUPR, respectively. When it came to the ACC, the nSSM and the sSSM achieved results that were about 0.1 higher than the miGraph. This is because the max rule eliminated most noisy non- representative tweets. They also performed competitively in the PR metric, surpassing 0.73 while the miVLAD only achieved 0.67. As for the RE, the performance of the nSSM and the sSSM was about 0.2 better than the CCE. High RE indicated that the max rule counteracted the class imbalance from the dataset where the ratio of the negative users to positive ones was $2:1$. When compared with the FS metric, their performance was again competitive, reaching around 0.68, whereas that of the miGraph was only 0.52. The ACCs of the miVLAD and CCE were around 0.76 and their AUPRs were close to 0.71, which

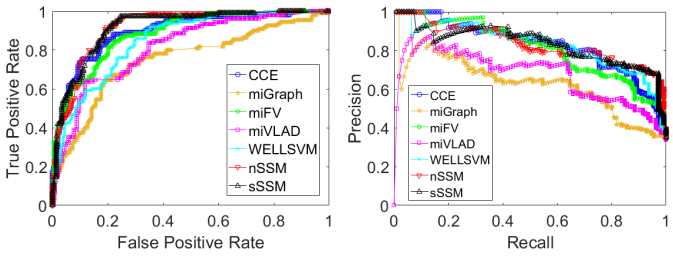
were superior to other baseline methods. The miGraph and the miVLAD were the worst among all methods: their PRs were below 0.7 and their AUCs achieved in the vicinity of 0.75. The performance of the sSSM was similar to the nSSM in six metrics. This fact showed that the smoothness of the objective function had an unnoticeable effect on model performance. Different choices for ρ had a limited effect on the model performance, implying that the nSSM and the sSSM were robust to different choices of ρ: all six measurements remained stable whatever ρ they chose.

Figure 1 showed the ROC and the PR curve of the nSSM, the sSSM and baselines. In the ROC curve, the X axis and the Y axis denote False Positive Rate and True Positive Rate, respectively. In the PR curve, the X axis and the Y axis denote Recall and Precision, respectively. Overall, the ROC curve of the nSSM and the sSSM almost covered baselines in Figure 1(a), which was consistent with Table II. The miFV and the CCE performed similarly, they both covered the miGrpah and miVLAD. The similar patterns were displayed in Figure 1(b) : the nSSM and the sSSM dominated in the PR curve. Different from ROC curve, the CCE performed competitively compared with the nSSM and the sSSM in the PR curve: they overlapped in most regions. The miGraph and the miVLAD still achieved the worst: they were surrounded by four other methods.

TABLE II
MODEL PERFORMANCE ON THE SYNTHETIC DATASET AND THE TWITTER DATASET UNDER SIX METRICS: THE NSSM AND THE SSSM OUTPERFORMED OTHERS.

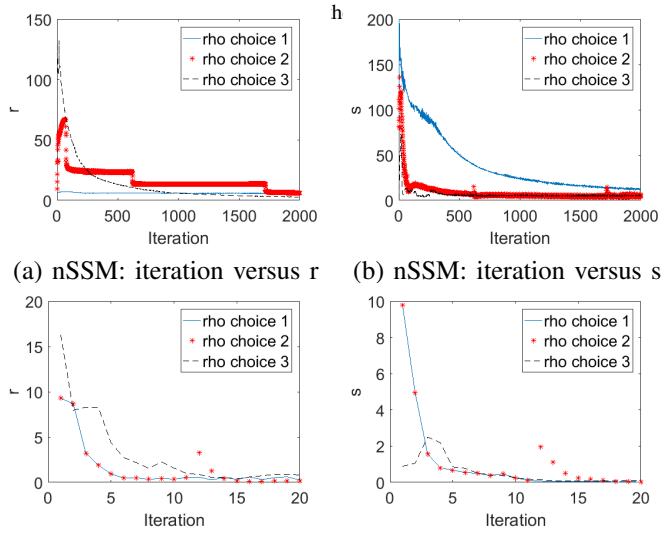
Synthetic Dataset						
Method	ACC	PR	RE	FS	AUC	AUPR
CCE	0.8408	0.8166	0.2494	0.3818	0.9067	0.6991
miGraph	0.8026	0	0	0	0.5	0
miFV	0.8380	0.6998	0.3179	0.4241	0.7875	0.5666
miVLAD	0.8364	0.6634	0.3458	0.4527	0.7796	0.5563
WELLSVM	0.7823	0.4804	0.2978	0.3125	0.7694	0.4284
nSSM(ρ(1))	0.9124	0.8383	**0.6893**	**0.7563**	0.9442	0.8488
nSSM(ρ(2))	0.9105	**0.8631**	0.6510	0.7411	0.9433	0.8471
nSSM(ρ(3))	0.9125	0.8397	0.6883	0.7562	0.9441	0.8486
sSSM(ρ(1))	0.9121	0.8610	0.6622	0.7481	**0.9449**	**0.8504**
sSSM(ρ(2))	0.9125	0.8545	0.6718	0.7515	0.9446	0.8496
sSSM(ρ(3))	**0.9127**	0.8551	0.6723	0.7521	0.9446	0.8496
Twitter Dataset						
Method	ACC	PR	RE	FS	AUC	AUPR
CCE	0.7405	0.7616	0.4124	0.5308	0.8118	0.7136
miGraph	0.7188	0.6779	0.4229	0.5194	0.7415	0.6246
miFV	0.7761	0.7324	0.6010	0.6595	0.8405	0.7564
miVLAD	0.7538	0.6721	0.6217	0.6453	0.7841	0.6806
WELLSVM	0.6985	**0.8635**	0.1942	0.3111	0.8373	0.7271
nSSM(ρ(1))	**0.8015**	0.7813	0.6236	0.6931	**0.8804**	**0.7870**
nSSM(ρ(2))	0.8009	0.7790	0.6234	0.6924	0.8745	0.7827
nSSM(ρ(3))	0.7977	0.7356	**0.6872**	**0.7098**	0.8755	0.7775
sSSM(ρ(1))	0.7901	0.7497	0.6234	0.6802	0.8699	0.7761
sSSM(ρ(2))	0.7913	0.7572	0.6200	0.6810	0.8696	0.7755
sSSM(ρ(3))	0.7939	0.7535	0.6344	0.6876	0.8673	0.7726

3) r and s for Three Choices of ρ: Four subfigures plotted on the prime residual r and the dual residual s with three choices of ρ for the nSSM and the sSSM (Figure 2). As for the nSSM, in Figure 2(a), ρ=(1) reduced r into less than 10 at the 50th iteration, while the r of either of two other choices was higher than 20. The stage-like shape of ρ=(2) indicated that ρ changed several times during iterations. at the the 2000th iteration, three choices of ρ all reduced r into less than 3. In Figure 2(b), things were different from Figure 2(a). s of ρ=(1)



(a) ROC curve (b) PR curve

Fig. 1. The ROC and PR curve of all methods on the Twitter dataset: the h



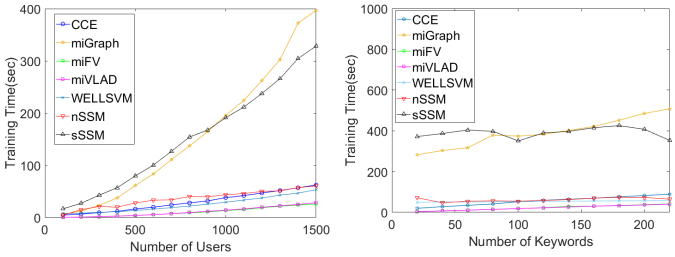
(a) nSSM: iteration versus r (b) nSSM: iteration versus s

(c) sSSM: iteration versus r (d) sSSM: iteration versus s

Fig. 2. r and s for the nSSM and the sSSM: different choices of ρ have different convergence patterns.

was about 50 whereas the other two choices reduced s into less than 10 at the 500th iteration. ρ=(2) and ρ=(3) balanced the decline between r and s from the integration of the above two figures, whereas there was a huge gap between r and s of ρ=(1) before 500 iterations. When it came to the sSSM, three choices of ρ performed similarly. In Figure 2(c) and (d), three curves started where r and s were higher than 8 and 1, respectively, then dropped remarkably near 0 at the 10th iteration. After some fluctuations they achieved convergence at the 20th iteration.

4) Scalability analysis: To examine the scalability of the nSSM and the sSSM, we measured training time of all methods when varying number of users and keywords. The training time was calculated by the average of running 20 times.



(a) Scalability on number of users (b) Scalability on number of keywords

Fig. 3. Scalability on number of keywords and users: almost all methods increased linearly with the number of keywords and users.

TABLE III
TOP-10 MOST FREQUENT KEYWORDS FOR POSITIVE AND NEGATIVE TWEETS ON THE TWITTER DATASET IDENTIFIED BY NSSM.

Rank	Positive Tweets	Negative Tweets
1	headache	bad
2	sore	feeling
3	sick	cold
4	arm	cause
5	throat	face
6	swollen	flu
7	bad	shot
8	flu	heart
9	shot	sick
10	pain	cool

Figure 3(a) compared the running time of all methods when the number of users changed from 100 to 1500. Basically, the running time of almost all methods except miGraph increased linearly with the number of users. The miGraph performed a quadratic-like shape and consumed the most running time. The miFV and the miVLAD were the most efficient methods of all: they completed within 30 seconds with 1500 users and almost overlapped in curves. The CCE and the nSSM were fast in implementation even though they exhausted twice as much time as the miFV and the miVLAD. They overlapped in curves by coincident. When it came to the sSSM, it needed more time than the miGraph when the number of users was less than $1,000$. However, the miGraph surpassed the sSSM with more than $1,000$ users.

To examine the scalability for an increasing number of keywords, Figure 3(b) showed the running time of all methods when the number of keywords ranged from 20 to 220. We also found that running time of all methods increased linearly with the number of keywords. However, the slopes of all curves were much smaller than counterparts in Figure 3(a). This implied that the effect of users on running time was more obvious than that of keywords. Surprisingly, the running time of the nSSM and the sSSM remained stable, which implied that their running time was insensitive to the number of keywords. Especially, the training time of the sSSM fluctuated narrowly between 350 seconds and 400 seconds. The miGraph was the most inefficient method again, which consumed 500 seconds with 220 keywords.

5) Case Studies: We found some interesting keyword patterns which distinguished positive (i.e., adverse-relevant) tweets from negative (i.e., adverse-irrelevant) ones. Table III compared the top-10 most frequent keywords of positive tweets and negative ones identified by nSSM. Several symptom-descriptive keywords such as 'headache', 'sore', 'arm' and 'throat' demonstrated that positive tweets identified by our method were indeed adverse-event relevant. They implied that several common adverse symptoms from flu shots were headaches, arm pain, and throat pain. For negative tweets, several keywords such as 'cool', 'feeling' and 'cause' were general words, which were unrelated to flu shot adverse events. We also found that several keywords such as 'flu','shot' and 'bad' appeared both in positive tweets and negative tweets. This implied that they were not useful for detecting flu shot adverse events.

To further explore semantic patterns in a deep insight, Table IV illustrated five symptoms found in adverse-relevant tweets extracted by nSSM. The first and second column listed symptoms and examples of adverse-relevant tweets, respectively. Keywords were highlighted in bold types. Most of them were pain in a certain organ, such as arm pain, neck pain and headache. Arm pain was the most common symptom because flu shots were put in the arm. Five examples of arm pain showed that these users suffered from pain for a long time. Headache, neck pain and throat pain happened sometimes. The tweet example of fever demonstrated that this user was seriously affected by side effects from flu shots: he or she displayed multiple symptoms including throat pain, nose clog and coughing. Therefore, it is mandatory to detect such serious adverse events in time to prevent the risk of side effects from flu shots.

TABLE IV
REPRESENTATIVE TWEETS AND SYMPTOMS FOUND IN POSITIVE TWEETS EXTRACTED BY NSSM.

Symptoms	Positive Tweets Extracted by nSSM
arm pain	**Flu shot** this afternoon = very **sore arm** this evening
	My **arm** still **sore** from that **flu shot**
	As soon as I walk in my apartment my **arm** decides to remind me I got a **flu shot** today
	I got my **flu shot**. Hate how it **hurts** when they give the **shot** they do it **slow arm hurts** like hell. **hate** doctors and **shots**.
	How does a simple **flu shot** immobilize ones left **arm**? Im **weak** as **hell**... **sore**
headache	got a **flu shot** yesterday and here comes a **headache**.
	Oohh this **headache**... from **flu shots?**
	Flu shot this morn. Now I have a **headache**. ARGH.
neck pain	I got a **flu shot**. Body **aches** are real. The back of my **neck** is **killing** me :(
throat pain	**Flu shot** update: My **throat** continues to feel tight and **clogged**, although not so much that I can't breathe.
	Flu shot dooo Walnut in my **throat** I cant feel my face.
fever	Receive a **flu shot** several days ago, now my **nose** is **clogged**, my **eyes** are **heavy**, my **throat** is so **sore** that I can't talk and I'm so tired, I can't stop **coughing**! small **fever**.

VII. CONCLUSION

Flu shot adverse surveillance is a crucial problem for healthcare. In contrast to traditional adverse event reporting systems with a long time delay, social media provide a promising alternative to detect flu shot adverse events due to timeliness and comprehensiveness. This paper has presented a semi-supervised multi-instance learning model that automatically and effectively addresses noisiness and heterogeneity of the data. Two optimization problems and their corresponding algorithms, namely the nSSM and the sSSM, have been developed to optimized parameters accurately and efficiently. Experimental results demonstrated that the nSSM and the sSSM outperformed other baseline models in six metrics. Case studies showed that our proposed approaches effectively found keyword patterns and five symptoms described in the adverse-relevant tweets.

ACKNOWLEDGEMENT

L. Zhao's work is supported by the NSF under grant # 1755850. Y. Ye's work is partially supported by the NSF under grants CNS-1618629, CNS-1814825 and OAC-1839909, NIJ 2018-75-CX-0032, and HEPC.dsr.18.5.

REFERENCES

[1] Massih-Reza Amini and Patrick Gallinari. Semi-supervised logistic regression. In *Proceedings of the 15th European Conference on Artificial Intelligence*, pages 390–394. IOS Press, 2002.

[2] Jaume Amores. Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence*, 201(4):81–105, 2013.

[3] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. *Advances in Neural Information Processing Systems*, 15(2):561–568, 2002.

[4] R. Ball, M. Robb, S. A. Anderson, and G Dal Pan. The fda's sentinel initiativea comprehensive approach to medical product surveillance. *Clinical Pharmacology Therapeutics*, 99(3):265268, 2015.

[5] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[6] CM Bishop. Pattern recognition and machine learning: springer new york. 2006.

[7] Stephen Boyd, Neal Parikh, Eric Chu, and Borja Peleato. Distributed optimization via alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3:1–122, 2010.

[8] Yi Cai, Jingcheng Du, Jing Huang, Susan S Ellenberg, Sean Hennessy, Cui Tao, and Yong Chen. A signal detection method for temporal variation of adverse effect with vaccine adverse event reporting system data. *BMC medical informatics and decision making*, 17(2):76, 2017.

[9] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[10] Liangzhe Chen, K. S. M. Tozammel Hossain, Patrick Butler, Naren Ramakrishnan, and B. Aditya Prakash. Flu gone viral: Syndromic surveillance of flu on twitter using temporal topic models. In *IEEE International Conference on Data Mining*, pages 755–760, 2014.

[11] Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Semi-supervised learning for neural machine translation. In *Meeting of the Association for Computational Linguistics*, pages 1965–1974, 2016.

[12] David Cohn, Rich Caruana, and Andrew Kachites Mccallum. Semi-supervised clustering with user feedback. *Constrained Clustering*, pages 17–31, 2009.

[13] Thomas G. Dietterich, Richard H. Lathrop, and Toms Lozano-Prez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(12):31–71, 1997.

[14] A Doblas, C Domingo, H. G. Bae, C. L. Bohrquez, Ory F De, M Niedrig, D Mora, F. J. Carrasco, and A Tenorio. Yellow fever vaccine-associated viscerotropic disease and death in spain. *Journal of Clinical Virology*, 36(2):156–158, 2006.

[15] David A Freedman. *Statistical models: theory and practice*. cambridge university press, 2009.

[16] Clark C Freifeld, John S Brownstein, Christopher M Menone, Wenjie Bao, Ross Filice, Taha Kass-Hout, and Nabarun Dasgupta. Digital drug safety surveillance: monitoring pharmaceutical products in twitter. *Drug safety*, 37(5):343–350, 2014.

[17] M. Guillaumin, J. Verbeek, and C. Schmid. Multimodal semi-supervised learning for image classification. In *Computer Vision and Pattern Recognition*, pages 902–909, 2010.

[18] S. M. Hwang, K. W. Choe, S. H. Cho, S. J. Yoon, D. E. Park, J. S. Kang, M. J. Kim, B. C. Chun, and S. M. Lee. The adverse events of influenza a (h1n1) vaccination and its risk factors in healthcare personnel in 18 military healthcare units in korea. *Japanese Journal of Infectious Diseases*, 64(3):183–9, 2011.

[19] Hayate Iso, Shoko Wakamiya, and Eiji Aramaki. Forecasting word model: Twitter-based influenza surveillance and prediction. In *Proceedings of the 26th International Conference on Computational Linguistics*, pages 76–86, 2016.

[20] L. K. Real-time disease surveillance using twitter data: demonstration on flu and cancer. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1474–1477, 2013.

[21] Anurag Kumar and Bhiksha Raj. Audio event detection using weakly labeled data. 2016.

[22] Alex Lamb, Michael J Paul, and Mark Dredze. Separating fact from fear: Tracking flu infections on twitter. In *HLT-NAACL*, pages 789–795, 2013.

[23] Vasileios Lampos, Tijl De Bie, and Nello Cristianini. Flu detector-tracking epidemics on twitter. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 599–602. Springer, 2010.

[24] Yu-Feng Li, Ivor W Tsang, James T Kwok, and Zhi-Hua Zhou. Convex and scalable weakly labeled svms. *The Journal of Machine Learning Research*, 14(1):2151–2188, 2013.

[25] Alejandro Metke-Jimenez, Sarvnaz Karimi, and Cecile Paris. Evaluation of text-processing algorithms for adverse drug event extraction from social media. In *International Workshop on Social Media Retrieval and Analysis*, pages 15–20, 2014.

[26] Tanushree Mitra, Scott Counts, and James W Pennebaker. Understanding anti-vaccination attitudes in social media. In *ICWSM*, pages 269–278, 2016.

[27] Michael J Paul and Mark Dredze. You are what you tweet: Analyzing twitter for public health. *Icwsm*, 20:265–272, 2011.

[28] Marcel Salathé and Shashank Khandelwal. Assessing vaccination sentiments with online social media: implications for infectious disease dynamics and control. *PLoS computational biology*, 7(10):e1002199, 2011.

[29] Tom T Shimabukuro, Michael Nguyen, David Martin, and Frank DeStefano. Safety monitoring in the vaccine adverse event reporting system (vaers). *Vaccine*, 33(36):4398–4405, 2015.

[30] Alessio Signorini, Alberto Maria Segre, and Philip M Polgreen. The use of twitter to track levels of disease activity and public concern in the us during the influenza a h1n1 pandemic. *PloS one*, 6(5):e19467, 2011.

[31] Fei Wang, Tao Li, and Changshui Zhang. Semi-supervised clustering via matrix factorization. In *Siam International Conference on Data Mining, SDM 2008, April 24-26, 2008, Atlanta, Georgia, Usa*, pages 1–12, 2008.

[32] Huiling Wang and Tinghuai Wang. Boosting objectness: Semi-supervised learning for object detection and segmentation in multi-view images. In *ICASSP*, 2016.

[33] Junxiang Wang and Liang Zhao. Multi-instance domain adaptation for vaccine adverse event detection. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 97–106, 2018.

[34] Junxiang Wang, Liang Zhao, Yanfang Ye, and Yuji Zhang. Adverse event detection by integrating twitter data and vaers. *Journal of Biomedical Semantics*, 9(1):19, Jun 2018.

[35] Xiu Shen Wei, Jianxin Wu, and Zhi Hua Zhou. Scalable multi-instance learning. In *IEEE International Conference on Data Mining*, pages 1037–1042, 2014.

[36] Ryen W White, Nicholas P Tatonetti, Nigam H Shah, Russ B Altman, and Eric Horvitz. Web-scale pharmacovigilance: listening to signals from the crowd. *Journal of the American Medical Informatics Association Jamia*, 20(3):404, 2013.

[37] Xin Xu and Eibe Frank. Logistic regression and boosting for labeled bags of instances. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 272–281. Springer, 2004.

[38] E Yomtov and E Gabrilovich. Postmarket drug surveillance without trial costs: Discovery of adverse drug reactions through large-scale analysis of web search queries. *Journal of Medical Internet Research*, 15(6):e124, 2013.

[39] Hsiangfu Yu. Libshorttext: A library for short-text classification and analysis libshorttext: A library for short-text classification and analysis. 2013.

[40] Zixing Zhang, Fabien Ringeval, Bin Dong, Eduardo Coutinho, Erik Marchi, and Bjorn Schuller. Enhanced semi-supervised learning for multimodal emotion recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5185–5189, 2016.

[41] Liang Zhao, Junxiang Wang, Feng Chen, Chang Tien Lu, and Naren Ramakrishnan. Spatial event forecasting in social media with geographically hierarchical regularization. *Proceedings of the IEEE*, 105(10):1953–1970, 2017.

[42] Liang Zhao, Junxiang Wang, and Xiaojie Guo. Distant-supervision of heterogeneous multitask learning for social event forecasting with multilingual indicators. In *AAAI Conference on Artificial Intelligence*, 2017.

[43] Zhi Hua Zhou, Yu Yin Sun, and Yu Feng Li. Multi-instance learning by treating instances as non-i.i.d. samples. *Computer Science*, pages 1249–1256, 2008.

[44] Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li. Multi-instance learning by treating instances as non-iid samples. In *Proceedings of the 26th annual international conference on machine learning*, pages 1249–1256. ACM, 2009.

[45] Zhi-Hua Zhou and Min-Ling Zhang. Solving multi-instance problems with classifier ensemble based on constructive clustering. *Knowledge and Information Systems*, 11(2):155–170, 2007.

[46] J Zowe. Nondifferentiable optimization. In *Computational Mathematical Programming*, pages 323–356. Springer, 1985.