Exploiting Social Relations for Sentiment Analysis in Microblogging

Xia Hu[†], Lei Tang[‡], Jiliang Tang[†], Huan Liu[†]
[†]Arizona State University, Tempe, AZ 85287, USA
[‡]@WalmartLabs, San Bruno, CA 94066, USA
{xiahu, jiliang.tang, huan.liu}@asu.edu, leitang@acm.org

ABSTRACT

Microblogging, like Twitter[1], has become a popular platform of human expressions, through which users can easily produce content on breaking news, public events, or products. The massive amount of microblogging data is a useful and timely source that carries mass sentiment and opinions on various topics. Existing sentiment analysis approaches often assume that texts are independent and identically distributed (i.i.d.), usually focusing on building a sophisticated feature space to handle noisy and short messages, without taking advantage of the fact that the microblogs are networked data. Inspired by the social sciences findings that sentiment consistency and emotional contagion are observed in social networks, we investigate whether social relations can help sentiment analysis by proposing a Sociological Approach to handling Noisy and short Texts ($SANT$) for sentiment classification. In particular, we present a mathematical optimization formulation that incorporates the sentiment consistency and emotional contagion theories into the supervised learning process; and utilize sparse learning to tackle noisy texts in microblogging. An empirical study of two real-world Twitter datasets shows the superior performance of our framework in handling noisy and short tweets.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Classification*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing

General Terms

Algorithm, Performance, Experimentation

Keywords

Sentiment Classification, Microblogging, Twitter, Noisy and Short Texts, Social Context, Social Correlation

[1] `https://twitter.com/`

WSDM '13, February 4–8, 2013, Rome, Italy.

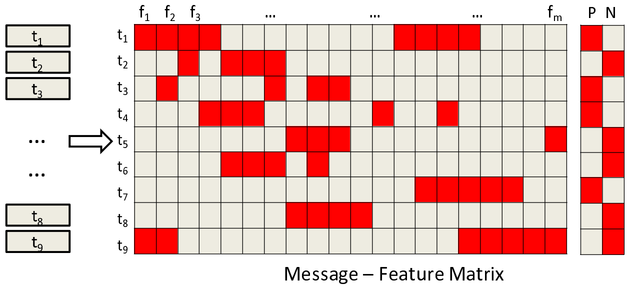
1. INTRODUCTION

Microblogging services are extensively used to share information or opinions in various domains. With the growing availability of such an opinion-rich resource, it attracts much attention from those who seek to understand the opinions of individuals, or to gauge aggregated sentiment of mass populations. For example, advertisers may want to target users who are enthusiastic about a brand or a product in order to launch a successful social media campaign. Aid agencies from around the world would like to monitor sentiment evolutions before, during, and after crisis to assist recovery and provide disaster relief. The sheer volumes of microblogging data present opportunities and challenges for sentiment analysis of these noisy and short texts.

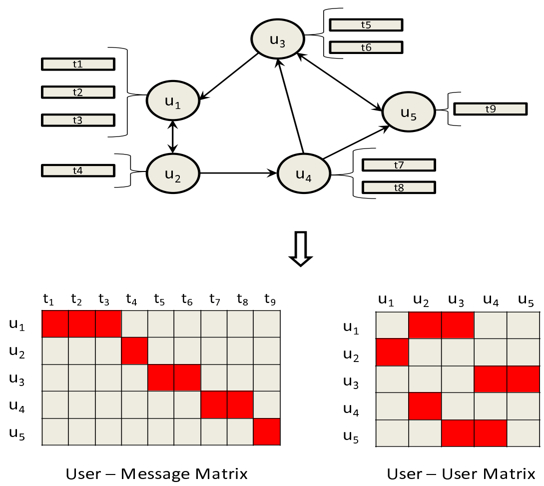
Sentiment analysis has been extensively studied for product and movie reviews [32], which differ substantially from microblogging data. Unlike standard texts with many words that help gather sufficient statistics, the texts in microblogging only consist of a few phrases or 1-2 sentences. Also, when composing a microblogging message, users may use or coin new abbreviations or acronyms that seldom appear in conventional text documents. For example, messages like "It is coooooool", "OMG :-(", are intuitive and popular in microblogging, but some are not formal words. It is difficult for machines to accurately identify the semantic meanings of these messages, though they provide convenience in quick and instant communications for human beings. Existing methods [2, 8] rely on pre-defined sentiment vocabularies [39], which are highly domain-specific.

Meanwhile, microblogging platforms often provide additional information other than text. For example, in Figure 1, we depict two kinds of data available in microblogging. Figure 1(a) shows the content of messages, in the form of a message-feature matrix. Traditional methods measure the similarity between text documents (messages) purely based on content information. A distinct feature of microblogging messages is that they are potentially networked through user connections, which may contain useful semantic clues that are not available in purely text-based methods. Besides content information, relations between messages can be represented via a user-message matrix and a user-user interaction matrix, as shown in Figure 1(b). Traditional methods, if applied directly to the microblogging data, do not utilize the social relation information.

In social sciences, it is well-established that emotions and sentiments play a distinct role in our social life and correlate with our social connections. When experiencing emotions, people do not generally keep the emotions to themselves,



(a) Data Representation of Message Content



(b) Data Representation of Social Relations

Figure 1: Data Representation of Text and Social Relation Information in Microblogging

but rather, they tend to *show* them [19]. Also, people tend to "catch" others' emotions as a consequence of facial, vocal, and postural feedback, which has been recognized as *emotional contagion* [10] in social sciences. Emotional contagion may be important in personal relationships because "it fosters behavioral synchrony and the tracking of the feelings of others moment-to-moment even when individuals are not explicitly attending to this information" [10]. As a consequence of emotional contagion, Fowler and Christakis [6] reported the spread of happiness in a social network. Two social processes, selection and influence, are proposed to explain the phenomenon [22]: people befriend others who are similar to them (*Homophily* [26]), or they become more similar to their friends over time (*Social Influence* [25]). Both explanations suggest that connected individuals are more likely to have similar behaviors or hold similar opinions. Inspired by this sociological observation, we explore the utilization of social relation information to facilitate sentiment analysis in the context of microblogging.

In this paper, we aim to provide a supervised approach to sentiment analysis in microblogging by taking advantage of social relation information in tackling the noisy nature of the messages. In particular, we first investigate whether the social theories exist in microblogging data. Then we discuss how the social relations could be modeled and utilized for supervised sentiment analysis. Finally, we conduct extensive experiments to verify the proposed model. The main contributions of this paper are as follows:

- We formally define the problem of sentiment analysis in microblogging to enable the utilization of social relations for sentiment analysis;
- By verifying the existence of two social theories in microblogging, we build sentiment relations between messages via social relations;
- We present a novel supervised method to tackle the noisy and short texts by integrating sentiment relations between the texts; and
- We empirically evaluate the proposed *SANT* framework on real-world Twitter datasets and elaborate the effects of social relationships on sentiment analysis.

The remainder of this paper is organized as follows. In Section 2, we review related work. In Section 3, we formally define the problem we study. In Section 4, we conduct a study to verify the social theories. In Section 5, we propose a novel framework *SANT* for supervised sentiment analysis. In Section 7, we report empirical results. We conclude and present the future work in Section 8.

2. RELATED WORK

Recently, sentiment analysis on microblogging, which is considered to be an opinion-rich resource, has gained huge popularity and attracted researchers from many disciplines [3, 18, 20, 30]. Bollen et al. [3] proposed to measure the sentiments on Twitter over time, and compared the correlation between sentiments and major events, including the stock market, crude oil prices, elections and Thanksgiving. Also, Kim et al. [20] examined a tweet dataset about Michael Jackson's death to gain insight into how emotion is expressed on Twitter. O'Connor et al. [30] used sentiment analysis to automatically label the sentiments of tweets about politicians, and found strong correlation between the aggregated sentiment and the manually collected poll ratings.

Sentiment classification has been studied for years on various text corpus, like newspaper articles [33], movie reviews [31], and product reviews [5, 11, 23]. The basic idea of the methods is to build a sophisticated feature space, which can effectively represent the sentiment status of the texts. Existing methods, which are designed for traditional i.i.d. text data, cannot effectively make use of the abundant social relation information contained in microblogging. Following the methods for traditional texts, there are some existing efforts in the community on the microblogging data. Alec et al. [8] presented the results of machine learning algorithms for classifying the sentiments of Twitter messages using distant supervision. Barbosa and Feng [2] explored the linguistic characteristics of how tweets are written and the meta-information of words for sentiment classification. The ideas of the methods are consistent with traditional ones, ignoring the social relation information.

Some efforts have been made to explore the effect of external information sources [37, 41], especially social network information [35, 36], on sentiment analysis. Speriosu et al. [35] proposed to incorporate labels from a maximum entropy classifier, in combination with the Twitter follower graph. They simply used a user's followers as separate features and combined them with the content matrix. Tan et al. [36] proposed to improve user-level sentiment analysis of different topics [16] by incorporating social network information. However, our task is document-level sentiment classification, which has finer granularity than their work. In addition, our method simultaneously utilizes social relation information and handles noisy, short texts in microblogging.

3. PROBLEM STATEMENT

The problem we study in this paper is different from traditional sentiment classification since the latter normally only considers the content information. In this section, we first present the notations and then formally define the problem of sentiment classification on microblogging messages.

We use boldface uppercase letters (e.g., $\mathbf{A}$) to denote matrices, boldface lowercase letters (e.g., $\mathbf{a}$) to denote vectors, and lowercase letters (e.g., a) to denote scalars. The entry at the i^{th} row and j^{th} column of a matrix $\mathbf{A}$ is denoted as $\mathbf{A}_{ij}$. $\mathbf{A}_{i*}$ and $\mathbf{A}_{*j}$ denote the i^{th} row and j^{th} column of a matrix $\mathbf{A}$, respectively. $\|\mathbf{A}\|_1$ is the ℓ_1-norm and $\|\mathbf{A}\|_F$ is the Frobenius norm of matrix $\mathbf{A}$. Specifically, $\|\mathbf{A}\|_1 = \sum_{i=1}^{m}\sum_{j=1}^{n}|\mathbf{A}_{ij}|$ and $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|\mathbf{A}_{ij}|^2}$.

Given a corpus $\mathbf{T} = [\mathbf{X}, \mathbf{Y}]$, where $\mathbf{X} \in \mathbb{R}^{m\times n}$ is the content matrix, $\mathbf{Y} \in \mathbb{R}^{n\times c}$ is the sentiment label matrix, m is the number of features, n is the number of messages and c is number of sentiments, as shown in Figure 1(a). For each message in the corpus $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_n\}$, $\mathbf{t}_i = (\mathbf{x}_i, \mathbf{y}_i) \in \mathbb{R}^{m+c}$ consists of microblogging message content and sentiment label, where $\mathbf{x}_i \in \mathbb{R}^m$ is the message feature vector and $\mathbf{y}_i \in \mathbb{R}^c$ is the sentiment label vector. Following previous work [8, 35], in this paper, we focus on polarity sentiment classification, i.e., $c = 2$. It is practical to extend this setting to a multi-class sentiment classification task. $\mathbf{u} = \{u_1, u_2, \ldots, u_d\}$ is the user set, where d is the number of distinct users in the corpus. $\mathbf{U} \in \mathbb{R}^{d\times n}$ is a user-message matrix, as shown as the left matrix in Figure 1(b). In the user-message matrix, $\mathbf{U}_{ij} = 1$ (red frame in the figure) denotes that message $\mathbf{t}_j$ is posted by user u_i. $\mathbf{F} \in \mathbb{R}^{d\times d}$ is the user-user matrix, as shown in Figure 1(b). In the matrix, $\mathbf{F}_{ij} = 1$ (red frame in the figure) indicates that user u_i is connected by user u_j.

With the notations above, we formally define sentiment classification of microblogging messages as:

Given a corpus of microblogging messages $\mathbf{T}$ *with their content* $\mathbf{X}$ *and corresponding sentiment labels* $\mathbf{Y}$*, social relations for this corpus including the user-message relation* $\mathbf{U}$*, and user-user following relation* $\mathbf{F}$*, we aim to learn a classifier* $\mathbf{W}$ *to automatically assign sentiment labels for unseen messages (i.e., test data).*

4. DATA AND OBSERVATIONS

Before we proceed to our solution, in this section, we first introduce real-world data used in this work and present some explorations whether social theories have any impacts on sentiment analysis.

4.1 Data

Subsets of two publicly available Twitter datasets are employed: Stanford Twitter Sentiment (STS) and Obama-McCain Debate (OMD). Both datasets consist of raw tweets with their corresponding sentiment labels.

Table 1: Statistics of the Datasets

	STS	*OMD*
# of Tweets	22,262	1,827
# of Users	8,467	735
Max Degree of the Users	897	138
Min Degree of the Users	1	1
Avg. Tweets per User	2.63	2.49

The first dataset is the Stanford Twitter Sentiment (STS)[2]. Go et al. [8] created a collection of 40216 tweets with polarity sentiment labels to train a sentiment classifier. However, it lacks social network information among users in this dataset. We further refined the Twitter dataset according to authors' social relation information, which is the complete follower graph[3] crawled by Kwak et al. [21] during July 2009. According to the social network, we filter tweets whose authors have no friends or have published fewer than two tweets. Finally, it leaves a corpus of 22,262 tweets that consists of 11959 positive tweets and 10303 negative ones.

The second dataset is the Obama-McCain Debate (OMD)[4]. This dataset consists of 3,269 tweets posted during the presidential debate on September 26, 2008 between Barack Obama and John McCain [34]. The sentiment label of each tweet was annotated through Amazon Mechanical Turk[5]. Each tweet was manually labeled by at least three Turkers. In our experiment of polarity sentiment classification, we use tweets with sentiment labels. The majority of votes for a tweet is taken as a gold standard. In order to obtain the social relation information, we use the same social network as in refining the STS dataset. All the tweets whose authors have no friends or have published fewer than two tweets are filtered. This results in a corpus of 1,827 tweets, in which 747 have positive labels and 1080 have negative labels.

The statistics of the datasets are summarized in Table 1.

4.2 Social Theories in Microblogging

In social sciences, Fowler and Christakis [6] found that the spread of happiness appears to reach up to three degrees of separation in a social network. Recently, researchers reported the phenomenon of sentiment diffusion [27] in online social networks based on the theory of emotional contagion [10] between friends. The analysis indicates that, in terms of sentiment, social theories such as *Sentiment Consistency* [1] and *Emotional Contagion* [10] could be helpful for sentiment analysis. Sentiment Consistency suggests that the sentiments of two messages posted by the same user are more likely to be consistent than those of two randomly selected messages. Emotional Contagion reveals that the sentiments of two messages posted by friends are more likely to be similar than those of two randomly selected messages.

The two theories are derived from offline surveys and conversations. We would like to validate whether the two social theories hold true in microblogging data. For each of the theories, we form a null hypothesis: in terms of sentiment, there is no difference between relational data and random data. We test the hypotheses on each of the two datasets.

[2] http://www.stanford.edu/~alecmgo/cs224n/
[3] http://an.kaist.ac.kr/traces/WWW2010.html/
[4] https://bitbucket.org/speriosu/updown/src/5de483437466/data/
[5] https://www.mturk.com/

The sentiment difference score between two messages is defined as $\mathbf{T}_{ij} = ||\mathbf{y}_i - \mathbf{y}_j||_2$, where $\mathbf{y}_i$ is the sentiment label of message $\mathbf{x}_i$. To verify the existence of Sentiment Consistency, we construct two vectors $\mathbf{sc}_t$ and $\mathbf{sc}_r$ with equal number of elements. Each element of the first vector $\mathbf{sc}_t$ is obtained by calculating the sentiment difference score between $\mathbf{x}_i$ and $\mathbf{x}_j$, which are posted by the same user. Each element in the vector corresponds to a pair of related messages. The element of the second vector represents the sentiment difference score between $\mathbf{x}_i$ and another random message $\mathbf{x}_r$ in the corpus. We perform a two-sample t-test on the two vectors $\mathbf{sc}_t$ and $\mathbf{sc}_r$. The null hypothesis is that there is no difference between the two vectors, $H_0 : \mathbf{sc}_t = \mathbf{sc}_r$; the alternative hypothesis is that the sentiment difference between messages with Sentiment Consistency relation is less than those without, $H_1 : \mathbf{sc}_t < \mathbf{sc}_r$. Similarly, we construct another two vectors $\mathbf{ec}_t$ and $\mathbf{ec}_r$, and perform a two-sample t-test on the two vectors for verifying Emotional Contagion. The null hypothesis is $H_0 : \mathbf{ec}_t = \mathbf{ec}_r$ and the alternative hypothesis is $H_1 : \mathbf{ec}_t < \mathbf{ec}_r$. The t-test results, p-values, show that there is strong evidence (with the significance level $\alpha = 0.01$) to reject the null hypothesis in both tests on the two datasets. In other words, we observe the existence of what Sentiment Consistency and Emotional Contagion suggest in microblogging data. This preliminary study paves the way for our next study: how to explicitly model and utilize these social theories for sentiment classification task.

5. A SOCIOLOGICAL APPROACH – SANT

In this section, we first introduce data representation and modeling for message content, and then discuss how we model relations between messages based on the social theories. Finally, we present how to employ sparse learning to handle noisy and high-dimensional data.

5.1 Modeling Message Content

To find a better text representation for sentiment analysis, Pang and Lee [33] conducted experiments to investigate the effectiveness of different features on sentiment classification. Their two major findings are (1) although different feature construction methods, like N-grams, Part of Speech, adjectives, sentiment vocabulary, have comparable performance, the unigram model with term presence (but not frequency) as feature weight achieves the best results; and (2) no stemming or stop-word lists are used because some of them may carry sentiment information. Thus, we employ the unigram model to construct our feature space, use term presence as the feature weight, and do not perform stemming or remove stop-words. It is noted that our framework is not confined to the unigram model. We can also use other text representation methods for specific sentiment classification tasks.

The widely used method Least Squares is employed to fit the learned model to message content. In terms of multi-class classification problems, the Least Squares aims to learn c classifiers by solving the following optimization problem:

$$\min_{\mathbf{W}} \quad \frac{1}{2}\|\mathbf{X}^T\mathbf{W} - \mathbf{Y}\|_F^2, \tag{1}$$

where $\mathbf{W}$ represents the learned classifiers. This formulation is a traditional supervised classification method, where the messages are assumed to be independent and identically distributed. This method has been well studied, and it has a closed-form solution.

5.2 Modeling Message-Message Relations

In this subsection, we introduce our formulation to utilize social relations for sentiment analysis, and answer the question: "How can social relations be explicitly integrated into the sentiment classification framework?".

In order to transform user-centric social relations into sentiment relations between messages, we employ the social theories discussed in Section 4.2. Given the user-message matrix $\mathbf{U}$ and user-user matrix $\mathbf{F}$, the message-message sentiment relation matrix for *Sentiment Consistency* ($\mathbf{A}sc$) is defined as $\mathbf{A}sc = \mathbf{U}^T \times \mathbf{U}$, where $\mathbf{A}sc_{ij} = 1$ indicates that $\mathbf{t}_i$ and $\mathbf{t}_j$ are posted by the same user, and sentiments of the two messages are similar. The message-message sentiment relation matrix for *Emotional Contagion* ($\mathbf{A}ec$) is defined as $\mathbf{A}ec = \mathbf{U}^T \times \mathbf{F} \times \mathbf{U}$, where $\mathbf{A}ec_{ij} = 1$ indicates that the author of $\mathbf{t}_i$ is a friend of the author who wrote $\mathbf{t}_j$, and sentiments of the two messages are similar. The following derivations in this paper are based on the message-message sentiment relation matrix $\mathbf{A}$, which can be obtained as either the sentiment relation $\mathbf{A}sc$, $\mathbf{A}ec$, or the combination $\mathbf{A} = \mathbf{A}sc + \theta\mathbf{A}ec$, where θ controls the weight of two different sentiment relations in the model. In this paper, we focus on studying the effects of different sentiment relations on the sentiment classification performance, but not ways to combine them. We can simply combine these two relations with equal weight $\theta = 1$ to construct a relation matrix.

Based on the discussion above, to integrate sentiment relations between messages in sentiment classification, the basic idea is to build a latent connection to make two messages as close as possible if they are posted by the same user (*Sentiment Consistency*) or two users are follower/friend with each other (*Emotional Contagion*). Under this scenario, it can be mathematically formulated as solving the following objective function.

$$\begin{aligned} &\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{A}_{ij}\|\hat{\mathbf{Y}}_{i*} - \hat{\mathbf{Y}}_{j*}\|^2 \\ =& \sum_{k=1}^{c}\hat{\mathbf{Y}}_{*k}^T(\mathbf{D} - \mathbf{A})\hat{\mathbf{Y}}_{*k} \\ =& \; tr(\mathbf{W}^T\mathbf{X}\mathcal{L}\mathbf{X}^T\mathbf{W}), \end{aligned} \tag{2}$$

where $tr(\cdot)$ is the trace of a matrix. $\hat{\mathbf{Y}} = \mathbf{X}^T\mathbf{W}$ is the fitted value of the sentiment label $\mathbf{Y}$. $\mathcal{L} = \mathbf{D} - \mathbf{A}$ is the Laplacian matrix [4], where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a message-message sentiment relation matrix to represent a direct graph. $\mathbf{A}_{ij} = 1$ indicates that message t_i has relation with message t_j, and $\mathbf{A}_{ij} = 0$ otherwise. $\mathbf{D} \in \mathbb{R}^{n\times n}$ is a diagonal matrix with $\mathbf{D}_{ii} = \sum_{j=1}^{n}\mathbf{A}_{ij}$ indicating its diagonal element is the degree of a message in relation matrix $\mathbf{A}$.

As Laplacian matrix $\mathcal{L}$ is positive semi-definite, Eq. (2) can be rewritten as:

$$\begin{aligned} &tr(\mathbf{W}^T\mathbf{X}\mathcal{L}\mathbf{X}^T\mathbf{W}) \\ &= \|\mathbf{W}^T\mathbf{X}\mathcal{L}^{\frac{1}{2}}\|_F^2, \end{aligned} \tag{3}$$

For different message-message relations, we use different sentiment relation matrices $\mathbf{A}$ to obtain the Laplacian matrices $\mathcal{L}$. The optimization formulation, which integrates sentiment relations into the learning process, is defined as:

$$\min_{\mathbf{W}} \quad \frac{1}{2}||\mathbf{X}^T\mathbf{W} - \mathbf{Y}||_F^2 + \frac{\alpha}{2}\|\mathbf{W}^T\mathbf{X}\mathcal{L}^{\frac{1}{2}}\|_F^2, \tag{4}$$

where α is the regularization parameter to control the contribution of sentiment relation information.

5.3 Handling the Noisy & Short Texts – A Sparse Formulation

Compared with texts in traditional media, another distinct feature of texts in microblogging is that they are noisy and short [12, 14, 38], which leads to two problems. First, text representation models, like "bag of words" or n-gram, often lead to a high-dimension feature space because of the large-scale size of the dataset and vocabulary. Second, the short and noisy texts make the data representation very sparse. This high-dimension sparse representation poses significant challenges to building an interpretable model with high prediction accuracy.

To retain original information in the texts, we do not filter the terms according to any domain-specific sentiment vocabularies. When people speed-read through a text, they may not fully parse the sentence but instead seek a sparse representation for the incoming text using a few phrases or words [13]. Thus, we propose to provide a sparse reconstruction for the classification feature space. Recently, sparse regularization has been widely used in many data mining applications to obtain more stable and interpretable models. A natural approach for our problem is the lasso [7], the penalization of the ℓ_1-norm of the estimator. The ℓ_1-norm based linear reconstruction error minimization can lead to a sparse representation for the texts, which is robust to the noise in features. The multi-class classifier can be learned by solving the following optimization problem.

$$\min_{\mathbf{W}} \quad \frac{1}{2}\|\mathbf{X}^T\mathbf{W} - \mathbf{Y}\|_F^2 + \beta\|\mathbf{W}\|_1, \tag{5}$$

where β is the sparse regularization parameter. In the objective function, the first term is least squares loss. The second term is ℓ_1-norm regularization on weight matrix $\mathbf{W}$, which causes some of the coefficients to be exactly zero. Thus the lasso does a kind of continuous subset selection and also controls the complexity of the model. Further, we introduce the Laplacian regularization discussed in Section 5.2 to Eq. (5). The sentiment classification of microblogging data can be formulated as the following optimization problem.

$$\begin{aligned}\min_{\mathbf{W}} \quad & \frac{1}{2}||\mathbf{X}^T\mathbf{W} - \mathbf{Y}||_F^2 + \frac{\alpha}{2}\|\mathbf{W}^T\mathbf{X}\mathcal{L}^{\frac{1}{2}}\|_F^2 \\ & + \beta||\mathbf{W}||_1,\end{aligned} \tag{6}$$

where α and β are postive regularization parameters. By solving Eq. (6), the sentiment label of each message can be predicted by

$$\arg\max_{i\in\{p,n\}} \quad \mathbf{x}^T\mathbf{w}_i. \tag{7}$$

Next, we introduce an efficient algorithm to solve the optimization problem in Eq. (6).

6. ALGORITHMIC DETAILS

Since $\|\mathbf{W}\|_1$ is non-differentiable, the proposed objective function in Eq. (6) is non-smooth. In this section, we introduce an efficient algorithm to solve the optimization problem, and discuss its convergence rate and time complexity.

6.1 Optimization Algorithm for *SANT*

Motivated by [24, 29, 9], we propose to solve the non-smooth optimization problem in Eq. (6) by optimizing its equivalent smooth convex reformulations.

THEOREM 1. *Eq. (6) can be reformulated as a constrained smooth convex optimization problem:*

$$\underset{W\in\mathcal{Z}}{min} \;\; f(\mathbf{W}) = \frac{1}{2}\|\mathbf{X}^T\mathbf{W} - \mathbf{Y}\|_F^2 + \frac{\alpha}{2}\|\mathbf{W}^T\mathbf{X}\mathcal{L}^{\frac{1}{2}}\|_F^2, \tag{8}$$

where,

$$\mathcal{Z} = \{\mathbf{W}| \;\; \|\mathbf{W}\|_1 \le z\}, \tag{9}$$

$z \ge 0$ *is the radius of the* ℓ_1*-ball, and there is a one-to-one correspondence between* β *and* z*.*

REMARK 1. *The relationship between β and z is not critical because the optimal values of both are unknown. The two parameters are usually tuned using cross-validation. A sufficiently small z will cause some of the coefficients to be exactly zero, thus it also does a kind of continuous subset selection.*

Proof. The Hessian matrix of the objective function in Eq. (8) is positive semi-definite, thus the objective function $f(\mathbf{W})$ is convex and differentiable. It is easy to verify that $\|\mathbf{W}\|_1$ is a valid norm because it satisfies the three norm conditions, including the triangle inequality $\|\mathbf{A}\|_1 + \|\mathbf{B}\|_1 \le \|\mathbf{A}+\mathbf{B}\|_1$. Since any norm defines a convex set, $\mathcal{Z}$ is a closed and convex set.

As we can see, our problem defines a convex and differentiable function $f(\mathbf{W})$ in a closed and convex set $\mathcal{Z}$. Thus this problem is a constrained smooth convex optimization problem, which completes the proof. □

We first consider the optimization problem in Eq. (8) without the constraint part $W \in \mathcal{Z}$, and it is defined as:

$$\min_{\mathbf{W}} f(\mathbf{W}). \tag{10}$$

It is known that, in gradient descent method, $\mathbf{W}_{t+1}$ is updated in each step as:

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \frac{1}{\lambda_t}\nabla f(\mathbf{W}_t) \tag{11}$$

where λ_t is the step size, which is determined by the line search according to the Armijo-Goldstein rule. The smooth part of the optimization problem can be reformulated equivalently as a proximal regularization [17] of the linearized function $f(\mathbf{W})$ at $\mathbf{W}_t$, which is formally defined as:

$$\mathbf{W}_{t+1} = \arg\min_{\mathbf{W}} G_{\lambda_t,\mathbf{W}_t}(\mathbf{W}), \tag{12}$$

where,

$$\begin{aligned}G_{\lambda_t,\mathbf{W}_t}(\mathbf{W}) = & f(\mathbf{W}_t) + \langle\nabla f(\mathbf{W}_t), \mathbf{W} - \mathbf{W}_t\rangle \\ & + \frac{\lambda_t}{2}\|\mathbf{W} - \mathbf{W}_t\|_F^2.\end{aligned} \tag{13}$$

Considering the equivalence relationship and the constraints $\mathcal{Z}$, we propose to solve Eq. (8) through the following iterative step,

$$\mathbf{W}_{t+1} = \arg\min_{\mathbf{W}\in\mathcal{Z}} G_{\lambda_t,\mathbf{W}_t}(\mathbf{W}), \tag{14}$$

By ignoring terms that are independent of $\mathbf{W}$ in Eq. (13), the objective function in Eq. (14) boils down to:

$$\mathbf{W}_{t+1} = \arg\min_{\mathbf{W}\in\mathcal{Z}} \|\mathbf{W} - \mathbf{U}_t\|_F^2, \tag{15}$$

where $\mathbf{U}_t = \mathbf{W}_t - \frac{1}{\lambda_t}\nabla f(\mathbf{W}_t)$ and actually the solution of $\mathbf{W}$ is the Euclidean projection of $\mathbf{U}_t$ on $\mathcal{Z}$. $\nabla f(\mathbf{W}_t)$ is the gradient of $f(\mathbf{W}_t)$, and in our paper $\nabla f(\mathbf{W}_t)$ is defined as:

$$\nabla f(\mathbf{W}_t) = \mathbf{X}\mathbf{X}^T\mathbf{W}_t - \mathbf{X}\mathbf{Y} + \alpha\mathbf{X}\mathcal{L}\mathbf{X}^T\mathbf{W}_t. \tag{16}$$

Eq. (14) can be decomposed into n subproblems as:

$$\mathbf{w}_{t+1}^j = \arg\min_{\mathbf{w}^j\in\mathcal{Z}^j} \|\mathbf{w}^j - \mathbf{u}_t^j\|_2^2, \tag{17}$$

where $\mathbf{u}_t^j$, $\mathbf{w}^j$ and $\mathbf{w}_t^j$ are the j-th rows of $\mathbf{U}_t$, $\mathbf{W}$ and $\mathbf{W}_t$, respectively. $\mathcal{Z}^j$ is defined on $\|\mathbf{w}^j\|_1$. Given β, the Euclidean projection has a closed form solution as follows,

$$\mathbf{w}_{t+1}^j = \begin{cases} (1 - \frac{\beta}{\lambda_t\|\mathbf{u}_t^j\|})\mathbf{u}_t^j & \text{if} \quad \|\mathbf{u}_t^j\| \geq \frac{\beta}{\lambda_t} \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

The above method has the convergence ratio of $\frac{1}{\epsilon}$. As discussed in [24], our constrained smooth convex optimization problem can be further accelerated to achieve the optimal convergence $\frac{1}{\sqrt{\epsilon}}$. In particular, this accelerated algorithm is based on two sequences $\mathbf{W}_t$ and $\mathbf{V}_t$ in which $\mathbf{W}_t$ is the sequence of approximate solutions, and $\mathbf{V}_t$ is the sequence of search points, which is an affine combination of $\mathbf{W}_t$ and $\mathbf{W}_{t-1}$ as:

$$\mathbf{V}_t = \mathbf{W}_t + \gamma_t(\mathbf{W}_t - \mathbf{W}_{t-1}), \tag{19}$$

where γ_t is the combination coefficient. The approximate solution $\mathbf{W}_{t+1}$ is computed as a "gradient" step of $\mathbf{V}_t$ through $G_{\lambda_t,\mathbf{V}_t}$. Then the detailed algorithm about *SANT* with this accelerated optimization solution is shown in Algorithm 1.

In the algorithm, we use Nesterov's method [29] to solve the optimization problem in Eq. (6) from line 5 to 22. It is the line search algorithm for λ_t according to the Armijo-Goldstein rule from line 8 to 15. In line 20, η_t is set according to [24]. Based on the algorithm, we can have the solution to the convex optimization problem, and obtain the sentiment class label by Eq. (7).

6.2 Convergence and Complexity Analysis

In this subsection, we discuss the convergence rate and time complexity of the proposed Algorithm 1.

The convergence rate of Algorithm 1 is elaborated in the following theorem.

THEOREM 2. *[17] Assume that $\{\mathbf{W}_t\}$ is the sequence obtained by Algorithm 1, then for any t we have,*

$$f(\mathbf{W}_{t+1}) - f(\mathbf{W}^\star) \leq \frac{2\hat{L}_f\|\mathbf{W}^\star - \mathbf{W}_1\|_F^2}{(t+1)^2}, \tag{20}$$

where $\hat{L}_f = \max(2L_f, L_0)$, L_0 is an initial guess of the lipschitz continuous gradient L_f of $f(\mathbf{W})$ and $\mathbf{W}^\star$ is the solution of $f(\mathbf{W})$.

Proof. The detailed proof of the above theorem can be found in [17]. This theorem shows that the convergence rate of Algorithm 1 is $O(\frac{1}{\sqrt{\epsilon}})$ where ϵ is the desired accuracy. □

Based on the theorem, the algorithm will converge in $\frac{1}{\sqrt{\epsilon}}$ iterations, now we discuss time complexity of the proposed method *SANT* at each iteration. Given a collection of n messages with the feature space m, the objective function Eq. (6) consists of three components. First, for the least squares loss function, it costs $O(mn)$ floating point operations for calculating the function value and gradient of the objective function. Second, for the ℓ_1-norm regularization part, the time complexity is $O(2n)$ based on the Euclidean projection algorithm [24]. Third, the Laplacian regularization part, the time complexity is also $O(mn)$. Therefore, we can solve the objective function in Eq. (6) with a time complexity of $O(\frac{1}{\sqrt{\epsilon}}(mn + 2n + mn)) = O(\frac{1}{\sqrt{\epsilon}}(mn))$.

Algorithm 1: *SANT*: Sentiment Analysis for Noisy Texts with Social Relations

Input: $\{\mathbf{X}, \mathbf{Y}, \mathbf{U}, \mathbf{F}, \mathbf{W}_0, \lambda_1, \alpha, \beta\}$
Output: $\mathbf{W}$

1: Initialize $\eta_0 = 0$, $\eta_1 = 1$, $\mathbf{W}_1 = \mathbf{W}_0$, $t = 1$
2: $\mathbf{A}sc = \mathbf{U}^T \times \mathbf{U}$, $\mathbf{A}ec = \mathbf{U}^T \times \mathbf{F} \times \mathbf{U}$
3: $\mathbf{A} = \mathbf{A}_{sc} + \mathbf{A}_{ec}$
4: Construct Laplacian matrix $\mathcal{L}$ from $\mathbf{A}$
5: **while** Not convergent **do**
6:   Set $\mathbf{V}_t = \mathbf{W}_t + \frac{\eta_{t-1}-1}{\eta_t}(\mathbf{W}_t - \mathbf{W}_{t-1})$
7:   Set $\nabla f(\mathbf{W}_t) = \mathbf{X}\mathbf{X}^T\mathbf{W}_t - \mathbf{X}\mathbf{Y} + \alpha\mathbf{X}\mathcal{L}\mathbf{X}^T\mathbf{W}_t$
8:   **loop**
9:     Set $\mathbf{U}_t = \mathbf{V}_t - \frac{1}{\lambda_t}\nabla f(\mathbf{W}_t)$
10:     Compute $\mathbf{W}_{t+1}$ according to Eq. (18)
11:     **if** $f(\mathbf{W}_{t+1}) \leq G_{\lambda_t,\mathbf{V}_t}(\mathbf{W}_{t+1})$ **then**
12:       $\lambda_{t+1} = \lambda_t$, break
13:     **end if**
14:     $\lambda_t = 2 \times \lambda_t$
15:   **end loop**
16:   $\mathbf{W} = \mathbf{W_{t+1}}$
17:   **if** stopping criteria satisfied **then**
18:     break
19:   **end if**
20:   Set $\eta_{t+1} = \frac{1+\sqrt{1+4\eta_t}}{2}$
21:   Set $t = t + 1$
22: **end while**
23: $\mathbf{W} = \mathbf{W_{t+1}}$

7. EXPERIMENTS

In this section, we present empirical evaluation results to assess the effectiveness of our proposed framework, and answer the question: "Can social relations improve sentiment classification of microblogging messages?". In particular, we evaluate the proposed method on the two datasets introduced in Section 4. Impacts brought by size of training set, different relations and other factors that appear to affect the experiment are further discussed.

7.1 Performance Evaluation

Below we first present the finding of comparing *SANT* with the classical text-based sentiment classification methods, and then with the models that incorporate social relations in sentiment classification.

7.1.1 Comparison with Text-based Methods

In the first set of experiments, we use classification accuracy as the performance metric, and compare the proposed framework *SANT* with following text-based methods:

Table 2: Sentiment Classification Accuracy on STS Dataset

	$D_{10\%}$ (gain)	$D_{25\%}$ (gain)	$D_{50\%}$ (gain)	$D_{100\%}$ (gain)
LS	0.670 (N.A.)	0.704 (N.A.)	0.720 (N.A.)	0.713 (N.A.)
Lasso	0.699 (+4.22%)	0.722 (+2.56%)	0.746 (+3.50%)	0.759 (+6.38%)
MinCuts	0.677 (+0.93%)	0.705 (+0.27%)	0.727 (+0.89%)	0.757 (+6.10%)
LexRatio	0.699 (+4.25%)	0.746 (+5.97%)	0.753 (+4.55%)	0.763 (+6.94%)
SANT	**0.764 (+13.90%)**	**0.778 (+10.56%)**	**0.793 (+10.02%)**	**0.796 (+11.52%)**

Table 3: Sentiment Classification Accuracy on OMD Dataset

	$D_{10\%}$ (gain)	$D_{25\%}$ (gain)	$D_{50\%}$ (gain)	$D_{100\%}$ (gain)
LS	0.615 (N.A.)	0.634 (N.A.)	0.654 (N.A.)	0.660 (N.A.)
Lasso	0.626 (+1.81%)	0.663 (+4.62%)	0.698 (+6.76%)	0.709 (+7.49%)
MinCuts	0.659 (+7.16%)	0.664 (+4.80%)	0.674 (+3.06%)	0.697 (+5.64%)
LexRatio	0.613 (-0.18%)	0.633 (-0.22%)	0.655 (+0.18%)	0.659 (-0.06%)
SANT	**0.685 (+11.51%)**	**0.717 (+13.04%)**	**0.748 (+14.47%)**	**0.763 (+15.72%)**

- *LS*: Least squares [7] is a widely used supervised classification method for i.i.d. data.
- *Lasso*: Lasso [7] on tweet content only. This is one of the most popular sparse learning methods.
- *MinCuts*: Pang and Lee [31] utilized contextual information via the minimum-cut framework to improve polarity-classification accuracy. In the experiment, we use MinCuts package provided by the freely available software LingPipe[6].
- *LexRatio*: The methods [30, 40] count the ratio of sentiment words, from OpinionFinder subjectivity lexicon[7], in a tweet to determine its sentiment orientation. Due to its unsupervised setting, *LS* is used for the tweets do not contain any sentiment words.

There are three important parameters in our experiments, including α, β in Eq. (6) and θ in Section 5.2. All three parameters are positive. α is the parameter to control the contribution of sentiment relation information, β is the sparse regularization parameter, and θ is the parameter to combine the two sentiment relations. As a common practice, α and β are tuned via cross-validation. In the experiments, we set $\alpha = 0.05$ and $\beta = 0.1$ for general experiment purposes. We set $\theta = 1$ which means the two sentiment relation matrices are simply combined with equal weight.

Experimental results of the methods on the two datasets, STS and OMD, are respectively reported in Table 2 and 3. In the experiment, we used five-fold cross validation. To test the sensitivity of *SANT* to different sizes of training data, in the tables, $D_{percentage}$ denotes the amount of data used for training as a percentage of whole training dataset. For example, in each round of the experiment, 80% of the whole dataset is used for training. $D_{50\%}$ means we chose 50% of 80% thus using 40% of the whole dataset as training data. The test dataset is always 20% of the whole dataset. In the tables, “gain” represents the percentage improvement of the methods as compared to the traditional *LS* method. In the round of the experiment, the result denotes the average score of 10 test runs. By comparing the results of different methods, we draw the following observations:

(1) Compared with the text-based methods, *SANT* achieves consistently better performance on both datasets with different sizes of training data. The highest improvement with respect to *LS* is obtained on the OMD dataset when the training dataset is D_{100}. We apply t-test to compare *SANT* with the best methods *MinCuts* and *LexRatio*. The experiment results demonstrate that, by exploiting social relations, our proposed model is able to achieve significant improvement (with the significance level $\alpha = 0.01$) as compared to the state-of-the-art methods.

(2) The sparse learning based method *Lasso* achieves better performance than least squares based method *LS*. This shows that a sparse solution of the feature space is an effective way to tackle with noisy microblogging data. The introduction of sparse regularization has positive impacts on the proposed sentiment classification method.

(3) It is noted that the proposed *SANT* also achieves significant improvement with respect to the baseline methods when using a small training dataset. It outperforms the *LS* baseline of 13.90% and 11.51% in the two datasets using only 10% as the training dataset, which demonstrates that our proposed method is robust to a training dataset with small number of training samples. In addition, *SANT* achieves better performance with only 10% training data comparing with *LS* with all training data. It indicates that, by integrating social relation information, our proposed model can significantly save labeling cost. We will discuss the sensitivity of our proposed method to various sizes of training datasets in Section 7.2.

In summary, the proposed model consistently achieves better performance than the state-of-the-art methods based on text alone. It suggests that the social relation information positively help improve sentiment classification. In the next subsection, we compare *SANT* with the models that make use of social relations.

7.1.2 Incorporating Social Relations

To further evaluate *SANT*, in the second set of experiments, we use the following methods in this set of experiments; in other words, all methods utilize social relation information in classification.

[6]http://alias-i.com/lingpipe/

[7]http://www.cs.pitt.edu/mpqa/opinionfinder_1.html

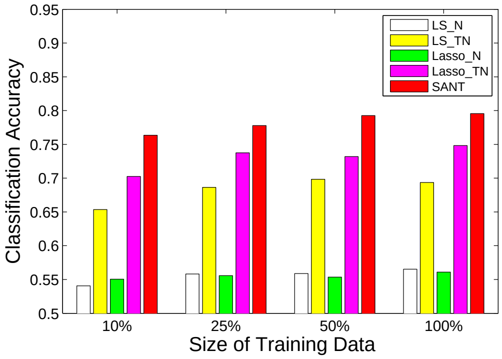


Figure 2: Sentiment Classification on STS Dataset

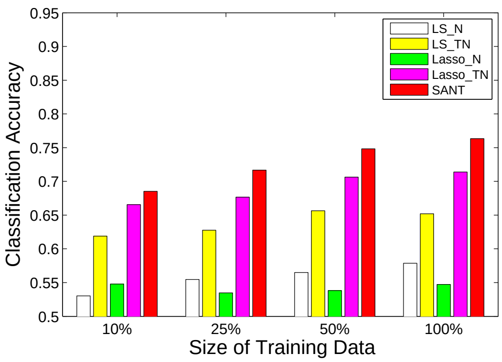


Figure 3: Sentiment Classification on OMD Dataset

- LS_N : Least squares is applied on sentiment relation information only. Following previous work [35], tweet-tweet relation information is used as feature expansion for each tweet. If a tweet t_i is related to another tweet t_j, we add the tweet id t_j as a feature into t_i's feature vector. The following *LS_TN*, *Lasso_N* and *Lasso_TN* methods use the same technique to integrate sentiment relation information between tweets.
- LS_TN: Least squares on tweet content and sentiment relation information together. A sentiment relation matrix is utilized as a feature augmentation of the content feature space. The final feature space is constructed with the equally weighted combination of content matrix and tweet-tweet sentiment relation matrix.
- $Lasso\_N$: Lasso on sentiment relation information only. This is the sparse version of the method LS_N.
- $Lasso\_TN$: Lasso on tweet content and sentiment relation information together. This is the sparse version of the method *LS_TN*. The relation information between tweets is employed as a simple feature expansion, which differs from our proposed method.

Following parameter and experiment settings discussed in the first experiment, we conduct the baseline methods on the two Twitter datasets with different percentages of training data. The classification performance of the methods are

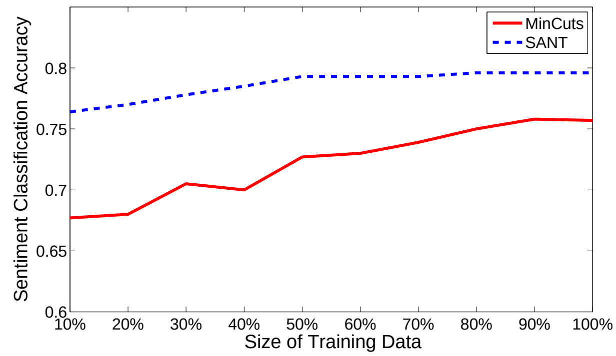


Figure 4: Sensitivity of $SANT$ to Training Data Sizes on STS

plotted in Figures 2 and 3, from which we can draw the following observations:

(1) Among the five methods, $SANT$ achieves the best performance on both datasets with different sizes of training data. It indicates that, comparing with other methods of incorporating social relations, our proposed model successfully utilize the social relations for sentiment analysis.

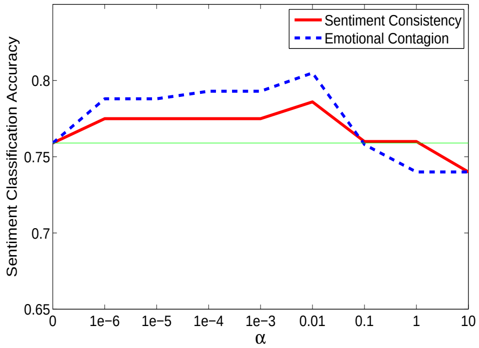
(2) *LS_TN* and *Lasso_TN* achieve better performance than *LS_N* and *Lasso_N*. In addition, the social relation information based methods *LS_N* and *Lasso_N* are slightly better than randomly ($accuracy = 0.5$) assigning a sentiment label to the messages. The results demonstrate that it is inaccurate to rely on sentiment relation information only to determine the sentiment of a microblogging message.

In summary, the existing methods perform differently in sentiment classification. In some cases, the use of social relation information does not help performance improvement. It suggests that the way of using social relations is also important. The superior performance of the proposed method $SANT$ validates its excellent use of social relation information in sentiment analysis.

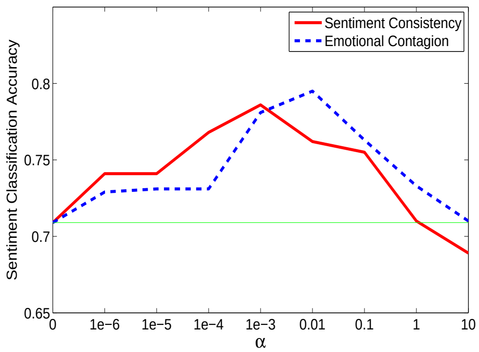
7.2 Sensitivity to Training Data Sizes

One difficulty in sentiment analysis is the lack of manually labeled training data. The publicly available datasets are always insufficient for training purposes in supervised learning methods. We showed that our proposed method is robust to small size of training data in Tables 2 and 3. In order to further investigate the sensitivity of the proposed $SANT$ framework to the size of training data, in Figure 4, we plot the sentiment classification accuracy with training data from 10% to 100% on the STS dataset.

In the figures, we compare the performance of $SANT$ and $MinCuts$. $SANT$ consistently outperforms $MinCuts$ with different sizes of training data. Compared with the results from using the whole training dataset, $SANT$ achieves greater improvement with respect to $MinCuts$ when the size of training data is small. $SANT$ does not show significant changes when the size of the training data changes, which demonstrates that our proposed method is not sensitive to training data sizes. This property has its significance due to the widely existed "lack of manually labeled training data" problem in sentiment classification tasks.



(a) STS Dataset



(b) OMD Dataset

Figure 5: Performance Variation of *SANT*

7.3 Effect of Sentiment Relations

In the experiments, we combined the two sentiment relations with equal weight. To further understand the effect of each relation on the performance of sentiment classification, experiments are conducted with separate sentiment relation on the two datasets. The parameter α controls the contribution of sentiment relation to the model. We varied the value of α from 0 to 10. The results are presented in Figures 5(a) and 5(b). The red curve shows the performance of "Sentiment Consistency" (SC), the blue dotted curve depicts the performance of "Emotional Contagion" (EC), and the green line is baseline without sentiment relation information.

In Figures 5(a) and 5(b), the curves of SC reach the peak at $\alpha = 0.01$ and $\alpha = 0.001$ respectively. For most parameter settings, the classification accuracy with SC is higher than the baseline. This demonstrates that, with only one sentiment relation, $SANT$ can improve the sentiment classification performance as well. When the value of parameter α is not too extreme, SC is not sensitive to the parameter setting. This is an appealing property of the proposed method because it is not necessary to make much effort to tune the parameter. The method can consistently achieve good performance with a large range of parameter settings. The trend of curve EC is similar to SC in the figures.

With different parameter settings, the two relations achieve comparable results. Intuitively, the first sentiment relation SC should have a stronger impact to the model than EC relation. A potential reason is that the constructed matrix of SC is much more sparse than the latter one.

7.4 Multi-Class Sentiment Classification

As discussed in Section 3, following [8, 35], we focused on the polarity sentiment classification task in this paper. It is observed that many tweets do not show clear emotions in real-world applications. Users may post objective expressions about entities and events. For example, in the OMD dataset used in our experiment, besides tweets with positive and negative sentiments, we still have tweets in other categories. In this case, our proposed model can be easily applied to this application, which classifies the tweets as positive, negative and neutral. We next present some preliminary results.

We added the tweets with neutral and other as sentiment labels in the OMD data to construct a three class dataset. The dataset consists 3,269 tweets with the class label as positive, negative and neutral. We compared the performance of our proposed model $SANT$ with the baseline methods LS and *LexRatio*. Among the three methods, our proposed method has the best performance at 58.3% and it achieves 11.66% improvement as compared to LS. Although the focus of this paper is polarity sentiment classification, our method is quite general to be applied to real-world multi-class (> 2) sentiment analysis applications.

8. CONCLUSIONS AND FUTURE WORK

Different from texts in traditional media, microblogging texts are noisy, short, and embedded with social relations, which presents challenges to sentiment analysis. In this paper, we propose a novel sociological approach ($SANT$) to handle networked texts in microblogging. In particular, we extract sentiment relations between tweets based on social theories, and model the relations using graph Laplacian, which is employed as a regularization to a sparse formulation. Thus the proposed method can utilize sentiment relations between messages to facilitate sentiment classification and effectively handle noisy Twitter data. We further develop an optimization algorithm for $SANT$. Experimental results show that the user-centric social relations are helpful for sentiment classification of microblogging messages. Empirical evaluations demonstrate that our framework significantly outperforms the representative sentiment classification methods on two real-world datasets, and $SANT$ achieves consistent performance for different sizes of training data, a useful feature for sentiment classification.

This work suggests some interesting directions for future work. For example, it would be interesting to investigate the contributions of different sentiment relations to sentiment classification. Other information, like spatial-temporal patterns, could be potentially useful to measure the sentiment consistency of people as well [28]. For example, people in Miami might be happier about the temperature than people in Chicago during winter time. We can further explore how sentiments diffuse in the social network and how people's sentiments correlate with internal (their friends) and external (public events [15]) factors. With the analysis, it is possible for us to understand the differences of sentiment between the online world and physical world.

Acknowledgments

We thank the anonymous reviewers for their comments. This work is, in part, supported by ONR (N000141110527) and (N000141010091).

9. REFERENCES

[1] R. Abelson. Whatever became of consistency theory? *Personality and Social Psychology Bulletin*, 1983.

[2] L. Barbosa and J. Feng. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of ACL*, 2010.

[3] J. Bollen, A. Pepe, and H. Mao. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. *CoRR, abs/0911.1583*, 2009.

[4] F. Chung. *Spectral graph theory*. Number 92. Amer Mathematical Society, 1997.

[5] X. Ding and B. Liu. The utility of linguistic rules in opinion mining. In *Proceedings of SIGIR*, 2007.

[6] J. Fowler and N. Christakis. The dynamic spread of happiness in a large social network. *BMJ: British medical journal*, 2008.

[7] J. Friedman, T. Hastie, and R. Tibshirani. The elements of statistical learning, 2008.

[8] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *Technical Report, Stanford*, 2009.

[9] Q. Gu and J. Han. Towards feature selection in network. In *Proceedings of CIKM*, 2011.

[10] E. Hatfield, J. Cacioppo, and R. Rapson. Emotional contagion. *Current Directions in Psychological Science*, 1993.

[11] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of SIGKDD*, 2004.

[12] X. Hu and H. Liu. Text analytics in social media. *Mining Text Data*, pages 385–414, 2012.

[13] X. Hu, N. Sun, C. Zhang, and T.-S. Chua. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *Proceedings of CIKM*, pages 919–928, 2009.

[14] X. Hu, L. Tang, and H. Liu. Enhancing accessibility of microblogging messages using semantic knowledge. In *Proceedings of the 20th CIKM*, 2011.

[15] Y. Hu, A. John, D. Seligmann, and F. Wang. What were the tweets about? topical associations between public events and twitter feeds. *ICWSM*, 2012.

[16] Y. Hu, A. John, F. Wang, and S. Kambhampati. Et-lda: Joint topic modeling for aligning events and their twitter feedback. In *AAAI*, 2012.

[17] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *ICML*, 2009.

[18] S. Kamvar and J. Harris. We feel fine and searching the emotional web. In *Proceedings of WSDM*, pages 117–126, 2011.

[19] D. Keltner and G. Bonanno. A study of laughter and dissociation: Distinct correlates of laughter and smiling during bereavement. *Journal of Personality and Social Psychology*, page 687, 1997.

[20] E. Kim, S. Gilbert, M. Edwards, and E. Graeff. Detecting sadness in 140 characters: Sentiment analysis of mourning michael jackson on twitter. 2009.

[21] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *Proceedings of WWW*, 2010.

[22] K. Lewis, M. Gonzalez, and J. Kaufman. Social selection and peer influence in an online social network. *PNAS*, pages 68–72, 2012.

[23] B. Liu. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, pages 1–167, 2012.

[24] J. Liu, S. Ji, and J. Ye. Multi-task feature learning via efficient l 2, 1-norm minimization. In *Proceedings of UAI*, 2009.

[25] P. Marsden and N. Friedkin. Network studies of social influence. *Sociological Methods & Research*, 1993.

[26] M. McPherson, L. Smith-Lovin, and J. Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, pages 415–444, 2001.

[27] M. Miller, C. Sathi, D. Wiesenthal, J. Leskovec, and C. Potts. Sentiment flow through hyperlink networks. In *Proceedings of ICWSM*, 2011.

[28] A. Mislove, S. Lehmann, Y. Ahn, J. Onnela, and J. Rosenquist. Pulse of the nation: Us mood throughout the day inferred from twitter. 2010.

[29] Y. Nesterov and I. Nesterov. *Introductory lectures on convex optimization: A basic course*. 2004.

[30] B. O Connor, R. Balasubramanyan, B. Routledge, and N. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of ICWSM*, 2010.

[31] B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of ACL*, 2004.

[32] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, pages 1–135, 2008.

[33] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of EMNLP*, 2002.

[34] D. Shamma, L. Kennedy, and E. Churchill. Tweet the debates: understanding community annotation of uncollected sources. In *Proceedings of the first SIGMM workshop on Social media*, pages 3–10. ACM, 2009.

[35] M. Speriosu, N. Sudan, S. Upadhyay, and J. Baldridge. Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the First Workshop on Unsupervised Learning in NLP*, 2011.

[36] C. Tan, L. Lee, J. Tang, L. Jiang, M. Zhou, and P. Li. User-level sentiment analysis incorporating social networks. In *Proceedings of SIGKDD*, 2011.

[37] J. Tang and H. Liu. Feature selection with linked data in social media. In *SDM*, 2012.

[38] J. Tang, X. Wang, H. Gao, X. Hu, and H. Liu. Enriching short texts representation in microblog for clustering. *Frontiers of Computer Science*.

[39] J. Wiebe, T. Wilson, and C. Cardie. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39:165–210, 2005.

[40] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT and EMNLP*, 2005.

[41] X. Zhao, G. Li, M. Wang, J. Yuan, Z.-J. Zha, Z. Li, and T.-S. Chua. Integrating rich information for video recommendation with multi-task rank aggregation. In *Proceedings of ACM MM*, 2011.