Topic Sentiment Analysis in Twitter: A Graph-based Hashtag Sentiment Classification Approach

Xiaolong Wang [†] [*], Furu Wei [‡], Xiaohua Liu [‡], Ming Zhou [‡], Ming Zhang [†]
[†] School of EECS, Peking University, Beijing, China
[‡] Microsoft Research Asia, Beijing, China
[†] xwang95@illinois.edu, mzhang@net.pku.edu.cn
[‡] {fuwei, xiaoliu, mingzhou}@microsoft.com

ABSTRACT

Twitter is one of the biggest platforms where massive instant messages (i.e. tweets) are published every day. Users tend to express their real feelings freely in Twitter, which makes it an ideal source for capturing the opinions towards various interesting topics, such as brands, products or celebrities, etc. Naturally, people may anticipate an approach to receiving the common sentiment tendency towards these topics directly rather than through reading the huge amount of tweets about them. On the other side, Hashtags, starting with a symbol "#" ahead of keywords or phrases, are widely used in tweets as coarse-grained topics. In this paper, instead of presenting the sentiment polarity of each tweet relevant to the topic, we focus our study on hashtag-level sentiment classification. This task aims to automatically generate the overall sentiment polarity for a given hashtag in a certain time period, which markedly differs from the conventional sentence-level and document-level sentiment analysis. Our investigation illustrates that three types of information is useful to address the task, including (1) sentiment polarity of tweets containing the hashtag; (2) hashtags co-occurrence relationship and (3) the literal meaning of hashtags. Consequently, in order to incorporate the first two types of information into a classification framework where hashtags can be classified collectively, we propose a novel graph model and investigate three approximate collective classification algorithms for inference. Going one step further, we show that the performance can be remarkably improved using an enhanced boosting classification setting in which we employ the literal meaning of hashtags as semi-supervised information. Experimental results on a real-life data set consisting of 29,195 tweets and 2,181 hashtags show the effectiveness of the proposed model and algorithms.

Categories and Subject Descriptors

I.2.7 [**ARTIFICIAL INTELLIGENCE**]: Natural Language Processing—*Text analysis*

General Terms

Algorithms, Experimentation

Keywords

Sentiment Analysis, Hashtag, Twitter, Graph-based Classification

1. INTRODUCTION

Twitter [1] is popular for its massive spreading of instant messages (i.e. tweets) and the nature of freedom. Bursts of world news, entertainment gossips about celebrities, and discussions over the recently released products are all collected in Twitter vividly. Beyond merely displaying news and reports, the Twitter itself is also a large platform where different opinions are presented and exchanged. No matter where people come from, what religious belief they hold, rich or poor, civilized or uneducated, they comment, discuss, compliment, argue and complain over topics they are interested in, sharing their own feelings freely. It has been well recognized that these user-generated content with rich sentiment information should be utilized for many applications such as search engines and other information systems.

While tweet level sentiment analysis results indeed provide very useful information, the overall or general sentiment tendency towards topics are more appealing in some scenarios. For example, people are curious about how others feel about Apple's new product "iPhone4" and it will offer great convenience for them if major opinions are collected from massive tweets. Fans of Lady Gaga are fascinated about what is going on with their superstar and the reaction from other people. While reading news about political elections, it is expected to get an overview about the support and opposition for presidential candidates in Twitter at the same time. In all these scenarios, a comprehensive sentiment tendency analysis towards the topic during a time period is in need. In this paper, to address this demand, we utilize the unique characteristic of *hashtag* in Twitter.

In twitter, hashtags are a community-driven convention for adding additional context and metadata to tweets. They are created organically by Twitter users as a way to categorize messages and to highlight topics, which is done by simply prefixing a word or a phrase with a hash symbol, such as "#hashtag". The extensive use of hashtags makes Twitter more expressive and welcomed by people. We measured on a dataset with around 0.6 million randomly selected tweets and found that around 14.6% of them have at least one hashtags in each. When only considering the subjective

[*]This work has been done when the author was visiting Microsoft Research Asia.

[1]http://twitter.com/

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

tweets (tweets with positive/negative sentiment expressions), this number increases to 27.5%. The statistics shows a great potential for sentiment analysis with hashtags in Twitter. Another aspect of analysis illustrates the close connection among the topic, sentiment and hashtags in Twitter. To be precise, hashtags can be categorized into three types. Most hashtags (*topic hashtags*) serve as user-annotated coarse topics, like in tweet "*yesterday I watched ur movie again, and this time I cried. love u so much! #Justin_Bieber*". In other cases, hashtags (*sentiment hashtags*) could be an easy way to highlight the sentiment information. This category of hashtags are composed of sentiment words only, such as "#love", "#sucks", etc. Besides, the third kind of hashtags (*sentiment-topic* hashtags) are those in which the topical word and the sentiment words appear together without separating blanks. For example, "#iloveobama" (I love Obama) directly expresses positive opinion towards President Obama. Hence, hashtags falling in this category are even more informative since they explicitly indicate the sentiment target and its expression at the same time. Based on these observations, we believe the hashtag-level sentiment analysis will bring about much understanding about topics in Twitter.

One intuitive idea about the hashtag-level sentiment classification is to aggregate the sentiment polarity with the classification results for each corresponding tweet containing the hashtag. However, this straightforward method does not perform well in our experiments. One major reason contributing to the poor performance is that even for the state-of-the-art sentiment classification algorithm, the accuracy for tweet-level sentiment classification is usually not as high as expected, making the hashtag-level classification task even more challenging and intractable. We do not focus on the tweet-level sentiment analysis. Instead, we aim to seek other characteristics of hashtags to produce robust and reliable hashtag-level sentiment classification results. Specifically, besides the tweet-level sentiment analysis results, we have identified other two types of information which is powerful for determining the sentiment polarity of hashtags. First, the co-occurrence relationship among hashtags is important. In our Twitter dataset, we observe that for any two co-occurring hashtags, the probability to share the same sentiment polarity is over 0.8055. However, when they are randomly chosen, the value drops to 0.5324. This comparison implies the possibility to employ this pair-wise information to boost the classification performance. Second, the hashtag literal meaning is another useful feature. For the sentiment hashtags (e.g. "#love", "#sucks"), we find they often appear together with topic hashtags (e.g. "#iPad", "#Obama") to form tweets, conveying the sentiment tendency towards the topics clearly; For the sentiment-topic hashtags like "#iloveobama", they are sufficiently self-explainable to indicate the sentiment polarity and the targets explicitly. Accordingly, we are motivated to incorporate the co-occurrence relationsip and the literal information of hashtags into the classification framework, leading us to the the hashtag graph model as presented in this paper.

Particularly, we propose a novel graph model, which incorporates the co-occurrence information of hashtags, to tackle the problem and adopted three popular inference algorithms for the graph-based classification (Loopy belief propagation (LBP) [18], Relaxation labeling (RL) [19] and Iterative classification algorithms (ICA) [15]). Furthermore, we utilized the literal meaning of hashtags as semi-supervision information in our enhanced boosting setting. We compare the results with the SVM-voting baseline, which employs a two-stage support vector machine (SVM) [4, 2] to generate the tweets' sentiment polarity probability distribution and then votes for hashtag classification. Experiment results on a real-life tweet corpus demonstrates the effectiveness of the proposed model and algorithms.

Paper Organization The rest of this paper is organized as follows. We present our graph model and three classification algorithms in section 3 after a brief review of the related work in Section 2. Then, we present the experiments in Section 4 and conclude the paper in Section 5 with some future work.

2. RELATED WORK

The task of sentiment analysis (SA), a.k.a. opinion mining, has been a hot topic in the research community for years. Previous research on sentiment analysis [8, 30] mainly focuses on product or movie reviews, which are experimentally convenient and easy for evaluation. For other document types including webpages and news, efforts are also made to explore the same task [28]. While the bulk of such work has focused on the document level, some others [27, 23, 22] address the sentiment analysis in the phrase and sentence level which regards sentences (phrases) as classification samples. The objective of above works is to obtain the sentiment polarity for given text (snippets, sentences, or documents). As compared to this thoroughly studied problem, the sentiment analysis for topics is rarely investigated. Though some work attempt to incorporate the sentiment factor into topic models like probabilistic latent semantic indexing (PLSI) or latent Dirichlet allocation (LDA) to give the description about opinion generation [12, 10], it is still hard to reach an agreement for the definitions about topics and how to explain the meaning of sentiment classification (positive/negative) for them. The problem lies in that the definition for topic/entity sentiment polarity using only one-bit representation (positive or negative) is not well-posed. In our work, we try to avoid this critical question but instead aim to provide a sentiment-based snapshot for topics in one period through analyzing corresponding tweets and investigating other features. This task is more clarified and clearer because the opinion tendency for a given topic in a certain time interval is usually associated with some burst events and hence the sentiment classification for topics makes sense.

In terms of methodology, natural language processing techniques and machine learning approaches are two major popular methods for sentiment analysis. Many research followed the natural language processing way to tackle the problem. Nasukawa and Yi [14] proposed an approach to identify the semantic relationship between the target and expression with a syntactic parser and sentiment lexicon. In addition, Ding and Liu [7] used linguistic rules to detect the sentiment orientations in product reviews. Although rule-based methods for identifying the sentiment polarity and targets are effective, the major drawbacks are that it cannot be extended without expert knowledge and the coverage of rules is not satisfactory. While on the other hand, Pang et al. [17] investigated three machine learning methods to produce automated classifiers to generate the class labels for movie reviews. They tested on Naïve Bayes, Maximum Entropy and Support Vector Machine, and evaluated the contribution of different features including unigrams, bigrams, adjectives and POS-tags. Their experimental results found that the SVM classifier with unigram presence features outperformed other competitors. In [16], they tried to separate the subjective portions from the objective ones through finding minimum cuts in graphs to achieve better sentiment analysis performance. Generally, the machine learning based approaches usually have higher recall than rule-based methods because of the strong generalization ability of classifiers. However, the performance of classifiers is extremely sensitive to the quality of training data [13, 3, 29], making the text-level sentiment analysis using machine learning techniques rather unreliable. As a result, in this paper, we not only leverage the sen-

timent classification for each tweet but also incorporate the link information among hashtags and the literal meaning of them to solve the hashtag sentiment classification problem, which is expected to be more robust and reliable.

Recently, the opinion mining research has begun to pay more and more attention to social networks such as Twitter because they give rise to the massive user-generated publishing activities. In Twitter, a huge amount of tweets contain sentiment information. Barbosa and Feng [2] first investigat a two-stage SVM (subjectivity and polarity) classifier which seems to be more robust regarding biased and noisy data. In this paper, we adopt this two-stage classification framework to build our tweet-level classifier. In Twitter, some unique characteristics can also be utilized for sentiment classification. Davidov et al. [5] employ hashtags and smileys as sentiment labels for classification to allow diverse sentiment types for short texts. In their another paper [6], they analyze the use of "#sarcasm" hashtags and addressed the problem of sarcastic tweets recognition. Jiang et al. [9] propose to take the target of sentiment into consideration in Twitter sentiment analysis, where the hashtags were also utilized as unigram features. Although the hashtag has become a key feature in many micro-blog services, to our best knowledge, our paper is the first to address the task of hashtag-level sentiment classification.

3. HASHTAG-LEVEL SENTIMENT CLASSIFICATION

We start this section with a formal definition for the task of hashtag-level sentiment classification [2]. Given a set of hashtags $\mathcal{H} = \{h_1, h_2, \ldots, h_m\}$ where each hashtag h_i is associated with a set of tweets $\mathcal{T}_i = \{\tau_{i1}, \tau_{i2}, \ldots, \tau_{in}\}$, we aim to collectively infer the sentiment polarities, $\mathbf{y} = \{y_1, y_2, \ldots, y_m\}$ where $y_i \in \{pos, neg\}$[3], for $\mathcal{H}$. We assume the hashtags in $\mathcal{H}$ are *with* sentiments. The reason lies in that we are particularly interested in the hot hashtags (i.e. topics) which are usually accompanied with sentiment since people tend to express rich sentiment information in their tweets towards these hot topics. The hashtag-level sentiment classification inherently bases upon the tweet-level sentiment analysis results. Let $\mathcal{C}_\mathcal{T}$ be a tweet-level classifier where each tweet τ can be assigned with positive or negative probability $\Pr_{pos}(\tau)$ and $\Pr_{neg}(\tau)$, ensuring that $\Pr_{pos}(\tau)+\Pr_{neg}(\tau) = 1$ to form a binary probability distribution. Here, neutral tweets are ignored since they are not useful for the polarity prediction of hashtags. We develop $\mathcal{C}_\mathcal{T}$ using the state-of-the-art sentiment analysis method, which is presented in details in Section. 4.2.

We can obviously induce the sentiment polarity y_i for the hashtag h_i through aggregating the results from $\mathcal{C}_\mathcal{T}$ by a simple voting strategy. This approach, as stated in Section. 4.3, takes the classification for each hashtag independently. As seen in our experiments, the result is not promising. We have shown that hashtags co-occurring in tweets have much higher probability to share the same sentiment polarity than that if they are randomly selected. This observation clearly motivates us to conduct the hashtag-level sentiment classification collectively, which has been proven to be effective in link-based text classification [24, 20]. In the rest of this section, we will first introduce the hashtag graph model and then present the classification framework and the approximate algorithms for inference.

[2]For the sake of simplicity, we restrict our scenario within the context of Twitter, although applying this framework to other microblogs where hashtags also exist is straightforward.

[3]Hereafter, we use *pos* and *neg* to represent positive and negative label, respectively.

3.1 The Hashtag Graph Model

We define a hashtag graph $\mathbf{HG} = \{\mathcal{H}, \mathcal{E}\}$, in which the edge set $\mathcal{E}$ consists of links between hashtags and each edge e_{ij} represents an undirected link between hashtags h_i and h_j, which co-occur in at least one tweet. Figure. 1 illustrates an example of the hashtag graph, in which hashtags are linked if and only if they co-occur at least once in tweets. Here we take the hastag "#obama" as an example. The surrounding hashtags are generally of three categories: (1) topics which is closely connected to Obama (e.g. "#president" and "#healthcare", etc.); (2) sentiment hashtags which expresses subjective opinions towards Obama, like "#ideal", "#leader" and (3) sentiment-topic hashtags which indicate the target and the sentiment polarity simultaneously, such as "#iloveobama". From this figure, as we can see, the neighbor hashtags more or less lend some sentiment tendency to "#obama". Consequently, It would be unwise if we independently determine the sentiment polarity of each hashtag. Our graph model is aimed at incorporating the co-occurrence relationship and deciding sentiment polarity collectively.

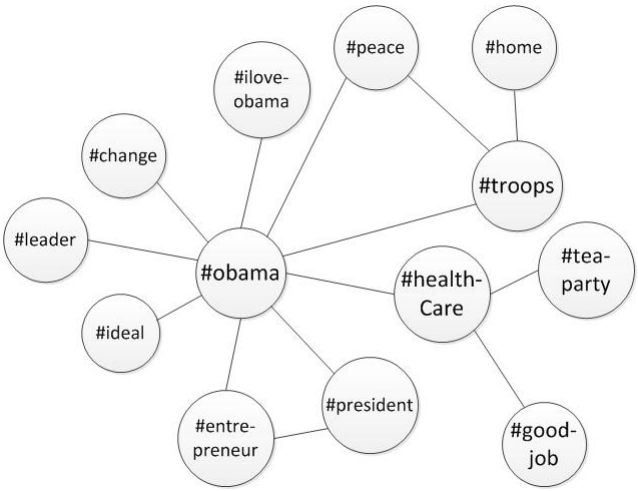


Figure 1: An example of a Hashtag Graph Model

Given the hashtag graph, our ultimate goal is to assign each hashtag h_i with a proper sentiment label $y_i \in \{pos, neg\}$. We make the Markov assumption that the determination of sentiment polarity y_i can only be influenced by either the content of corresponding tweets $\tau \in \mathcal{T}_i$ or sentiment assignments of neighbor hashtag $h_j \; s.t. (h_i, h_j) \in \mathcal{E}$, which results in our HG a *pairwise Markov Network* [21]. This leads us to the following factorized distribution:

$$\log\left(\Pr\left(\mathbf{y}|\mathbf{HG}\right)\right) = \sum_{h_i \in \mathcal{H}} \log\left(\phi_i(y_i|h_i)\right) + \sum_{(h_j,h_k)\in\mathcal{E}} \log\left(\psi_{j,k}(y_j, y_k|h_j, h_k)\right) - \log Z \quad (1)$$

where the first and second sums correspond to the potential functions of a tweet-based factor and a hashtag-hashtag factor. Z is the regularization factor. The potential function of tweet-based factor can be directly obtained through calculation of the polarity probability for each corresponding tweet; while the hashtag-hashtag factor potential function should incorporate the link information to

allow the neighbor hashtags to influence the classification result. The potential functions will be explained together with the inference algorithms in the subsequent section. Given this formula, our objective is to maximize the following function with appropriate sentiment labels:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} \log\left(\Pr\left(\mathbf{y}|\mathbf{HG}\right)\right) \quad (2)$$

3.2 Approximate Collective Classification Algorithms

As for solving the assignment inference problem given the graph model, many efforts are made and some good inference algorithms are proposed as well. In [11], a structured logistic regression based algorithm is investigated as inference method in a link-based text classification framework. Angelova and Weikum [1] combine Relaxation Labeling [19] inference algorithm and Naïve Bayes to form a context-aware approach for hyperlinked text classification. In this paper, we investigate LBP, RL and ICA as inference approaches to solve the hashtag-level sentiment classification problem.

With graph $\mathbf{HG}$ containing cycles and no apparent structure, it becomes infeasible to apply exact inference to the optimization function (Equation. 2). Instead, we present three approximate collective classification algorithms (ACCAs): Loopy Belief Propagation (LBP), Relaxation Labeling (RL) and Iterative Classification Algorithm (ICA) to infer the probable sentiment assignment to each hashtag.

3.2.1 Loopy Belief Propagation (LBP)

As an iterative algorithm, LBP tries to classify each node in a graph through belief message passing. It is originally proposed to work for tree-like networks as a Bayes likelihood-ration updating rule [18]. Although not guaranteed to converge to a fixed point after any number of iterations, LBP shows surprisingly good performance in practice, as discovered by [25]. In fact, the propagation process tries to reach the stationary points of the Bethe approximation free energy for a factor graph [26].

We define the potential functions as follows:

$$\phi_i(y_i|h_i) = \sum_{\tau \in \mathcal{T}_i} \Pr_{y_i}(\tau) \quad (3)$$

$$\psi_{j,k}(y_j, y_k|h_j, h_k) = \frac{\#(h_j, h_k)}{\#(h_j) + \#(h_k)} \cdot \mathrm{I}_{y_j = y_k} \quad (4)$$

where $\mathrm{I}_{y_j=y_k}$ is the identity function with value 1 when $y_j = y_k$ and 0 otherwise. $\#(h_j, h_k)$ denotes the number of co-occurrence for hashtag h_j and h_k and $\#(h_j)$ is the number of occurrence that h_j appears in tweets. Through this setting, the co-occurrence relationship and the label information of the hashtags are taken into account in the hashtag-hashtag factor potential function.

To obtain the result, we compute in an iterative fashion. The Algorithm. 1 is described in pseudo code. In the algorithm, α is a provisional computed normalized factor which keeps that $m_{i\to j}(pos) + m_{i\to j}(neg) = 1$. The sum of polarity probability of corresponding tweets was used as tweet-based factor potential function. The tweet-level classifier is used as a weak classifier to generate the initial classification probability for hashtags. Propagation procedure can be viewed as a boosting process which relabels the hashtags after iteration terminates. During the loops, the positive (or negative) messages conveyed from hashtag h_i to h_j, as denoted by $m_{i\to j}(pos)$ (or $m_{i\to j}(neg)$), are continuously updated until convergence is reached, as shown in the innermost loop of the iterative procedure. These converged message values are then used to compute the final class labels at last.

Algorithm 1: Loopy Belief Propagation

Input: Hashtag Graph $\mathbf{HG}$
Output: Sentiment label for each hashtag h
begin
  foreach $(h_i, h_j) \in \mathcal{E}$ **do**
    foreach $y \in \{pos, neg\}$ **do**
      $m_{i\to j}(y) \leftarrow 1$
      $m_{j\to i}(y) \leftarrow 1$
  repeat
    foreach $h_i \in \mathcal{H}$ **do**
      foreach $h_j \in \mathrm{N}(h_i)$ **do**
        foreach $y_j \in \{pos, neg\}$ **do**
          $m_{i\to j}(y_j) \leftarrow$
          $\alpha \sum_{y_i} \psi_{i,j}(y_i, y_j)\phi_i(y_i) \prod_{h_k \in N(h_i)\setminus h_j} m_{k\to i}(y_i)$
  until *all* $m_{i\to j}(y_j)$ *stop changing*;
  foreach $h_i \in \mathcal{H}$ **do**
    $\hat{y_i} \leftarrow \arg\max_{y \in \{pos,neg\}} \alpha\phi_i(y) \prod_{h_j \in N(h_i)} m_{j\to i}(y)$
  return $\{\hat{y_i}\}$

3.2.2 Relaxation Labeling (RL)

Relaxation Labeling is an alternative inference algorithm for graph-based classification models. Rosenfeld et al. [19] first investigate the RL algorithm in the vision community. Later, it was applied as a general rational classification algorithm. In [1], the algorithm was adopted for text categorization. Unlike LBP, which explicitly defines the potential functions for tweet-based factor and hashtag-hashtag factor, RL assumes that $d_{i,j}$ denotes the "improtance" of node j to its neighbor i and $r(y_i, y_j)$ to be the "compatibility" between labels y_i and y_j, and hence updates the polarity probability of hashtags accordingly at each iteration.

Here, we use $b_i(y_i)$ to denote the possibility that hashtag h_i is labeled with assignment y_i. To measure the compatibility of two sentiment labels, we compute the correlation of any two label (positive or negative) types, as suggested in the nonlinear probabilistic case by [19]. To be precise, at the initialization stage of the algorithm, we aggregate the averaged polarity probability of corresponding tweets as the sentiment probability distribution for hashtags, and then assign the label with the max probability to hashtags. Then, the marginal probability of a label and joint probability of any two labels (*pos-pos*, *neg-neg* and *pos-neg*) can be directly estimated from the assignments. The correlation function is defined as:

$$\mathrm{r}(y_i, y_j) = \frac{\mathrm{p}(y_i, y_j) - \mathrm{p}(y_i)\mathrm{p}(y_j)}{(\mathrm{p}(y_i) - \mathrm{p}(y_i)^2)^2(\mathrm{p}(y_j) - \mathrm{p}(y_j)^2)^2} \quad (5)$$

Naturally, it is expected that the hashtags that are more likely to co-occur in tweets to have more mutual influence. We hereby have:

$$d_{i,j} = \frac{\#(h_i, h_j)}{\#(h_i)} \quad (6)$$

The procedure is presented in Algorithm. 2

3.2.3 Iterative Classification Algorithm (ICA)

In [15], the iterative classification algorithm is presented to explore the relational data. Algorithm. 3 begins by classifying each hashtag using its tweet-based factor potential function. Each iteration recomputes the hashtag's polarity distribution conditioned on

Algorithm 2: Relaxation Labeling

Input: Hashtag Graph $\mathbf{HG}$
Output: Sentiment label for each hashtag h
begin
  foreach $h_i \in \mathcal{H}$ **do**
    foreach $y_i \in \{pos, neg\}$ **do**
      $b_i(y_i) \leftarrow \frac{\sum_{\tau \in \mathcal{T}_i} \Pr_{y_i}(\tau)}{\sum_{\tau \in \mathcal{T}_i} \sum_{y} \Pr_{y}(\tau)}$
  repeat
    foreach $h_i \in \mathcal{H}$ **do**
      foreach $y_i \in \{pos, neg\}$ **do**
        $q_i(y_i) \leftarrow \sum_{h_j \in N(h_i)} d_{i,j} \left[\sum_{y_j} r(y_i, y_j) b_j(y_j) \right]$
        $\alpha_i \leftarrow \sum_{y} b_i(y) \left[1 + q_i(y)\right]$
        $b_i(y_i) \leftarrow \frac{1}{\alpha_i} b_i(y_i) \left[1 + q_i(y_i)\right]$
  until *all* $b_i(y_i)$ *stabilize*;
  foreach $h_i \in \mathcal{H}$ **do**
    $\hat{y_i} \leftarrow \arg\max_{y \in \{pos, neg\}} b_i(y)$
  return $\{\hat{y_i}\}$

the current neighborhood polarity probabilities. At the end of each iteration, it relabels the top-k confident hashtags, where k is a value linearly increasing with the number of iterations. And the last iteration will update polarity assignments for all hashtags.

In our experiment, we have:

$$p_i(y_i|\mathbf{HG}, \mathbf{y}) \propto \exp\Big(\sum_{h_j \in N(h_i)} d_{i,j} \cdot p_j(y_i) \cdot \phi_i(y_i|h_i) \Big) \quad (7)$$

We describe the algorithm below in Algorithm. 3

Algorithm 3: Iterative Classification Algorithm

Input: Hashtag Graph $\mathbf{HG}$
Output: Sentiment label for each hashtag h
begin
  foreach $h_i \in \mathcal{H}$ **do**
    $y_i \leftarrow \arg\max_{y \in \{pos, neg\}} \phi(y|h_i)$
  for $t = 1$ to M **do**
    foreach $h_i \in \mathcal{H}$ **do**
      compute $p_i(y_i|\mathbf{HG}, \mathbf{y})$
      Store $p_i \leftarrow \max_{y} p_i(y|\mathbf{HG}, \mathbf{y})$
      Store $y_i \leftarrow \arg\max_{y} p_i(y|\mathbf{HG}, \mathbf{y})$
    $k \leftarrow |\mathcal{H}| \frac{t}{M}$
    Update the hashtag labels with top-k p_i
  return $\{y_i\}$

3.3 Enhanced Boosting Classification

The three approaches mentioned above use the sentiment labels of corresponding tweets obtained from the tweet-level classifier to initialize the sentiment polarity distribution for every hashtag. Through taking the co-occurrence relationship among hashtags into consideration, the graph-based model, however, boosts the classification result.

Based on previous observations, the sentiment hashtag and sentiment-topic hashtags can indicate the opinion tendency from its literal information. Examples like "#iloveobama", "#sucks", "#awesome", "#horrible", are all self-explainable to infer the sentiment information merely from the hashtag itself explicitly. An interesting observation is that these two kinds of hashtags are usually accompanied with topical hashtags. One simple example tweet is: "*Restoring my #Ipod yet again. #Apple software is such crap that I have to do this routinely. #Apple, fix your software for #Ipod. It #sucks!*". The topical hashtags "#Apple", "#Ipod" show up together with hashtag "#sucks". This pattern strongly conveys negative opinion from "#sucks" to "#Apple" and "#Ipod".

We expect to further enhance the classification algorithms through introducing a semi-supervised adaptive iteration manner which also take advantage of the literal meaning of hashtags. In our work, we first construct a strong sentiment lexicon, specified with labels $\{pos, neg\}$. We assume that hashtags containing these words have the same polarity as the sentiment lexicon. In the three graph-based algorithms, provided the semi-supervision information, we do not use the tweet-level classifier to initialize its sentiment distribution of these self-explainable hashtags. Instead, we fix their sentiment polarity probability from the beginning. In other words, these label-fixed hashtags are not involved in polarity distribution updating, but only offer sentiment influence to others. Experimental results show that this semi-supervised manner of enhanced boosting classification significantly improves the performance.

Because the enhanced boosting settings for three inference algorithms are similar, due to the page length constraint, we only present the major difference of the boosted LBP algorithm as compared with the original version.

For hashtag set $\tilde{\mathcal{H}} \subseteq \mathcal{H}$, each hashtag h_i in $\tilde{\mathcal{H}}$ contains strong sentiment lexicon which is sufficient to indicate the sentiment polarity $y^*_{h_i}$. Unlike the original initialization stage, for $h_i \in \mathcal{H}$:

$$\phi_i(y_i|h_i) = \sum_{\tau \in \mathcal{T}_i} \mathrm{I}_{y^*_{h_i} = y_i} \quad (8)$$

$$\psi_{i,j}(y_i, y_j|h_i, h_j) = \frac{\#(h_i, h_j)}{\#(h_i) + \#(h_j)} \cdot \mathrm{I}_{y_i = y_j \wedge y^*_{h_i} = y_i} \quad (9)$$

Thus the formulas ensure the sentiment assignment for hashtags in $\tilde{\mathcal{H}}$ to be the same as its contained sentiment words. The main difference with the unboost version lies in the iteration procedure, we show the different part in Algorithm. 4:

Algorithm 4: Enhanced Boosting Loopy Belief Propagation

Input: Hashtag Graph $\mathbf{HG}$
Output: Sentiment label for each hashtag h
begin
  ...
  repeat
    foreach $h_i \in \mathcal{H}$ **do**
      foreach $h_j \in \mathrm{N}(h_i)$ **do**
        if $h_j \in \tilde{\mathcal{H}}$ **then**
          continue;
        else
          foreach $y_j \in \{pos, neg\}$ **do**
            $m_{i \to j}(y_j) \leftarrow$
            $\alpha \sum_{y_i} \psi_{i,j}(y_i, y_j) \phi_i(y_i) \prod_{h_k \in N(h_i) \setminus h_j} m_{k \to i}(y_i)$
  until *all* $m_{i \to j}(y_j)$ *stop changing*;
  ...

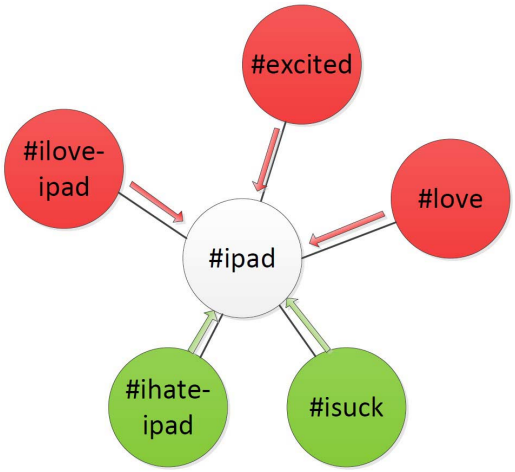


Figure 2: An example of the enhanced boosting classification setting in which strong sentiment hashtags only provide polarity influence to neighbors. Hashtags in red are positive label-fixed nodes and green are negative.

To illustrate this better, we present an example in Figure. 2, where the hashtag "#ipad" has several strong sentiment neighbors such as "#love" and "#isuck". In our enhanced boosting setting, these colored neighbors will not get involved in dynamic updating themselves but only send polarity influence to surrounding neighbors. The propagation from "#ipad" to these colored neighbors will be neglected and blocked.

4. EXPERIMENTAL STUDY

4.1 Data Collection and Evaluation

The evaluation of the hashtag-level sentiment classification is challenging because it is difficult to collect the "golden standard" data set. Although human annotation is possible, we maintain that the workload is rather demanding for large scale evaluation data. What makes it more unreliable is that the satisfactory inter-annotator agreement cannot be achieved, with two contributing factors being that hashtags are often used in tweets with different sentiments, and the sentiment polarity of tweets cannot always be determined with confidence. Instead, in our experiments, to evaluate the performance of the hashtag sentiment classification and to collect the training data for enhanced boosting classification, we use a self-annotation manner to label the dataset.

The data collection process is described as follows. We first ran a coarse-grained selection to find hashtags that we are interested in. We picked 10 topics including "Obama", "Bush", "Lady Gaga", "Justin Bieber", "Islam", "Lakers", "Youtube", "iPad", "Android" and "Microsoft". Then we searched from the tweets pool for hashtags containing the topic words as our seeds. This seed set was hence expanded into our hashtag set $\mathcal{H}$ by retrieving all hashtags that has co-occurred with at least one of the seed hashtags. Finally, for the selected hashtags in $\mathcal{H}$, we labeled hashtags containing sentiment words[4] with appropriate sentiment polarity labels (*pos*, *neg*). This subset of $\mathcal{H}$, denoted by $\tilde{\mathcal{H}}$, is used as our label-fixed set for enhanced boosting classification and test set for evaluation to measure the accuracy, precision, recall and F1 metrics. In addition, we conduct a case study to illustrate some interesting results in Section. 4.6.

In our experiments, our tweets pool has about 0.6 million tweets which were collected in one week period from Twitter. After the seeds selection and data enrichment process, we obtain $\mathcal{H}$ consisting of 2,181 hashtags which occur in 29,195 tweets. The size of edge set $\mathcal{E}$ is 27,430. Selecting hashtags containing strong sentiment words results in a subset $\tilde{\mathcal{H}}$ containing 947 examples, which has 595 positive samples and 352 negative samples. The remaining hashtags in $\mathcal{H}$ do not have a automatic annotated groundtruth, but the classification of them can be evaluated through the case study. This dataset is used for measuring the performance of hashtag sentiment classification algorithms. For enhanced boosting classification approaches, this dataset will be spilled into the training set and test set to evaluate the classification result with cross validation.

4.2 Tweet Level Sentiment Classifier

In this paper, we build the hashtag-level sentiment classification on top of the tweet-level sentiment analysis results. Basically, we adopted the state-of-the-art tweet-level sentiment classification approach [2], which uses a two-stage SVM classifier to determine the sentiment polarity of a tweet. The first (i.e. subjectivity) classifier determines whether a tweet is neutral or subjective while the second one (i.e. polarity classifier) assigns a subjective tweet with positive or negative polarity. The SVM^{light} package [5] is used in our experiments. The two SVM classifiers take the same features as input, which are divided into two categories:

- Content features: including unigram words, punctuation and emoticons. We treat the presence of a token (unigram word, punctuation, or emoticon) as a binary feature which is 1 if the corresponding token occurs in tweet and 0 otherwise.
- Sentiment lexicon features: we employ the lexicon from the General Inquirer [6] and count the number of positive or negative words in tweets as features. There are two dimensions in the feature vector which denote the number of positive and negative words in the tweet.

Classifier	Accuracy	Precision	Recall	F1
subjectivity(1)	83.13%	59.45%	36.59%	45.27%
polarity(2)	88.96%	90.49%	94.82%	92.60%
(1)+(2)	84.13%	/	/	/

Table 1: Performance of the tweet-level classifier

We use the subjectivity classifier to filter out the neutral tweets. The output of the polarity classifier for a subjective tweet is a real value score s which is positive when predicting the tweet t as positive and negative when predicting as negative. Since we need to convert this value into the polarity probability, we use an empirical threshold $\xi = 2$ and the following formula is adopted, which is similar to the manner introduced in [16] :

$$\Pr_{pos}(t) = \begin{cases} 1 & s >= \xi \\ 0.5 + s/(2\xi) & s \in (-\xi, \xi) \\ 0 & s <= -\xi \end{cases} \quad (10)$$

$$\Pr_{neg}(t) = 1 - \Pr_{pos}(t) \quad (11)$$

[4]In our experiments, we selected 50 strong positive and 50 strong negative words as our sentiment lexicon

[5]http://svmlight.joachims.org/

[6]http://www.wjh.harvard.edu/ inquirer/

Setup	Accuracy(%)	Pos-Precision(%)	Pos-Recall(%)	Pos-F1(%)	Neg-Precision(%)	Neg-Recall(%)	Neg-F1(%)
SVM-voting	55.96	64.03	**68.23**	66.06	39.61	35.22	37.29
LBP	56.28	73.26	47.89	57.92	44.44	70.45	54.50
RL	58.07	71.90	54.62	62.08	45.45	63.92	53.12
ICA	59.23	71.81	57.81	64.05	46.36	61.64	52.92
LBP-Boost	**77.72**	97.30	66.69	**79.14**	62.91	95.88	75.97
RL-Boost	72.97	**98.33**	57.80	72.81	57.99	**98.33**	72.95
ICA-Boost	77.40	95.57	67.05	78.88	**63.02**	96.04	**76.11**

Table 2: Performance of the hashtag-level classifiers.

We manually annotated around 15,000 tweets which are randomly selected from the 0.6 million tweets. The tweets are labeled with positive, negative or neutral. Table. 1 shows results of the two-stage SVM tweet-level classifier with 5-fold cross validation. For the two binary classifiers, i.e. subjectivity SVM and polarity SVM, we report the accuracy, precision, recall and F1 values for subjective and positive classes respectively. We also give the overall accuracy for the tweet-level sentiment classifier.

It should be noted that while the accuracy gives an overall evaluation of the classification performance, the precision, recall and F1 values are equally important. These metrics reveal much more information about the classification property, especially when the data is imbalanced. In real-life tweets data, the subjective class is only a minority part: merely 23.8% tweets are subjective in our 15,000 tweets dataset. It is worthy observing that the subjectivity classifier, though have a high accuracy (83.13%), shows extremely low precision (59.45%) and recall (36.59%) for the subjective class, which points out the low ability for discriminating subjective tweets. However, the error analysis and the improving for the tweet-level sentiment classifier is out of the scope of this paper.

4.3 The SVM-Voting Baseline

Intuitively, we can aggregate the hashtag-level sentiment polarity from the results of the tweets containing the hashtag through simple voting methods. We build the SVM-voting baseline approach on the tweet-level classifier. To estimate the positive/negative probability for one hashtag, we use the average polarity distribution of tweets containing hashtag h:

$$\Pr(y_i|h_i) = \frac{\sum_{\tau\in\mathcal{T}_i}\Pr_{y_i}(\tau)}{\sum_{\tau\in\mathcal{T}_i}\Pr_{pos}(\tau)+\sum_{\tau\in\mathcal{T}_i}\Pr_{neg}(\tau)} \quad (12)$$

$$y = \arg\max_{y_i\in\{pos,neg\}} \Pr(y_i|h_i) \quad (13)$$

The results of the SVM-Voting method is presented in Table. 2, which is not promising. The reason lies in that the tweet-level sentiment analysis results are not reliable. We maintain that the low discriminating ability for subjective class is the major cause of this low accuracy (55.96%). This observation motivates us to exploit other available information for this task, as detailed in Section 4.4.

4.4 Evaluation of Graph-based Hashtag-level Sentiment Classification

We compare the three approximate collective classification algorithms with the SVM-voting baseline in this section. The aim is to examine the effectiveness of employing the hashtag co-occurrence information and enhanced boosting techniques in the context of hashtag-level sentiment classification.

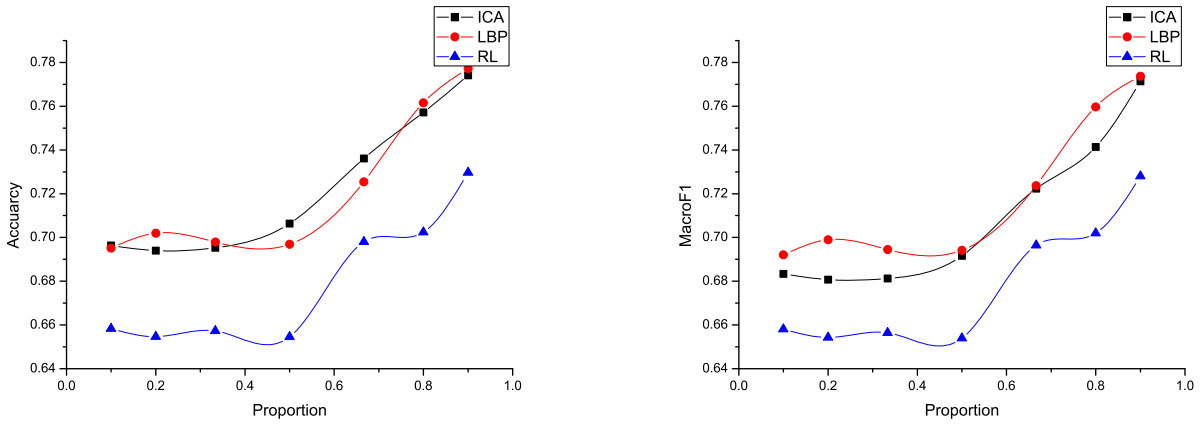
Table. 2 shows the comparison results. The iteration number for ICA is set to $M = 10$ in our experiments. As shown, linked-based algorithms (i.e. LBP, RL and ICA) achieve encouraging results compared to the baseline approach, which clearly demonstrates the effectiveness of the hashtag co-occurrence information in the proposed graph-based model for hashtag-level sentiment classification. The improvement mainly comes from the identifying of negative class, the F1 score of which has a remarkable increase from 37.29% to 54.50% (for LBP). Since the number of negative hashtags (352 out of 947) is much less than that of the positive ones, this benefit from link-based algorithms is not very significant. Another major reason for this result, as we found in the experiments, is that the hashtag Graph $\mathbf{HG}$ suffers the problem of sparseness. The graph density $\mathcal{D}$ [7] is as low as 0.011, which indicates $\mathbf{HG}$ is very sparse. Therefore, many hashtags receive little sentiment propagation from their few neighbors and the final classification result is biased by the SVM-voting outcome.

We further add the enhanced boosting versions of the graph-based algorithms into our performance comparison. In this boosting setting, we run each experiment with 10-fold cross validation, separating the hashtags containing strong sentiment words $\tilde{\mathcal{H}}$ (947 samples) into a training set and a test set (the proportion of the training set to $\tilde{\mathcal{H}}$ is 0.9). The polarity distribution of the labeled set is fixed during iterations. These label-fixed hashtags only play the role of sentiment propagation. Their impact to neighbors is strengthened and results in a notable performance gain, which can be shown from the comparison. In Table. 2, we observe that the most significant improvement is from the boosting loopy belief propagation, with the high classification accuracy up to 77.72%. The other two boosting ACCAs also increase the performance considerably, as compared with the corresponding un-boosting versions. As for the precision of positive (up to 98.33%) and negative class (up to 63.02%), significant progress is also witnessed. Taking these factors into consideration, we conclude that the literal information of hashtags and the semi-supervised boosting classification approaches are able to greatly enhance the discriminating ability for hashtag-level sentiment classification.

4.5 Effects of the Amount of Supervision

In our enhanced boosting setting, we utilized the literal meaning of hashtags as semi-supervision information. Specifically, we separate the hashtag set $\tilde{\mathcal{H}}$ into a label-fixed training set and an test set for performance evaluation. The training set provides sentiment influence to other hashtags in iterations but blocks the propagation back from them. We vary the size of label-fixed set used in each algorithm by randomly selecting a certain number of labeled hashtags. The size of training set is measured by the proportion to the

[7] To measure the density of a graph, we use the metric "Graph Density", which is defined as: $\mathcal{D} = \frac{2|\mathcal{E}|}{|\mathcal{H}|(|\mathcal{H}|-1)}$.



(a) (b)

Figure 3: Accuracy and MacroF1 with different training set. The X-axis value denotes the proportion of labeled-fixed training set to all strong sentiment hashtags $\tilde{\mathcal{H}}$ (947 hashtags). The remaining hashtags in $\mathcal{H}$ are unlabeled.

Hashtag	SVM	HG	Neighbors offering correct impacts	Example tweets
#ipad	neg	pos	#love, #free #iloveApple	Games for Cats on #iPad #jaja #ILoveApple http://youtu.be/vaif2uq_0Vc
#youtube	neg	pos	#fun, #video, #twitter	#youtube channel: :D more views, more comments. MORE #FUN! :D #pdkgaming
#obama	neg	pos	#iloveobama, #change, #peace, #ideal, #freedom	#Hope #Change #Peace #Freedom #Dignity #Nonviolence #Beliving #ideal, #peace & #stability #Obama,am i watching mr #LarryKing or it's all about #UsA interests
#islam	pos	neg	#jihad, #terror, #igiveup	*PLS RT*: The Documentary Tehran Doesn't Want u 2 See http://v.gd/OePtiI #islam #terror #jihad #tcot #tlot #a4a #jcot #sgp #ocra #hhrs #gop
#gaga	pos	neg	#hate, #gay	Say I worship #Gaga... Ok... does that mean I am #gay?
#bush	pos	neg	#kill, #iraq, #fail	MitchDaniels was #Bush's budget director. #FAIL Rs goin2Radical...

Table 3: Case study: examples of hashtags classified correctly only in our proposed graph model.

total labeled 947 hashtags $\tilde{\mathcal{H}}$. Meanwhile, we should keep it in mind that there are altogether 2181 hashtags in **HG**, the remaining of which are not included in the performance test for calculation of precision, recall, F1 and accuracy metrics, but will be used for case study in the following section.

Figure. 3 presents the values of macro F1 and accuracy against varying amount of label-fixed hashtags. The reported values are calculated with 10-fold cross validation. It reflects a strong consistence of the changing tendency for the two metrics. We observe that when little labeled data is provided (0.1 to 0.4), the performance for the three algorithms is stable. The values for macro F1 and accuracy has a notable increase when more labeled hashtags are added, strongly indicating that increasing the amount of training set will effectively improve the overall performance for our graph models. Besides, we can see the performance for ICA and LBP are very close with each other. While on the other hand, we argue that the performance of RL is below the other two algorithms all along. This is because that the correlation between two labels is calculated with the initial probability distribution and will not be updated afterwards in RL.

4.6 Case Study

We investigate the result of hashtag-level sentiment classification by looking at some specific examples. We list some interesting hashtags that can be classified correctly only by our proposed graph model in Table. 3. Since we do not intend to highlight the performance of any specific ACCA, we present the result obtained from LBP only. We list the hashtags together with their neighbors that offer impacts to change their polarity assignments into correct labels[8] and corresponding tweets.

In this list, topic hashtags like "#obama" were classified as negative by SVM-voting at first. This is not true since through our analysis, we found that "#iloveobama", "#change", and "#ideal", and other positive hashtags are often show up together with "#obama", and these neighbor hashtags are created by users to highlight their sentiment tendency towards "#obama", as shown in the following tweets in Table. 3. The collective classification for hashtags can be extremely effective especially when the tweets are not straightforward enough for sentiment classification with the two-stage SVM. The example tweets for "#ipad": "Games for Cats on #iPad #jaja #ILoveIPad http://youtu.be/vaif2uq_0Vc", fails to be predicted correctly with the tweet-level classifier since it cannot capture the positive sentiment from "#ILoveApple" at all. Another tweet talking about Lady Gaga "Say I worship #Gaga... Ok... does that mean I am #gay?" implicitly conveys the negative sentiment which is far too difficult for the tweet-level sentiment classification.

There are three reasons for the low performance of the SVM-voting baseline: (1) Tweets are short and it is hard to infer the sentiment polarity only with the unigram and sentiment lexicon features; (2) tweets contain links directing (or redirecting, such as "http://bit.ly/eZJDoJ") to videos or news that reflect the author's underlying sentiment towards the topics cannot be analyzed successfully. (3) Tweets contain both positive and negative sentiment expression towards topics, like "*iTunes is good software but it fails as usually as you use :) #fail #itunes #apple*". These factors make the tweet-level classifier and our baseline rather sensitive to noisy

[8]After reprocessing, hashtags are case-insensitive.

data, leading to an poor hashtag-level sentiment analysis performance.

Incorporating the neighbor hashtag sentiment information gives us a chance to relabel hashtag polarity collectively. This method improves the performance since it tolerates the error introduced by the tweet-level classification and allow the mutual sentiment influence among hashtags.

5. CONCLUSION REMARK AND FUTURE WORK

In this paper, we investigate a novel task, i.e. sentiment classification of hashtags in Twitter. We believe this is important for sentiment analysis of topics since hashtags can be approximately viewed as user-annotated topics. We develop the baseline approach on sentiment analysis results of the tweets containing the hashtag through simple voting strategy. The performance of this intuitive approach is not encouraging as we expected. In order to improve the hashtag-level sentiment classification, we propose a graph model to boost the results from the voting baseline, which effectively incorporates the tweets sentiment information and hashtags co-occurrence relationship. The preliminary results demonstrate that our graph model is able to give competitive performance as compared with the baseline. Going one step further, by extracting the literal sentiment hint from hashtags, we construct the enhanced boosting hashtag classification framework, in which self-explainable hashstags are label-fixed and not updated for polarity, but only offering sentiment influence to neighbor hashtags. Experiment results show significant improvements are achieved in this boosting settings.

There exists some possible extensions to our work. It would be interesting if we can produce a short summary for hashtags based on the sentiment classification. For example, for a new product it is expected to present a list of related features together with typical sentiment expressions, beyond the one-bit snapshot (positive or negative). In addition, we envision to employ the classification of hashtags to enhance the tweet-level sentiment categorization in return.

6. ACKNOWLEDGMENTS

This study is partially supported by the HGJ Grant (No. 2011ZX 01042-001-001) as well as The Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant ("FSSP" Grant No.20100001110203).

7. REFERENCES

[1] R. Angelova and G. Weikum. Graph-based text classification: learn from your neighbors. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 485–492, 2006.

[2] L. Barbosa and J. Feng. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 36–44, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[3] D. Blaheta. Handling noisy training and testing data. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 111–116. Association for Computational Linguistics, 2002.

[4] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[5] D. Davidov, O. Tsur, and A. Rappoport. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 241–249, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[6] D. Davidov, O. Tsur, and A. Rappoport. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, CoNLL '10, pages 107–116, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[7] X. Ding and B. Liu. The utility of linguistic rules in opinion mining. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 811–812, New York, NY, USA, 2007. ACM.

[8] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM.

[9] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 151–160, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[10] C. Lin and Y. He. Joint sentiment/topic model for sentiment analysis. In *Proceeding of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 375–384, New York, NY, USA, 2009. ACM.

[11] Q. Lu and L. Getoor. Link-based classification. In *Machine Learning, Proceedings of the Twentieth International Conference*, ICML '03, pages 496–503. AAAI Press, 2003.

[12] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 171–180, New York, NY, USA, 2007. ACM.

[13] P. Melville, N. Shah, L. Mihalkova, and R. Mooney. Experiments on ensembles with missing and noisy data. *Multiple Classifier Systems*, pages 293–302, 2004.

[14] T. Nasukawa and J. Yi. Sentiment analysis: capturing favorability using natural language processing. In *Proceedings of the 2nd International Conference on Knowledge Capture*, K-CAP '03, pages 70–77, 2003.

[15] J. Neville and D. Jensen. Iterative classification in relational data. In *Proceedings of the AAAI 2000 Workshop Learning Statistical Models from Relational Data*, AAAI '00, pages 42–49. AAAI Press, 2000.

[16] B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, ACL '04, pages 271–278, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[17] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '02, 2002.

[18] J. Pearl. Reverend bayes on inference engines: A distributed

hierarchical approach. In *Proceedings of the National Conference on Artificial Intelligence*, AAAI '82, pages 133–136, 1982.

[19] A. Rosenfeld, R. A. Hummel, and S. W. Zucker. Scene labeling by relaxation operations. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(6):420–433, 1976.

[20] P. Sen and L. Getoor. Link-based classification. In *Technical Report*, 2007.

[21] B. Taskar, P. Abbeel, and D. Koller. Discriminative probabilistic models for relational data. In *Proceedings of the 18th Conference in Uncertainty in Artificial Intelligence*, UAI '02, pages 485–492. Morgan Kaufmann, 2002.

[22] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354. Association for Computational Linguistics, 2005.

[23] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity: an exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, 35(3):399–433, 2009.

[24] Y. Yang, S. Slattery, and R. Ghani. A study of approaches to hypertext categorization. *J. Intell. Inf. Syst.*, 18:219–241, March 2002.

[25] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Generalized belief propagation. In *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems*, NIPS '00, pages 689–695. MIT Press, 2000.

[26] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312, 2005.

[27] H. Yu and V. Hatzivassiloglou. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing-Volume 10*, pages 129–136. Association for Computational Linguistics, 2003.

[28] W. Zhang, C. Yu, and W. Meng. Opinion retrieval from blogs. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, CIKM '07, pages 831–840, New York, NY, USA, 2007. ACM.

[29] X. Zhu, X. Wu, and Y. Yang. Effective classification of noisy data streams with attribute-oriented dynamic classifier selection. *Knowledge and Information Systems*, 9(3):339–363, 2006.

[30] L. Zhuang, F. Jing, and X. Y. Zhu. Movie review mining and summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 43–50, New York, NY, USA, 2006. ACM.