# Adaptive Co-Training SVM for Sentiment Classification on Tweets

Shenghua Liu, Fuxin Li
Institute of Computing Technology
Chinese Academy of Sciences
Beijing, China 100190
liu.shengh@gmail.com
lifuxin@software.ict.ac.cn

Fangtao Li
Google Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043
fangtao06@gmail.com

Xueqi Cheng, Huawei Shen
Institute of Computing Technology
Chinese Academy of Sciences
Beijing, China 100190
{cxq,shenhuawei}@ict.ac.cn

## ABSTRACT

Sentiment classification is an important problem in tweets mining. There lack labeled data and rating mechanism for generating them in Twitter service. And topics in Twitter are more diverse while sentiment classifiers always dedicate themselves to a specific domain or topic. Thus it is a challenge to make sentiment classification adaptive to diverse topics without sufficient labeled data. Therefore we formally propose an adaptive multiclass SVM model which transfers an initial common sentiment classifier to a topic-adaptive one. To tackle the tweet sparsity, non-text features are explored besides the conventional text features, which are intuitively split into two views. An iterative algorithm is proposed for solving this model by alternating among three steps: optimization, unlabeled data selection and adaptive feature expansion steps. The algorithm alternatively minimizes the margins of two independent objectives on different views to learn coefficient matrices, which are collaboratively used for unlabeled tweets selection from the topic that the algorithm is adapting to. And then topic-adaptive sentiment words are expended based on the above selection, in turn to help the first two steps find more confident and unlabeled tweets and boost the final performance. Comparing with the well-known supervised sentiment classifiers and semi-supervised approaches, our algorithm achieves promising increases in accuracy averagely on the 6 topics from public tweet corpus.

## Categories and Subject Descriptors

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

## General Terms

Algorithms, Design, Experimentation.


*CIKM'13,* Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.

http://dx.doi.org/10.1145/2505515.2505569.

## Keywords

co-training; sentiment classification; multiclass SVM; semi-supervised learning; tweet sentiment

## 1. INTRODUCTION

Recent sentiment classification works show many interests in large scale tweets or blogs [10, 29, 34]. Unlike the product reviews usually companied with a scoring mechanism that quantifies the overall sentiment, tweets lack labeled data. It is a labor intensive task to manually label a large number of tweets. Emoticons were ever used as noisy labels for Twitter sentiment classification [9] to tackle this problem, which brought unexpected noise only relying on the emoticons to label tweets sentiment classes, and the neutral class could not be labeled this way. Semi-supervised approaches [23, 24, 39, 40] have been used for sentiment classification on other medium than Microblog. Semi-supervised Support Vector Machines [1, 7], a.k.a. S3VM, is one of the more promising candidates to utilize the unlabeled data combining with a small amount of labeled ones, since SVM minimizes the structural risk. S3VM was first introduced as Transductive Support Vector Machines (TSVM) by Vapnik [38] and Joachims [16], originally designed for directly estimating labels of unlabeled points. And collaborative training (co-training) framework [3] is an alternative wrapper, and achieves a good performance, which is often used in the scenarios whose features are easily split into different views. [30] demonstrated that algorithms that manufacture a feature split outperform algorithms not using a split, meanwhile algorithms leveraging a natural independent split of features performs better.

Various topics are discussed in timelines and trending topics of Twitter. A sentiment classifier is always dedicated to its training domain, since of the domain-dependent labeled data and features. Take a comment of "read the book" as an example, it could be positive in a book review while negative in a movie review. However, it is impossible to give prelabeled tweets for all kinds of topics that will be discussed in the future, even labeling a small amount of tweets for semi-supervised learning. [27] studied on transferring sentiment classifier developed on a topic for one social medium to a different medium, which is also a topic-dependent approach for cross-medium sentiment classification. Cross-domain or cross-topic adaptation in the context of sentiment analysis was studied in [2, 22, 32], which explicitly borrowed a bridge to connect the topic-dependent features to a known or common features. However, such bridges are built assuming that the parallel sentiment words exists for each pair of topics, such as books, DVDs, electronics, kitchen appliances etc. In product reviews, which is not necessarily applicable to various topics in tweets, and mining the bridges relies on sufficient labeled data. Besides, all above studies focus on the review data set. None of them conduct the cross-domain sentiment classification for twitter data set, which is more difficult because of the sparse content, diverse topics, and lack of labeled data. Detecting and tracking topics from tweets is another research topic, and ad-hoc Microblog search task proposed in Text REtrieval Conference (TREC) 2011 [14] and 2012 [15] is a choice for people to query tweets on a given topic.

Therefore we focus on the cross-domain sentiment analysis on twitter, and propose an adaptive multiclass SVM model which transfers an initial common sentiment classifier to a topic-adaptive one, by simultaneously augmenting the labeled set and expanding topic-adaptive features. To tackle the content sparsity of tweets, non-text features including biological clock, emoticons, and punctuations are explored besides the conventional text features. Because text and non-text features are naturally independent split, co-training framework is a good choice to wrap adaptive S3VM model for collaboratively transferring the sentiment classifier. An iterative algorithm is proposed for solving this model by alternating among optimization, unlabeled data selection, and adaptive feature expansion steps. Since the nonconvex of the problem, some heuristic policy is adopted in the iteration for its convergence. Comparing with the well-known supervised sentiment classifiers, including Naïve Bayes, Decision Tree, multiclass SVM, RF (Random Forest); and semi-supervised approaches, such as multiclass S3VM, and co-training multiclass SVM, our algorithm achieves significance improvements in accuracy on the 6 topics from the public tweet corpus. To better evaluate our algorithms, we test it with some different randomly sampling ratios.

The rest of paper is organized as follows. Section 2 investigates the related work. In section 3, we describe the details of adaptive co-training SVM, and the iterative algorithm to solve it. Extracting the text and non-text features, and the candidates of adaptive features are discussed in section 4. Experiments are conducted in section 5. And section 6 concludes the whole paper.

## 2. RELATED WORK

Sentiment analysis is a hot topic in the area of Natural Language Processing and text mining in recent years [25, 33]. One of the most important tasks is sentiment classification, which aims to classify opinion text into different sentiment polarities, such as positive and negative [21, 37]. For example, Turney [37] presented an effective unsupervised learning algorithm, called semantic orientation, for classifying reviews as recommended or not recommended. A web-kernel based measurement was proposed as PMI-IR, which is independent to the corpus collection in hand. Another important task is opinion extraction, which aims to extract the sentiment words and targets from text [13, 20]. There are also other tasks, including rating prediction [21], aspects mining [28, 35] etc. However, these studies focus on the sentiment of one domain, and did not consider the adaptiveness across topics.

Microblogs as social medias have attracted many studies on sentiment analysis [9, 18, 26, 36]. Go et al [9] introduced a distant supervised learning approach for automatically classifying the sentiment of tweets using emoticons as noisy labels for training data. Tumasjan et al [36] showed that Twitter can be considered as a valid indicator of political opinion. Kouloumpis et al [18] leveraged the existing hashtags in tweets to build training data and demonstrated that part-of-speech features might not be useful for sentiment analysis of tweets. Mehta et al [26] used the Twitter data as a corpus for sentiment analysis and tracking the influence of a particular brand activity on the social network. In our work, we focus on the adaptive sentiment classification across different topics from tweets.

Among the existing semi-supervised learning methods, co-training proposed by Blum [3] in 1998, is attracting more and more attentions. In the original work, some experiments were conducted, and the results were rather encouraging. Nigam and Ghani [30] performed extensive experiments comparing the performance of co-training and another popular algorithm that uses unlabeled data: Expectation-Maximization. These experiments show that co-training outperforms EM even on tasks where there is no natural split of features. And Nigam and Ghani [31] experimentally showed that co-training method can improve the supervised classification performance even though the theoretical assumption cannot be satisfied. Kiritchenko [17] adopted the co-training methods to address the email classification problem. They trained the classifiers using SVM and Na´ıve Bayes, and the results showed that SVM was a better choice. To tackle the lackness of labeled tweets, and adaptiveness to different topics, we build a formal multiclass S3VM model with adaptive feature variables that collaboratively transferring a common classifier to a classifier on a brand new topic.

Cross-domain sentiment classification has been widely studied in recent years. Blitzer [2] proposed an approach called structural correspondence learning (SCL) for domain adaptation. It employed the pivot features as the bridge to help cross-domain classification. Pan [32] proposed a spectral feature alignment (SFA) algorithm to bridge the gap between the domains with domain independent words. Li Fangtao [22] proposed the cross-domain sentiment lexicon extraction for sentiment classification. [12] and [8] conducted cross-domain sentiment classification on topic level. They employed probabilistic topic model to bridge different domains in semantic levels. However, all above studies focus on the review data set. None of them conduct the cross-domain sentiment classification for twitter data set. As described in introduction, twitter data set is much different from the review. It contains fewer words, which need to extract more features. It also contains more diversified topics, which need more data to train a precise original classifier. We also notice that [27] conducted the experiments on cross-media sentiment analysis with news, blogs and twitter data sets. They also find twitter data set is very different from other resources. In this paper, we focus on the cross-domain sentiment analysis on twitter.

## 3. ADAPTIVE CO-TRAINING SVM

Since the sentiment classification requires telling among the objective i.e. neutral, positive and negative expressions, it is obviously a multiclass classification problem. The training data is given in $(x_i, y_i)$, where $x_i$ is the feature vector with each element as the value of the corresponding feature, and $y_i$ is the class that the data belongs to. Let $K$ be the number of classes we need to classify to, and $y_i \in \{1, \cdots, K\}$. Each tweet belongs and only belongs to one class.

In this section, we firstly introduce multiclass SVM model based on "one-versus-rest" strategy as preliminaries. And we then formally build semi-supervised multiclass SVM model adapting to unlabeled data on a specific topic, which only binary classifier, SVMs, in semi-supervised setting was formulated in the existing work. Furthermore, the semi-supervised multiclass SVM model is extended to adapt to sentiment words on a specific topic as well. At last, both text and non-text features are extracted and naturally split to apply co-training scheme for sentiment classification on tweets, and an iterative procedure that alternates among three steps, *i.e.* optimization, unlabeled data selection, and adaptive feature expansion, is proposed to solve the model.

## 3.1 Multiclass SVM

SVMs model is originally build for binary classification, thus we investigate the SVM-based classifiers for multiple classes. There are intuitive ways to solve multiclass with SVMs. The most common technique in practice has been to build $K$ "one-versus-rest" classifiers, and to choose the class which classifies the test data with greatest margin. Or we build $K(K-1)/2$ "one-versus-one" classifiers, and choose the class that is selected by the most classifiers. In our work, the "on-versus-rest" strategy is used for multiclass.

Based on "one-versus-rest" strategy which optimizes the structural risk, multiclass SVM model can be formally build as follows.

$$\min_{\mathbf{w},\xi} \quad \frac{1}{2}\sum_{i=1}^{K} w_i^T w_i + \frac{C}{n}\sum_{i=1}^{n}\xi_i \tag{1}$$

$$s.t. \quad w_{y_i}^T x_i - w_y^T x_i \geq 1 - \xi_i, \forall y \in \{1, \cdots, K\} \setminus \{y_i\} \tag{2}$$

$$\xi_i \geq 0, \; i = 1, \cdots, n$$

where $\mathbf{w}$ is a matrix with each column $w_i$ as the coefficient vector corresponding to the features, $\xi_i$ is the non-negative slack variable for input $i$, and $C$ is a constant coefficient. To better capture how $C$ scales with the training set, $C$ is divided by the total number of labeled data. It is seen that model (1) is a structured SVM model which results in a single support vector machine, instead of multiple SVMs for binary classification.

Constraint (2) shows the "one-versus-rest" strategy with greatest margin. And we call the $w_y^T x_i$ the confidence score of data $t_i$ belonging to class $y$. Thus in the prediction, the class label of data $t_i$ is

$$y_i' = \arg\max_y \{w_y^T x_i\} \tag{3}$$

By introducing the maximum function, the constraints in multiclass SVMs model (1) can be written as equation (4).

$$\xi_i = \max_{y \neq y_i}\{0, 1 - w_{y_i}^T x_i + w_y^T x_i\} \tag{4}$$

And the multiclass SVMs model is rewritten as

$$\min_{\mathbf{w},\xi} \frac{1}{2}\sum_{i=1}^{K} w_i^T w_i + \frac{C}{n}\sum_{i=1}^{n}\max_{y \neq y_i}\{0, 1 - w_{y_i}^T x_i + w_y^T x_i\} \tag{5}$$

## 3.2 Adapting to unlabeled data

The sentiment classifiers trained using labeled tweets from one topic are not usually adaptive to another one. Thus we choose labeled data set $L$ from an even mixture of various topics. The initial sentiment classifier is trained using such a labeled set $L$, which is viewed as a common classifier since of the small amount of mixed labeled data and common sentiment lexicon. To adapt to sentiment classification on a topic $e$, the unlabeled tweet set $U$ of topic $e$ is used for transferring without any more costs of manually labeling on topic $e$ in a semi-supervised way.

With adding those unlabeled tweets as an optimization term in the optimization model (5), the semi-supervised multiclass SVM model adapting to unlabeled tweets on topic $e$ is formed as follows.

$$\min_{\mathbf{w},\xi} \frac{1}{2}\sum_{i=1}^{K} w_i^T w_i + \frac{C}{|L|}\sum_{t_i \in L}\max_{y \neq y_i}\{0, 1 - w_{y_i}^T x_i + w_y^T x_i\} + \frac{C'}{|U|}\sum_{t_j \in U}\max_{y \neq y_j}\{0, 1 - w_{y_j'}^T x_j + w_y^T x_j\} \tag{6}$$

where $|L|$ and $|U|$ indicate the number of elements in sets $L$ and $U$ separately. Furthermore, the selected unlabeled tweet $t_j$ is predicted to be class $y_j'$ as equation (3). Thus the following equation satisfied.

$$w_{y_j'}^T x_j = \max_y \{w_y^T x_j\}$$

We define the function submax$\{\cdot\}$ as the second largest value in set $\{\cdot\}$. So the slack for unlabeled data in the third minimization term of model (6) is as follows.

$$\max_{y \neq y_j'}\{0, 1 - (w_{y_j'}^T x_j - w_y^T x_j)\}$$

$$= \max\{0, 1 - \max_y\{w_y^T x_j\} + sub\max_y\{x_y^T x_j\}\}$$

$$= h_l(\max_y\{w_y^T x_j\} - sub\max_y\{w_y^T x_j\})$$

It is the hinge loss $h_l(\cdot) = \max\{0, 1 - \cdot\}$ of the difference between the largest and the second largest confidence score of tweet $t_i$ among all the sentiment classes. In practice, not all the unlabeled tweets are picked for the adaptive training. Only the most confident classifying results are preferred to be added to avoid bringing much noise. We define the normalized confidence score $S_j$ of tweet $t_j$ with predicted class $y_j'$ as

$$S_j = \frac{w_{y_j'}^T}{\sum_y w_y^T x_j} = \frac{\max_y\{w_y^T x_j\}}{\sum_y w_y^T x_j}$$

Therefore given a confidence threshold $\tau$, the unlabeled tweets $t_j$ satisfying with $S_j \geq \tau$ are selected. Thus the optimization model with confidence threshold is as follows.

$$\min_{\mathbf{w},\xi} \frac{1}{2}\sum_{i=1} K w_i^T w_i + \frac{C}{|L|}\sum_{t_i\in L}\max_{y\neq y_i}\{0, 1 - w_{y_i}^T x_i + w_y^T x_i\}$$
$$+\frac{C'}{|\alpha|}\sum_{t_j\in U}\alpha_j h_l(\max_y\{w_y^T x_j\} - sub\max_y\{w_y^T x_j\}) \quad (7)$$
$$s.t. \quad \alpha_j = \max\{0, sign(S_j - \tau)\}\ \forall t_j \in U$$

where $C'$ is a constant coefficient for unlabeled optimization term, $\alpha$ is binary vector, and each value indicates whether the unlabeled tweet is selected for optimization. Function sign$(\cdot)$ returns 1 if the variable is positive, 0 if it is zero, and $-1$ if it is negative. $|\alpha|$ is the number of ones in binary vector $\alpha$.

## 3.3 Adapting to features

In the other hand, tweets on various topics may have quite different sentiment lexicons [32]. Thus we add the topic-adaptive sentiment words as variables in feature vector $x_i$ of tweet $t_i$, while keeping the common sentiment lexicons fixed as common feature values. Let $x_i$ with $i = 1, \ldots, v$ are the fixed common feature values, and $x_i$ with $i = v + 1, \ldots, v+u$ are the topic-adaptive feature variables. Initially, $x_i = 0, i = v + 1, \ldots, v + u$, which guarantees a common classifier is trained. In the adaptive training, the feature values of topic-adaptive sentiment words are evaluated in the selected unlabeled tweets.

The weight of a topic-adaptive sentiment word $\pi$ belonging to a class $y$ is as follows.

$$\varphi_y(\pi) = \sum_{j, y'_j = y} \alpha_j f_j(\pi) \cdot w_{y'_j}^T x_j$$

where $f_j(\pi)$ is the term frequency of word $\pi$ in tweet $t_j$. The equation shows that $\varphi_y(\pi)$ is the weighted summation of the term frequency of word $\pi$ in the tweets $t_j$ with predicted class $y'_j$ being $y$. The sentiment class that word $\pi$ belongs to is

$$\arg\max_y\{\varphi_y(\pi)\}$$

And the feature value $x_\pi = \max_y\{\varphi_y(\pi)\}$. In the same way, we do not hope all the topic-adaptive sentiment words are added. Thus given a threshold $\theta$, the selection vector $\beta$ is defined in the following way.

$$\beta_\pi = \max\{0, sign(w_\pi - \theta)\} \quad (8)$$

And we define the significance $w_\pi$ of sentiment word $\pi$ belonging to a predicted class as follows.

$$w_\pi = \frac{\max_y\{\varphi_y(\pi)\}}{\sum_y \varphi_y(\pi)}$$

The normalized $w_\pi$ indicates the significance of sentiment word $\pi$ in the class $\arg\max_y\{\varphi_y(\pi)\}$ than that in other classes. Thus the values of topic-adaptive features in the objective function of model (7) are calculated as follows.

$$x_\pi = \beta_\pi \cdot \max_y\{\varphi_y(\pi)\},\ \pi = v+1, \ldots, v+u \quad (9)$$

**Algorithm 1** Co-training for sentiment classification.

**Given:**
A set of labeled data $L$;
A set of unlabeled data $U$;

**loop**
Train sentiment classifier $C_1$ on feature set $\chi_1$;
Train sentiment classifier $C_2$ on feature set $\chi_2$;
Select the most confident and unlabeled data from $U$, according to $C_1$ and $C_2$;
Add selected data into $L$.
**end loop**
**return** $L$, $C_1$, and $C_2$.

## 3.4 Adaptive co-training algorithm

In the conventional co-training framework, features are independently split into two views, denoted as $\chi_1$ and $\chi_2$. Two classifiers $C_1$ and $C_2$ are trained based on feature sets $\chi_1$ and $\chi_2$ separately using initial labeled data $L$. As Algorithm 1 shows, the unlabeled and confident data are selected to augment labeled data set $L$, which is used for the next iteration. And the final sentiment classification result is obtained by the classifier trained on the combining features $\chi_1 + \chi_2$ using augmented labeled data set $L$.

Since a tweet is extremely short, it is necessary to extract more features besides the traditional sentiment words. Some intrinsic properties of tweets are explored as non-text features, which include post time, emoticons, and punctuations, which are naturally applicable across different topics. Therefore, it is inspiring to use the non-text features to find more topic-related unlabeled data for adaptive training, with the help of co-training framework.

In order to make adaptive multiclass S3VMs model (7) in a co-training scheme, we define another selection vector $\alpha'$ of unlabeled data for another multiclass S3VMs.

$$\alpha'_j = \max\{0, sign(S'_j - \tau)\} \quad (10)$$

And classification confidence $S'_j$ is calculated with non-text features $\chi'_j \in \chi_2$ as follows.

$$S'_j = \frac{\max_y\{(w'_y)^T x'_j\}}{\sum_y (w'_y)^T x'_j}$$

In practice, to avoid noises, we follow "agreement" strategy [4], and only select the confident and unlabeled data that both classifiers agree most. Thus we replace $\alpha_j$ with $\alpha_j \cdot \alpha'_j$ as the selection coefficient in the third optimization term of model (7) resulting in the objective of text view $\chi_1$ as formula (11), and the objective of non-text view $\chi_2$ has the same form as formula (11) by replacing coefficient matrix $\mathbf{w}$ and feature vector $x$ with $\mathbf{w}'$ and constant vector $x'$ corresponding to the feature values of $\chi_2$.

$$\min_{\mathbf{w},\xi} \frac{1}{2}\sum_{i=1}^{K} w_i^T w_i + \frac{C}{|L|}\sum_{t_i\in L}\max_{y\neq y_i}\{0, 1 - w_{y_i}^T x_i + w_y^T x_i\}$$
$$+\frac{C'}{\alpha^T\alpha'}\sum_{t_j\in U}\alpha_j\alpha'_j \cdot h_l(\max_y\{w_y^T x_j\} - sub\max_y\{x_y^T x_j\}) \quad (11)$$

In the adaptive co-training S3VM model for multiclass, there are coefficient matrices $\mathbf{w}$ and $\mathbf{w}'$ separately for text

**Algorithm 2** Algorithm of adaptive co-training for sentiment classification on topic $e$

**Given:**

Features are separated into two views: text view $\chi_1$ and non-text view $\chi_2$;

$x_i \in \chi_1$, contains fixed features of common sentiment words, with $i = 1, \ldots, v$, and variables of topic-adaptive sentiment words, with $i = v+1, \ldots, v+u$;

$x'_i \in \chi_2$, contains the intrinsic properties indicating sentiment of tweets;

A set of labeled tweets $L$ contains positive, neutral and negative classes on various topics;

A set of unlabeled data $U$ on topic $e$;

Classification confidence threshold $\tau$;

A threshold $\theta$ for adaptive feature expansion.

**loop**

**repeat**

[**Optimization**]

Minimize the objective in model (11) for $\mathbf{w}$ on feature set $\chi_1$;

Minimize the objective in model (11) for $\mathbf{w}'$ on feature set $\chi_2$;

[**Unlabeled data selection**]

Calculate the confidence scores $S_j$ and $S'_j$ according to $\mathbf{w}$ and $\mathbf{w}'$;

Select the $l$ most confident and unlabeled tweets $t_j$ in each sentiment class, such that $\alpha_j \cdot \alpha'_j = 1$;

Move them with predicted classes from $U$ into $L$;

**until** $\forall t_j \in U$ such that $\alpha_j \cdot \alpha'_j = 0$ **or** number of iterations $> M$.

[**Adaptive feature expansion**]

Calculate the significance $\omega$ according to $\mathbf{w}$, $\alpha$, and the latest feature vector $x$;

Select the $c$ most significant and topic-adaptive sentiment words $\pi$ in each sentiment class, such that $\omega_\pi \geq \theta$;

Update the feature vector $x$;

**end loop**

Train multiclass SVM $C^*$ on the features consist of $x$ and $x'$ using augmented $L$.

**return** $L, x$, and $C^*$.

.

and non-text feature sets $\chi_1$ and $\chi_2$, adaptive feature variables $x_i, i = v+1, \ldots, v+u$ in set $\chi_1$, and selection vector $\alpha$ and $\alpha'$ for unlabeled data to be estimated. Thus we can solve this model using an iterative procedure that alternates among the following three steps.

**Optimization step** for solving coefficient matrices $\mathbf{w}$ and $\mathbf{w}'$: by fixing selection vectors $\alpha$ and $\alpha'$, and feature vector $x$, we solve two minimization of multiclass SVM separately, since the feature sets $\chi_1$ and $\chi_2$ are designed conditionally independent.

**Unlabeled data selection step** for solving selection vectors $\alpha$ and $\alpha'$ : by fixing matrices $\mathbf{w}, \mathbf{w}'$ and feature vector $x$, the unlabeled data are selected by comparing the confidence scores $S_j$ and $S'_j$ with threshold $\tau$ according to the constraint in model (7) and equation (10) separately.

**Adaptive feature expansion step** for solving adaptive feature variables $x_i, i = v+1, \ldots, v+u$ : by fixing $\mathbf{w}, \mathbf{w}'$, and selection vectors $\alpha$ and $\alpha'$, topic-adaptive sentiment words are expanded according to equation (8), and their corresponding feature variables $x_i$ are evaluated as equation (9).

In the details of the algorithm, unlabeled tweets and topic-adaptive sentiment words are added gradually to avoid misleading of the overwhelming unlabeled data and adaptive features at early iterations. As shown in Algorithm 2, it is given a common feature set split into independent views: text view $\chi_1$ and non-text view $\chi_2$, an even mixture of labeled tweets from various topics, and a specific topic $e$ with its unlabeled tweet set $U$. The candidates of topic-adaptive sentiment words are extracted from unlabeled set $U$, which is described in section 4. In feature set $\chi_1$, we set the values of all the candidate words to zeros, which only common features take effects initially. By fixing topic-adaptive features, optimization and unlabeled data selection steps alternates until the rest tweets $t_j \in U$ such that $S_j < \tau$ or $S'_j < \tau$. In an iteration, at most $l$ unlabeled tweets selected from each sentiment classes. The iterations between optimization and unlabeled data selection agree well with the co-training framework. Furthermore, adaptive feature expansion is executed for expending at most c topic-adaptive sentiment words, and transferring the sentiment classifier in the aspect of features. Such a step is alternated with the above co-training iterations, until there are no adaptive words can be expanded or updated. Finally, at the end of the algorithm the augmented labeled set $L$, expended adaptive features $x$ and a combined sentiment classifier $C^*$ is output. Classifier $C^*$ is trained on the expanded adaptive features $x_i$ from text view $\chi_1$ and $x'_i$ from non-text view $\chi_2$, using the augmented set $L$.

# 4. FEATURE EXTRACTION

Features are split into two views, i.e. text feature set $\chi_1$ of sentiment words, and non-text feature set $\chi_2$, including emoticons, temporal features, and punctuation.

**Text features:**

Sentiment words consist of common ones and topic-adaptive ones. The common sentiment words are collected from WordNet and public sentiment lexicon. The feature values are evaluated by Point-wise Mutual Information and Information Retrieval (PMI-IR) [37], which uses the web search engine as the kernel to get the convincing mutual information based on a huge amount of database. The PMI-IR value of sentiment word $\bar{w}$ is calculated as follows.

$$PM\text{-}IR(\bar{\omega}) =$$

$$log_2 \left[ \frac{hits(\bar{\omega} \text{ NEAR ``excellent''}) \cdot hits(\text{``poor''})}{hits(\bar{\omega} \text{ NEAR ``poor''}) \cdot hits(\text{``excellent''})} \right],$$

$$\bar{\omega} \in P$$

where $hits(\cdot)$ is the number of records in the query results, *e.g.* $hits$(“excellent”) indicates the number of records that containing “excellent” in the database of web search engine. “NEAR” is one of the commonly supporting keywords of search engines, which means the word on the left side co-occurs with the right side one and no more than a certain number of words away. And $P$ is the common sentiment lexicon. So $x_{\bar{\omega}} = PM\text{-}IR(\bar{\omega}), \bar{\omega} \in P$.

The topic-adaptive lexicon is built based on the part-of-speech (POS) tagging. We select the frequent adjectives, verbs, nouns and adverbs from the tagged tweets, and added them as candidates of topic-adaptive sentiment word set $\Pi$ with initial feature values $x_\pi = 0, \pi \in \Pi$. And the whole vector space of text feature set $\chi_1$ is $v + u$, where $v = |P|$ and $u = |\Pi|$.

**Non-text features:**

*Temporal features.* One of the most notable differences between tweets and traditional documents lies in their real-time; therefore, peopleaŕs sentiments expressed in Twitter evolve dynamically over time. It is proved that the opinions expressed by users correlated well with their biological clock [19]. People tend to act differently in the morning and the noon, the beginning and ending of a week or month, spring and winter, etc. Thus we classify the post time into different hours, dates, day of week and months as temporal features. As for tweets post in the different time zones, we can map the post time into their local periods without difficulties.

*Emoticon feature.* A set of emoticons from Wikipedia are collected as a dictionary, such as :-), :), (∼_∼;),(> _ <), >:[, :-(, :(, etc. They are labeled with positive (+1), neutral (0), or negative (−1) emoticons. The corresponding values of an emoticon in a tweet are summed up as its feature value.

*Punctuation feature.* Punctuation marks such as exclamation mark (!), question mark (?), and their combinations and repeats, express the emotional intensity. Thus the term frequency of each punctuation mark in a tweet is counted for the feature.

# 5. EXPERIMENTS

## 5.1 Test Cases

In the experiment, we use three publicly available Corpuses for evaluation. One is Sanders-Twitter Sentiment Corpus consisting of 5,513 manually labeled tweets. These tweets were collected with respect to one of the four different topics (Apple, Google, Microsoft, and Twitter). After removing the non-English and spam tweets, we have 3,727 tweets left. Another one is the 9,413 tweets mentioned "Taco Bell" during January 24-31, 2011. And the last one is the first 2008 Presidential debate corpus [6] with sentiment judgments on 3,238 tweets. The detailed information of the corpus is shown in Table 1.

**Table 1: Corpus Statistics**

| Topics | Positive | Neutral | Negative | Total |
|---|---|---|---|---|
| Apple | 191 | 581 | 377 | 1149 |
| Google | 218 | 604 | 61 | 883 |
| Microsoft | 93 | 671 | 138 | 902 |
| Twitter | 68 | 647 | 78 | 793 |
| Taco Bell | 902 | 2099 | 596 | 3597 |
| President Debate | 1465 | 1019 | 729 | 3213 |

With some necessary preprocess of the tweets, we use the Stanford POS tagger to tag the tweets of each topic, and select the frequent adjectives, verbs, nouns and adverbs as the candidates for topic-adaptive sentiment words. In order to show how our algorithm performs with a small amount of labeled data, we randomly sample $p$ labeled tweets from each of the three sentiment classes as training data $L$. And the rest $(1 - p)$ tweets from each topic are used for testing.

## 5.2 Baselines and evaluation metrics

We use the well-known supervised and semi-supervised approaches as the baselines. The baselines and our comparing algorithms are listed in the following.

- **NB:** It is Naïve Bayes and a very simple but effective classification model.
- **DT:**It is a Weka [11] implementation of Decision Tree, which is a tree-like model in which internal node represents test on an attribute.
- **MSVM [5]:** It is a multiclass SVM classification which is based on Structural SVM and it is an instance of SVMstruct.
- **RF:** It is a Weka [11] implementation of Random Forest, which is an ensemble learning method for classification of a multitude of decision trees, and we tune the number of trees to be 10 for better performance.
- **MS3VM:** It is multiclass semi-supervised SVM which is our implementation of augmenting unlabeled tweets without expanding adaptive features.
- **CoMS3VM:** It is MS3VM algorithm in a co-training scheme.
- **ACoMS3VM:** It is our adaptive co-training SVM algorithm.

As for the evaluation, we calculate the accuracy, precision, recall, and F-score. Let the number of correctly classified tweets be true positive $tp_y$, and the number of incorrectly classified tweets be false positive $fp_y$ in all the tweets predicted as class $y$. Let the number of tweets correctly classified into other classes than class $y$ be true negative $tn_y$, and the number of tweets incorrectly classified into other classes than class $y$ be false negative $fn_y$, in all the tweets classified into other classes than class $y$. Thus the metrics are calculated as follows.

$$Accuracy = \frac{\sum_y (tp_y + tn_y)}{\sum_y (tp_y + fp_y + tn_y + fn_y)} \tag{12}$$

$$Precision(y) = \frac{tp_y}{tp_y + fp_y}$$

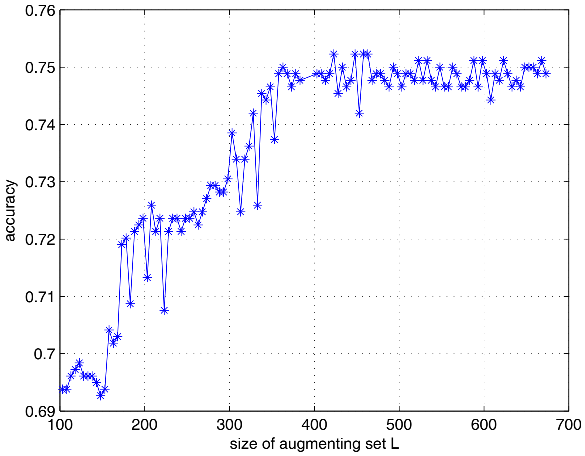
$$Recall(y) = \frac{tp_y}{tp_y + fn_y}$$

$$F\text{-}score(y) = \frac{Precision(y) + Recall(y)}{2}$$

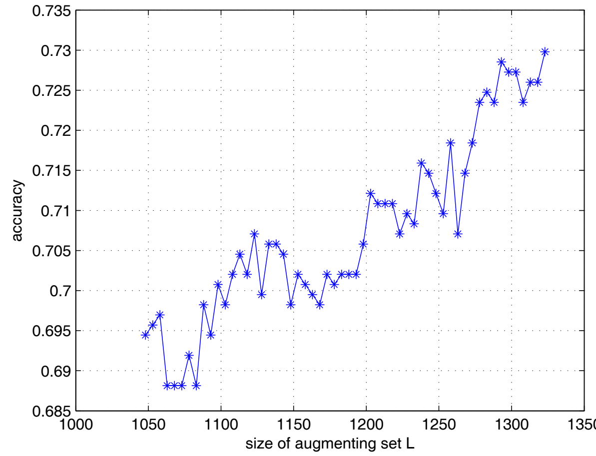
Considering that there are 3 sentiment classes, we average the precisions, recalls and F-scores for all classes as follows.

$$Precision = \frac{\sum_y Precision(y)}{K} \tag{13}$$

$$Recall = \frac{\sum_y Recall(y)}{K} \tag{14}$$




(a) Sample ratio $p = 1\%$




(b) Sample ratio $p = 10\%$

**Figure 1: Accuracy curves with the size of augmenting set $L$ during iterations of ACoMS3VM algorithm on the topic of Google**

$$F\text{-}score = \frac{\sum_y F\text{-}score(y)}{K} \quad (15)$$

For shortness, we use "Acc" to denote Accuracy defined as equation (12), and use "F-s." to denote F-score defined as equation (15) in the following tables.

## 5.3 Results and Comparison

### 5.3.1 *Different sample ratios* $p$

To show the topic adaptive ability, we perform our ACoMS3VM algorithm with an initial training set $L$ randomly sampled from all the 6 topics with ratios $p$. Table 2 illustrated the final sentiment classification results on the metrics of precision and F-score for the 6 adaptive topics. The incensements of our algorithm compared to MSVM are also listed in the last two rows of Table 2, which are averaged on 6 topics. It is seen that we can get the increases of at most 7.08% and 27.44% on accuracy and F-score separately with totally 113 labeled tweets from 6 topics, which is 1% sample ratio. And it is reasonable that the increase becomes less with more labeled training data, since the MSVM algorithm can already get enough supervised information to get a better solution.

To intuitively show the performances of our ACoMS3VM algorithm, we draw the accuracy curves with the size of augmenting set L during iterations as illustrated in Figure 1, from the runs of sample ratios 1% and 10% on the topic of Google. It shows the increases of accuracy with each iteration of optimization, unlabeled data selection and adaptive feature expansion. And all the iterations finally converge into a better position.

### 5.3.2 *Comparisons between baseline*

We compare our adaptive algorithm ACoMS3VM with the 6 baseline algorithms of NB, DT, MSVM, RF, MS3VM and CoMS3VM with in columns 2-15 of Table 3. The accuracies and F-scores are given separately for each baseline on all the 6 topics. The last "Average" row of the table averages the accuracies and F-scores of the 6 testing topics. And the increase percentages of our algorithm comparing to baselines are given in the last "Incr" row. It is indicates that our algorithm ACoMS3VM outperforms other baselines than CoMS3VM by 2.22% to 13.19% increase in accuracy. And even as for the semi-supervised co-training algorithm CoMS3VM with our extraction of features, our algorithm achieves 0.74% and 4.78% increases in accuracy and F-score separately. Although The ensemble learning method RF achieves the best F-score, ours achieves 13.19% accuracy improvement than RF. Therefore, our algorithm ACoMS3VM achieves reliable performances than the baselines.

At last we illustrate some of the initial common sentiment words, and the expanded topic-adaptive sentiment words of each iteration in Table 4 on the topic of President Debate. And Table 5 shows the augmented tweets for different sentiment classes on the topic of Apple. It is seen that the steps of adaptive feature expansion and unlabeled data selection pick reasonable topic-adaptive sentiment words and unlabeled tweets from positive, negative, and neutral classes in semi-supervised way.

# 6. CONCLUSIONS

Various topics are discussed in tweets of Microblog services. Sentiment classifications on tweets suffer from the problem of lack of labeled data, and adapting to some unknown topics in the future. We propose an adaptive multiclass SVM model which transfers an initial common sentiment classifier into a topic-adaptive one, by simultaneously augmenting the labeled set and expanding topic-adaptive features. To solve this nonconvex model, an iterative algorithm alternates among three steps.

- Optimization step is for minimizing the objectives with fixed labeled tweets and topic-adaptive features by solving a multiclass SVM model.

Table 2: Performance on different sample ratios

| $p =$ | | 1% | | 5% | | 10% | | 20% | | 40% | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\|L\|$ | | 113 | | 523 | | 1048 | | 2099 | | 4154 | |
| | | MSVM | ACoMS3VM | MSVM | ACoMS3VM | MSVM | ACoMS3VM | MSVM | ACoMS3VM | MSVM | ACoMS3VM |
| Apple | Acc. | 0.5089 | 0.5508 | 0.5146 | 0.5693 | 0.5284 | 0.5559 | 0.5424 | 0.5700 | 0.5588 | 0.5765 |
| | F-s. | 0.4196 | 0.4660 | 0.4340 | 0.5110 | 0.4356 | 0.4793 | 0.4812 | 0.5275 | 0.5305 | 0.5434 |
| Google | Acc. | 0.6938 | 0.7489 | 0.6877 | 0.7105 | 0.6944 | 0.7300 | 0.6946 | 0.7230 | 0.7235 | 0.7292 |
| | F-s. | 0.4033 | 0.5598 | 0.4852 | 0.5225 | 0.4865 | 0.5696 | 0.4813 | 0.5931 | 0.5808 | 0.5974 |
| Microsoft | Acc. | 0.7412 | 0.7593 | 0.7420 | 0.7608 | 0.7339 | 0.7500 | 0.7395 | 0.7534 | 0.7476 | 0.7458 |
| | F-s. | 0.4585 | 0.5512 | 0.4536 | 0.5254 | 0.4484 | 0.5039 | 0.4593 | 0.5178 | 0.5592 | 0.5532 |
| Twitter | Acc. | 0.8111 | 0.8568 | 0.8127 | 0.8317 | 0.8065 | 0.8252 | 0.8096 | 0.8242 | 0.8107 | 0.8151 |
| | F-s. | 0.4316 | 0.6704 | 0.4546 | 0.5701 | 0.4175 | 0.6646 | 0.4730 | 0.5222 | 0.5611 | 0.5901 |
| Taco Bell | Acc. | 0.5783 | 0.5988 | 0.5994 | 0.6091 | 0.6040 | 0.6143 | 0.6278 | 0.6345 | 0.6350 | 0.6373 |
| | F-s. | 0.3436 | 0.4321 | 0.4321 | 0.4590 | 0.4422 | 0.4752 | 0.5248 | 0.6355 | 0.4986 | 0.5104 |
| President Debate | Acc. | 0.3664 | 0.4471 | 0.4436 | 0.4459 | 0.4463 | 0.4505 | 0.4684 | 0.4729 | 0.6929 | 0.7029 |
| | F-s. | 0.3147 | 0.3426 | 0.3394 | 0.3591 | 0.3603 | 0.3468 | 0.4187 | 0.4249 | 0.5049 | 0.5181 |
| Average Incr% | Acc. | – | 7.08 | – | 3.35 | – | 2.95 | – | 2.47 | – | 0.92 |
| | F-s. | – | 27.44 | – | 13.40 | – | 17.33 | – | 13.48 | – | 2.40 |

Table 3: Comparisons with baselines in 10% sample ratio

| Topics | NB | | DT | | MSVM | | RF | | MS3VM | | CoMS3VM | | ACoMS3VM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc. | F-s. | Acc | F-s. | Acc. | F-s. | Acc. | F-s. | Acc. | F-s. | Acc. | F-s. | Acc | F-s. |
| Apple | 0.5118 | 0.3970 | 0.5098 | 0.3440 | 0.5284 | 0.4356 | 0.5157 | 0.4750 | 0.5353 | 0.4388 | 0.5387 | 0.4609 | 0.5637 | 0.4793 |
| Google | 0.6932 | 0.6060 | 0.6843 | 0.5560 | 0.6944 | 0.4865 | 0.5985 | 0.5820 | 0.7020 | 0.5076 | 0.7872 | 0.5686 | 0.7298 | 0.5696 |
| Microsoft | 0.7364 | 0.6640 | 0.7438 | 0.6350 | 0.7339 | 0.4484 | 0.7649 | 0.6900 | 0.7364 | 0.4600 | 0.7305 | 0.4847 | 0.7500 | 0.5039 |
| Twitter | 0.7708 | 0.7220 | 0.8152 | 0.7320 | 0.8066 | 0.4175 | 0.7364 | 0.7320 | 0.8166 | 0.3784 | 0.8058 | 0.4614 | 0.8252 | 0.5167 |
| Taco Bell | 0.5859 | 0.4930 | 0.5815 | 0.4280 | 0.6041 | 0.4422 | 0.4914 | 0.4750 | 0.6078 | 0.4502 | 0.6125 | 0.4630 | 0.6143 | 0.4752 |
| President Debate | 0.4443 | 0.3140 | 0.4581 | 0.2880 | 0.4463 | 0.3468 | 0.3750 | 0.3760 | 0.4574 | 0.3642 | 0.4376 | 0.3422 | 0.4581 | 0.3690 |
| Average | 0.6237 | 0.5327 | 0.6321 | 0.4972 | 0.6356 | 0.4295 | 0.5803 | 0.5550 | 0.6426 | 0.4332 | 0.6521 | 0.4635 | 0.6569 | 0.4856 |
| Incr % | 5.31 | -8.83 | 3.91 | -2.32 | 3.34 | 13.07 | 13.19 | -12.50 | 2.22 | 12.1 | 0.74 | 4.78 | – | – |

- Unlabeled data selection step is for selecting the confident and unlabeled data of the topic we are transferring to, and augmenting the labeled tweets with them.

- Adaptive feature expansion step is for expending more topic-adaptive sentiment words from those selected unlabeled data from the topic.

With the first two steps alternating to collaboratively augment the labeled data, the third is executed to expand topic-adaptive sentiment words from those selected unlabeled tweets. And in turn to help the first two steps find more confident and unlabeled tweets to boost the final performance of sentiment classification.

# 7. ACKNOWLEDGMENTS

Thanks a lot for the supply of corpuses from Mr. Niek Sanders of Sanders Analytics LLC, Nick Diakopoulos and Ayman Shamma, etc.

This paper is partially supported by National Natural Science Foundation of China (No. 61202213, 61202215, 61173008, 61232010), National Grand Fundamental Research 973 Program of China (No. 2013CB329602), Projects of Development Plan of the State High Technology Research (No. 2012AA011003) and Beijing nova program (No. Z121101002512063).

# 8. REFERENCES

[1] K. Bennett, A. Demiriz, et al. Semi-supervised support vector machines. *Advances in Neural Information processing systems*, pages 368–374, 1999.

[2] J. Blitzer, M. Dredze, and F. Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*, volume 7, pages 440–447, 2007.

[3] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100. ACM, 1998.

[4] M. Collins and Y. Singer. Unsupervised models for named entity classification. In *Proceedings of the joint SIGDAT conference on empirical methods in natural language processing and very large corpora*, pages 189–196, 1999.

[5] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2:265–292, 2002.

[6] N. A. Diakopoulos and D. A. Shamma. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1195–1198. ACM, 2010.

**Table 4: Adaptive sentiment words expanded on the topic of President Debate**

| sentiment class | initial | 1st expansion | 2nd expansion | 3rd expansion | 4th expansion | 5th expansion |
|---|---|---|---|---|---|---|
| | beautiful | totally | directly | defensive | overall | enough |
| | moderate | impressed | presidential | rapid | OK | middle |
| positive | sound | uneventful | debate | top | defensive | very |
| | adorable | condescending | dumb | economic | current | most |
| | awesome | strongest | federal | ready | much | over |
| | want | next | different | terrorist | republican | ahead |
| | completely | never | probably | specific | kinda | present |
| neutral | sustained | instead | post-debate | disrespectful | especially | public |
| | understanding | live | tortured | polar | sudden | bizarre |
| | keep | last | own | ever | crazy | yeah |
| | hate | tired | overall | national | up | away |
| | black | No | emotional | again | ago | forward |
| negative | alarming | Second | corporate | there | maybe | anyway |
| | stupid | Global | professorial | far | anymore | wow |
| | dislike | short | dead | less | nicely | social |

**Table 5: Selected unlabeled tweets on the topic of Apple**

| tweets | class |
|---|---|
| Today I was introduced as BigDealDawson at #LGFW ! O #twitter and #social media I love you! Teehee xx | |
| “I’m starting to get really concerned, sending hashtags in emails :P #twitter is taking over our lives :D” | |
| I’ve pretty much abandoned Facebook for Twitter. #twitter’slegit | positive |
| Gotta love #Twitter - shit goes round the World like lightning-on-speed... | |
| #I #am #so #good #at #twitter ;) | |
| @codytigernord Just a reminder that you fail on #twitter | |
| “Caramba, a mais de 4 dias ninguem fala em mim ou de mim aki no #twitter. Nem mesmo minha noiva. #Fuieskecido...” | |
| “And by the way, why did my iPhone 4 have to loose #Siri to get #Twitter?” | neutral |
| “Going in :( #work. Break at 2:30 and 5:30 #twitter time. See ya’ll in the #am | |
| “My #twitter age is 1 year 19 days 13 hours 37 minutes 17 seconds. Find out yours at http://t.co/XhRUA9Dz #twittertime | |
| “@D_REALRogers BE SLEEP N WHO NOEZ WHERE I BE BUT HOW IT’s it U so called sleep but every 5 sec u got a new #twitter post up #btfu | |
| RT @FuckingShinez: #Twitter = #Dead "this is why im never on it now" | |
| Just hit my hourly usage limit on #twitter. How does that even happen? All I’m doing is listing people... and I was almost done! #ugh | |
| “RT @mainey_maine: RT @ItalianJoya i better be able to see my RT’s tomorrow #twitter and tell that lil blue ass bird, (cont) http://t.co/xGHoev8k” | negative |
| Not really.. I rather study my notes than studying #twitter.. | |
| #Twitter are you freaking kidding me #wth... http://t.co/zKn2bu5R | |


[7] G. Fung and O. L. Mangasarian. Semi-superyised support vector machines for unlabeled data classification. *Optimization methods and software*, 15(1):29–44, 2001.

[8] S. Gao and H. Li. A cross-domain adaptation method for sentiment classification using probabilistic latent analysis. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1047–1052. ACM, 2011.

[9] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, pages 1–12, 2009.

[10] N. Godbole, M. Srinivasaiah, and S. Skiena. Large-scale sentiment analysis for news and blogs. *ICWSM*, 7, 2007.

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[12] Y. He, C. Lin, and H. Alani. Automatically extracting polarity-bearing topics for cross-domain sentiment classification. 2011.

[13] M. Hu and B. Liu. Mining opinion features in customer reviews. In *AAAI*, volume 4, pages 755–760, 2004.

[14] O. I., M. C., L. J., and S. I. Overview of the TREC 2011 microblog track. In *TREC'11*, 2011.

[15] S. I., O. I., and L. J. Overview of the TREC 2012 microblog track. In *TREC'12*, 2012.

[16] T. Joachims. Transductive inference for text classification using support vector machines. In *ICML*, volume 99, pages 200–209, 1999.

[17] S. Kiritchenko and S. Matwin. Email classification with co-training. In *Proceedings of the 2011 Conference of the Center for Advanced Studies on Collaborative Research*, pages 301–312. IBM Corp., 2011.

[18] E. Kouloumpis, T. Wilson, and J. Moore. Twitter sentiment analysis: The good the bad and the omg! In *ICWSM*, 2011.

[19] O. Kucuktunc, B. B. Cambazoglu, I. Weber, and H. Ferhatosmanoglu. A large-scale sentiment analysis for yahoo! answers. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 633–642. ACM, 2012.

[20] F. Li, C. Han, M. Huang, X. Zhu, Y.-J. Xia, S. Zhang, and H. Yu. Structure-aware review mining and summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 653–661. Association for Computational Linguistics, 2010.

[21] F. Li, N. Liu, H. Jin, K. Zhao, Q. Yang, and X. Zhu. Incorporating reviewer and product information for review rating prediction. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three*, pages 1820–1825. AAAI Press, 2011.

[22] F. Li, S. J. Pan, O. Jin, Q. Yang, and X. Zhu. Cross-domain co-extraction of sentiment and topic lexicons. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 410–419. Association for Computational Linguistics, 2012.

[23] S. Li, C.-R. Huang, G. Zhou, and S. Y. M. Lee. Employing personal/impersonal views in supervised and semi-supervised sentiment classification. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 414–423. Association for Computational Linguistics, 2010.

[24] S. Li, Z. Wang, G. Zhou, and S. Y. M. Lee. Semi-supervised learning for imbalanced sentiment classification. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three*, pages 1826–1831. AAAI Press, 2011.

[25] K.-L. Liu, W.-J. Li, and M. Guo. Emoticon smoothed language models for twitter sentiment analysis. In *AAAI*, 2012.

[26] R. Mehta, D. Mehta, D. Chheda, C. Shah, and P. M. Chawan. Sentiment analysis and influence tracking using twitter. *International Journal of Advanced Research in Computer Science and Electronics Engineering (IJARCSEE)*, 1(2):pp–72, 2012.

[27] Y. Mejova and P. Srinivasan. Crossing media streams with sentiment: Domain adaptation in blogs, reviews and twitter. In *ICWSM*, 2012.

[28] A. Mukherjee and B. Liu. Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 339–348. Association for Computational Linguistics, 2012.

[29] L. T. Nguyen, P. Wu, W. Chan, W. Peng, and Y. Zhang. Predicting collective sentiment dynamics from time-series social media. In *Proceedings of the First International Workshop on Issues of Sentiment Discovery and Opinion Mining*, page 6. ACM, 2012.

[30] K. Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In *Proceedings of the ninth international conference on Information and knowledge management*, pages 86–93. ACM, 2000.

[31] K. Nigam and R. Ghani. Understanding the behavior of co-training. In *Proceedings of KDD-2000 workshop on text mining*, volume 14. Citeseer, 2000.

[32] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, pages 751–760. ACM, 2010.

[33] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

[34] M. Thelwall, K. Buckley, and G. Paltoglou. Sentiment in twitter events. *Journal of the American Society for Information Science and Technology*, 62(2):406–418, 2011.

[35] I. Titov and R. McDonald. A joint model of text and aspect ratings for sentiment summarization. *Urbana*, 51:61801, 2008.

[36] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. *ICWSM*, 10:178–185, 2010.

[37] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 417–424. Association for Computational Linguistics, 2002.

[38] V. N. Vapnik. An overview of statistical learning theory. *Neural Networks, IEEE Transactions on*, 10(5):988–999, 1999.

[39] X. Wan. Co-training for cross-lingual sentiment classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 235–243. Association for Computational Linguistics, 2009.

[40] N. Yu and S. Kübler. Filling the gap: Semi-supervised learning for opinion detection across domains. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 200–209. Association for Computational Linguistics, 2011.