PerSentiment: A Personalized Sentiment Classification System for Microblog Users

Kaisong Song[1], Ling Chen[2], Wei Gao[3], Shi Feng[1], Daling Wang[1], Chengqi Zhang[2]
[1]School of Computer Science and Engineering, Northeastern University, Shenyang, China
[2]Centre for Quantum Computation and Intelligent Systems, University of Technology, Sydney, Australia
[3]Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha, Qatar
kaisongsong@gmail.com, {ling.chen, chengqi.zhang}@uts.edu.au
wgao@qf.org.qa, {fengshi, wangdaling}@ise.neu.edu.cn

ABSTRACT

Microblogging services are playing increasingly important roles in our daily life today. It is useful for microblog users to instantly understand the sentiment of a large number of microblogs posted by their friends and make appropriate response. Despite considerable progress on microblog sentiment classification, most of the existing works ignore the influence of personal distinctions of different microblog users on the sentiments they convey, and none of them has provided real-world personalized sentiment classification systems. Considering personal distinctions in sentiment analysis is natural and necessary as different people have different language habits, personal characters, opinion bias and so on. In this demonstration, we present a live system based on Twitter called PerSentiment, an *individuality-dependent* sentiment classification system which makes the first attempt to analyze the personalized sentiment of recent tweets and retweets posted by the authenticated user and the users he/she follows. Our system consists of four steps, i.e., requesting tweets via Twitter API, preprocessing collected tweets for extracting features, building personalized sentiment classifier based on a novel and extensible Latent Factor Model (LFM) trained on emoticon-tagged tweets, and finally visualizing the sentiment of friends' tweets to provide a guide for better sentiment understanding.

Categories and Subject Descriptors

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

Keywords

Sentiment Classification, Latent Factor Model, Microblogs

1. INTRODUCTION

In recent years, microblogging services (e.g., Twitter and Weibo) have become increasingly popular for allowing users to conveniently express their personal feelings and opinions about all kinds of issues in real time. Nearly 60 million tweets are published everyday in Twitter[1]. Mining the sentiment of such a huge amount of subjective texts has recently attracted tremendous research and commercial interest, because of its practical applications in various domains. For example, applying sentiment analysis on microblogs about products was used to investigate consumer attitudes towards products and brands. Sentiment analysis using twitter data even used in [8] for predicting presidential election results.

Although sentiment classification has been widely studied, most of the existing efforts in sentiment analysis [3, 6, 9] ignore the personal distinctions among different users who provide the subjective texts. Microblog users commonly exhibit different styles when expressing their feelings. While using the same wording, people may deliver distinct sentiment orientations. For instance, even if two users post the same tweet like "*I lose 3 pounds!*", it is possible that one expresses positive sentiment because he recently joins some fitness program, while the other is negative because of working overtime for a couple of days. Inspired by the technological advances of recommender systems [1, 2], we found that personal historical microblog posts and users' social relationships contribute to capturing latent personal distinctions (or individuality) of microblog users, which supplies the basis for personalized sentiment classification on microblog posts [7].

The state-of-the-art systems for predicting sentiment orientations (e.g., positive, negative and even neutral) [5, 10] do not deal with personalization and lack the capability of capturing different sentiment orientations in the similar posts of different microblog writers. Therefore, we aim to implement a personalized sentiment classification system which can spot individuality-dependent sentiment of tweets, i.e., sentiment reflecting the personal distinction (e.g., language habit, personal character, opinion bias and so on) depending on each writer and the already exchanged content. Figure 1 presents an example tweets list that is displayed in the homepage of a typical Twitter user. All tweets posted by the user and his/her friends are ordered chronologically (known as home timeline). It is often the case that a large number of tweets to be posted by friends (i.e., followees) in a short time, especially when a user follows a lot of friends. This brings heavy burdens for a user to understand instantly the sentiment of his friends and as a consequence to response promptly. In this work, there-

[1]`http://www.statisticbrain.com/twitter-statistics/`

WWW'16 Companion, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4144-8/16/04.
DOI: http://dx.doi.org/10.1145/2872518.2890540.

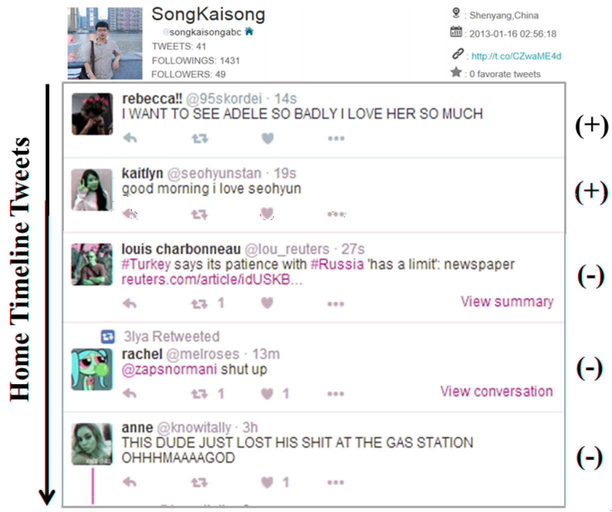

Figure 1: Example of home timeline tweets of the current user (a tweet list ordered in the homepage of a Twitter user). Notation + denotes positive sentiment and - represents negative ones.

fore, we develop a live and a personalized sentiment classification system called *PerSentiment*, publicly available at `http://52.33.127.210/sentiment/index.jsp`, which provides an easy way to explore the sentiment of the authenticated user's tweets and his/her friends via using visual charts. Meanwhile, our system also provides an auxiliary interface which allows users to experience real-time personalized sentiment analysis by viewing the sentiment orientation of a selected tweet.

2. RELATED WORK

Recently, there has been a growing interest in capturing the sentiment expressed on microblogging data [3, 6, 9]. However, these studies did not deal with personalization. Personalized sentiment model was rarely explored for microblog users. Li et al. [4] proposed a tensor factorization model by incorporating reviewer and product information, but it cannot be directly applied for our microblogging texts which are independent of specifiable products or objects.

Latent factor model (LFM), a widely used method of matrix factorization in online recommendation systems, can capture the hidden elements determining users' preferences which are commonly difficult to analyze. Agarwal et al. [1] proposed a factor model that incorporates rater-comment and rater-author interactions to rank the comments associated with a given article according to user preference. Chen et al. [2] proposed a collaborative tweet ranking model for recommending tweets to users. Different from these works, our demo is based on a novel extensible LFM for personalized tweet sentiment modeling [7] that incorporates the latent factors induced from *social, sentimental and topical evidence* into the matrix factorization process.

Most of the available systems for sentiment classification on microblogs are non-personalized. Zhao et al. [10] and Lipenkova [5] implemented non-personalized sentiment classification systems based on Weibo and Twitter services, respectively. To the best of our knowledge, our demo is the first available personalized sentiment clasification system.

3. USE CASE

Consider a user that logins to our system with his/her Twitter account. Our system will access his/her personal tweets securely under Twitter user authorization mechanism. We elaborate how to use *PerSentiment* by showing the system interface in Figure 2 which is mainly composed of three functional pages: *Home Timeline Analysis*, *User Timeline Analysis* and *Realtime Tweet Analysis*.

The *Home Timeline Analysis* page (Figure 2 (a)) displays the authenticated user's profile (e.g., head portrait, # of tweets, # of followings, # of followers, etc.) obtained from Twitter after successful login. Below the current user's profile, we display the top n (n=50) home timeline tweets and some other details including the head portraits of tweet posters, screen names, text contents, and the predicted sentiment labels of the corresponding microblog messages. The user can also keep track of newest home timeline tweets at all times by clicking the refresh button, just like Twitter. Meanwhile, our system allows convenient responses such as reply, retweet and like for the user to interact with other users timely and accurately according to the prediction. Moreover, clicking the head portrait of a particular user, the system is conducting sentiment analysis for that user's timeline, i.e., provides the personalized sentiment of the tweets.

The *User Timeline Analysis* page (Figure 2 (b)) displays a specified friend's profile and sentiment analysis results of his/her personal tweets. We show top n (n=50) user timeline tweets and present analysis results using pie chart and line chart. The pie chart provides an overview on tweet sentiment by showing the proportion of positive, negative and neutral tweets of the particular user. The line chart demonstrates the changes and trend of the user's tweet sentiment over time. Sentiment scores above 0.5 denote a positive sentiment trend, otherwise a negative trend. The time control slide bar below the line chart allows the authenticated user to view sentiment trend in any time interval by adjusting the bar. Meanwhile, the details of a tweet and its sentiment label are also displayed at the bottom of the page.

The *Realtime Tweet Analysis* page provides an interactive tweet sentiment analysis for any tweet that a user directly input at the text box in the interface (Figure 2 (c)). Note that this test module does not really publish or forward the test tweets onto Twitter for avoiding disturbing other users. The predicted sentiment score and the corresponding class are displayed above the input box.

4. SYSTEM IMPLEMENTATION

PerSentiment is designed as a browser-server Web application based on our extensible latent factor model for personalized sentiment classification [7]. Although the system builds upon the Twitter platform, it is straightforward to be deployed in other microblogging services like Weibo. Figure 3 shows the system architecture which consists of four modules: 1) PerSentiment GUI Module, 2) Tweet Crawling Module, 3) Tweet Preprocessing Module, and 4) Sentiment Classification Module. The solid line arrows in the figure indicate data flows between the browser and Twitter server; the dotted line arrows denote internal data flows. Next, we will describe the implementation detail of these modules.

4.1 PerSentiment GUI Module

As shown in Figure 2, the PerSentiment GUI consists of three main functional pages that display the user profile and

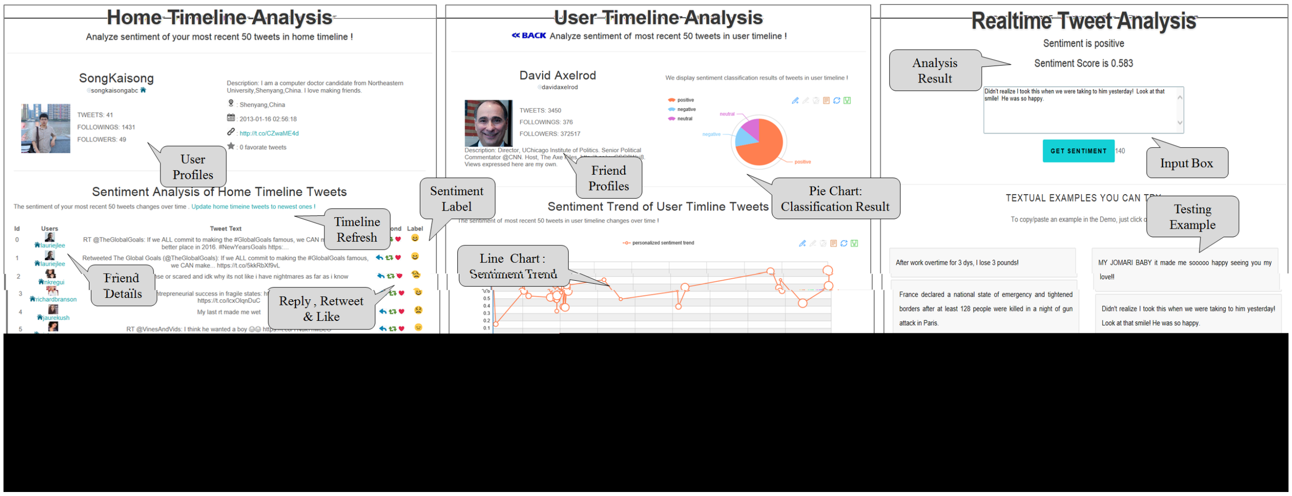

Figure 2: The use cases of PerSentiment system

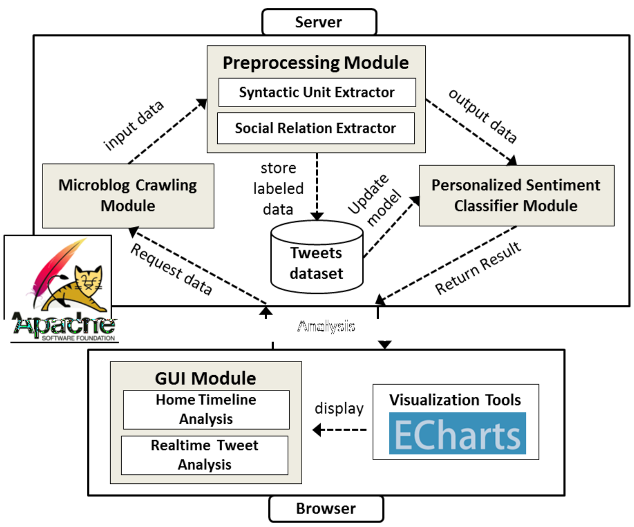

Figure 3: The architecture of PerSentiment

the results of sentiment analysis conducted automatically by the classification model. Many off-the-shelf tools are publicly available for implementing results visualization. Here we sort to Echarts[2], which provides mature methods for integrating different categories of charts into the functional pages while also allows customization for individual users.

4.2 Microblog Crawling Module

This module encapsulates a specialized crawler program implemented via the *twitter4j* Java library[3] for the Twitter API. The module works when a user logs into his/her Twitter account and authorizes our system to request the personal data. Specifically, the module sends secure authorized requests via the Twitter API under the authentication framework of OAuth 2.0, so the users' tweets, the social relationship with other users and the user profiles can be collected easily subject to the request rate limitation of Twitter[4]. Note that we do not request all data at a time in consideration of API restriction and response time. Instead, we crawl top the n (n=50) timeline tweets when the system handles the request of each user for analyzing the sentiment of home or user timeline tweets, which ensures the smooth and timely response of the system. Moreover, we do not store all tweets but those with "sentiment labels" for training purpose. Emoticon-based sentiment annotation is common in previous works [3, 10] such as textual emoticons (e.g., :) and :() or graphical ones (e.g, 😄 and 😫). In our case, historical tweets containing non-conflict emoticons are stored and then preprocessed for training our sentiment classification model.

4.3 Microblog Preprocessing Module

This module aims to obtain two important types of features from personal tweets, namely syntactic units and following relationship (see Section 4.4), which are generated by syntactic unit extractor and social relationship extractor, respectively. Syntactic units consider the dependency relation between words in order to relieve the sparsity of personal data and compensate the coarse-grained bag-of-word model with finer-grained representation; Social relationship enhances the captured latent factors of individuality with the intuition that followers share some common interested aspects with their followees thus conveying closely related sentiment [7].

Specifically, the syntactic unit extractor extracts *syntactic units* consisting of sentiment units and topic units based on dependency parsing that resorts to Stanford CoreNLP[5], an integrated suite of natural language processing tool which provides tokenization, part-of-speech (POS) tagging, syntactic parsing and lemmatization. Since noun, verb, adjective and adverb are usually indicative words, we reserve them in the form of 'lemma#pos' based on POS tagging and lemmatization in the first place. Next, we represent each tweet as a group of word pairs such as {sent#v, necklace#n} and {very#r shiny#a} after typed dependency analysis for the words with dependency relation between them. Finally,

[2] `http://echarts.baidu.com/index-en.html`
[3] `http://twitter4j.org/en/index.html`

[4] `https://dev.twitter.com/rest/public/rate-limiting`
[5] `http://nlp.stanford.edu/software/corenlp.shtml`

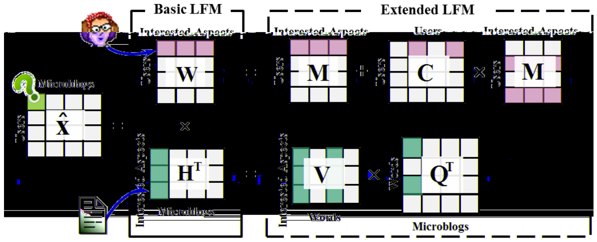

Figure 4: Our personalized sentiment classification model

these units are categorized as sentiment units or topic units considering whether they contain words in the SentiWordNet sentiment lexicon[6] or not.

The social relationship extractor extracts social connections between follower (current user) and followees (friends). We only extract all the direct following connections since the secondary and deeper connection cannot contribute to accuracy and also drags response time in our experiments.

4.4 Personalized Sentiment Classification

Our model is based on a latent factor model. The sentiment score matrix $\mathbf{X}$ can be factorized by the product of low-rank user-factor matrix $\mathbf{W}$ and tweet-factor matrix $\mathbf{H}$ so that the predicted score $\hat{x}_{ui} = \mathbf{W}_u \cdot \mathbf{H}_i^{\mathrm{T}}$, where u is any user and i is the corresponding tweet post published by u, and the latent factors reflect interested aspects shared between the two matrices. The factors are estimated by minimizing the regularized cost function $\sum_{(u,i)}(x_{ui} - \hat{x}_{ui})^2 + regularizer$.

Here, we proposed an extensible model as displayed in Figure 4. We first decompose $\mathbf{H}$ as $\mathbf{Q}\mathbf{V}^{\mathrm{T}}$ where $\mathbf{Q}$ is a tweet-word matrix and $\mathbf{V}$ is a word-factor matrix. Thus, any post $\mathbf{H}_i$ can be represented by a weighted linear combination of words $\mathbf{V}_k$ in post i. Meanwhile, we incorporate users' following relationship into the model to enhance personalization effect, for which we project $\mathbf{W}$ into $\mathbf{M} + \mathbf{C}\mathbf{M}$ with a newly estimated user-factor matrix $\mathbf{M}$ and an observed followee-follower connection matrix $\mathbf{C}$ where each entry $\mathbf{C}_{uv}$ indicates whether u follows v. As a result, the estimation $\hat{x}_{ui} = \mathbf{W}_u \cdot \mathbf{H}_i^{\mathrm{T}}$ can be reformulated as follow:

$$\hat{x}_{ui} = \mathbf{b} + (\mathbf{M}_u + \mathbf{C}_u\mathbf{M}) \cdot \left(\sum_{s \in S_i} \frac{1}{|S_i|} \sum_{k \in s} \mathbf{V}_k + \sum_{t \in T_i} \frac{1}{|T_i|} \sum_{k \in t} \mathbf{V}_k \right)$$

where S_i is the set of sentiment units and T_i is the set of topic units in post i, s is any sentiment unit in S_i and t is any topic unit in T_i, $\mathbf{V}_k$ is the feature vector of component word k, $\mathbf{b} = w + b_u + \sum b_k$ is a first-order baseline predictor for better generalization. We resort to sigmoid function $\sigma(\hat{x}_{ui}) = (1+e^{-\hat{x}_{ui}})^{-1}$ to map the predicted score into $(0, 1)$. We take $\sigma(\hat{x}_{ui}) \in (0.0, 0.45]$ as negative, $\sigma(\hat{x}_{ui}) \in [0.55, 1.0)$ as positive and $\sigma(\hat{x}_{ui}) \in (0.45, 0.55)$ as neutral.

We implement this model using factorization toolkit SVDFeature[7] which is designed to solve the feature-based matrix factorization efficiently. The feature-based setting allows us to define and include informative features (i.e., syntactic units and following relationship) into latent factor model for developing our extensible model. We use stochastic gradient descent for the model training so that the model can be updated continuously with the increase of personal data.

5. DEMONSTRATION AND CONCLUSION

The PerSentiment system is deployed as an Amazon EC2 micro instance provided by Amazon Web Services[8]. Our prototype system has been made publicly available on the website: `http://52.33.127.210/sentiment/index.jsp`.

System performance. This live prototype system is based on our recent research work [7] which achieved high average precision around 0.84 on tweet sentiment classification and significantly outperformed the basic LFM and non-personalized methods using 10-fold cross validation for evaluation. Our system also has an efficient response speed for analyzing sentiment of tweets and can return results in 3 seconds in average. There is a little overhead during system start as it is necessary for the data collection and the model training, but can be done as quickly as 60 seconds or so. All these suggest the superior performance of our system.

Conclusion. We presented *PerSentiment*, a Web-based application on personalized sentiment classification for microblog users. We implemented the prototype system by accessing personal tweets securely under Twitter user authorization. Three main components are realized for presenting the sentiment analysis results of the home and user timeline tweets based on the Web interfaces implemented using visualization tools and providing an interactive personalized sentiment analysis interface.

6. ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (61370074, 61402091), the Fundamental Research Funds for the Central Universities of China (N130604002, N140404012) and China Scholarship Council.

7. REFERENCES

[1] D. Agarwal, B.-C. Chen, and B. Pang. Personalized recommendation of user comments via factor models. In *EMNLP*, pages 571–582, 2011.

[2] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu. Collaborative personalized tweet recommendation. In *SIGIR*, pages 661–670, 2012.

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

[4] F. Li, N. Liu, H. Jin, K. Zhao, Q. Yang, and X. Zhu. Incorporating reviewer and product information for review rating prediction. In *IJCAI*, pages 1820–1825, 2011.

[5] J. Lipenkova. A system for fine-grained aspect-based sentiment analysis of chinese. In *ACL(System Demonstrations)*, pages 55–60, 2015.

[6] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *EMNLP*, pages 79–86, 2002.

[7] K. Song, S. Feng, W. Gao, D. Wang, G. Yu, and K.-F. Wong. Personalized sentiment classification based on latent individuality of microblog users. In *IJCAI*, pages 2277–2283, 2015.

[8] H. Wang, D. Can, A. Kazemzadeh, F. Bar, and S. Narayanan. A system for real-time twitter sentiment analysis of 2012 u.s. presidential election cycle. In *ACL(System Demonstrations)*, pages 115–120, 2012.

[9] F. Wu, Y. Song, and Y. Huang. Microblog sentiment classification with contextual knowledge regularization. In *AAAI*, pages 2332–2338, 2015.

[10] J. Zhao, L. Dong, J. Wu, and K. Xu. Moodlens: an emoticon-based sentiment analysis system for chinese tweets. In *KDD*, pages 1528–1531, 2012.

[6] `http://sentiwordnet.isti.cnr.it/`

[7] `http://svdfeature.apexlab.org/wiki/Main_Page`

[8] `https://aws.amazon.com/`