

SINA WEIBO AS A CORPUS FOR STUDYING PUBLIC OPINIONS

STAT 215B FINAL PROJECT, SPRING 2012

Christine Kuang, Siqu Wu, and Angie Zhu

May 7, 2012

Abstract

In this project, we try to develop a framework to study the public opinions via Sina Weibo – a Twitter equivalence in China. Text data are collected from Weibo via an application programming interface (API). Word processing techniques are used to extract useful information from the raw data. Several standard machine learning tools are applied to feature analysis and sentiment classifications.

1 Introduction

Recent years have seen a growing interest of the analysis of text corpus from social media such as Facebook and Twitter. Gwalt et al ([Gawalt et al., 2010]), for example, study word associations from a series of new articles on NY Times. Due to the increasing size of the corpus, modern sparse machine learning tools are needed in both text summarizing and topic classification. El Ghaoui et al. ([El Ghaoui et al., 2011]) gives an excellent summary of sparse learning tools that are useful in analyzing high dimension text data sets.

So far most of the literatures focus on the study of English text corpus. Researches on Chinese text data have been relative few. Tan and Zhang ([Tan and Zhang, 2008]) conduct an empirical study of sentiment analysis in Chinese documents using a combination of feature selection methods and classification method. Lee and Renganathan ([Lee and Renganathan, 2011]) study Chinese document via a maximum entropy method. In both of the above study, the complex structure of Chinese language makes it difficult to analyze the raw text data directly. We will explore this issue more in Section 3.2.

Due to the restriction on overseas websites such as Facebook or Twitter in China, domestic substitutes have become the major platforms for the internet citizens to express their opinions towards various social or political issues. Studying posts on those website thus provides interesting insights into the public opinion. For instance, some events in the recent years, such as the tragic Wenzhou train collision on July 23 2011¹ and the more recent Wang Lijun incident,² have split the public into two major groups among which one considers the government’s way of dealing with those incidences is good and one does not. In principle, we can draw text data from those websites identifying whether a particular post is related to the event of interest, and which group the post should be classified into.

¹For details, see http://en.wikipedia.org/wiki/Wenzhou_train_collision.

²For details, see http://en.wikipedia.org/wiki/Wang_Lijun_incident.

Sina Weibo 新浪微博 is the largest microblogging website and one of the most popular social network website in China. It had more than 300 million registered users as of February 2012 ([Cao, 2012]) and accounted for 65% of China’s microblog market by pageviews as of December 2011([Lee, 2012]).

In this project, we aim to develop a framework for studying public opinions using Sina Weibo as a corpus for a given topic. Posts are sampled from Weibo and then processed taking the characteristics of both Chinese language and Weibo posts into consideration. Two unsupervised learning tools, the sparse Gaussian graphical model and the sparse PCA are used in feature analysis. Also, the LASSO and the l_1 -norm SVM are used in classification.

2 Data Collection

A Weibo application programming interface (API) is set up to collect posts. For a given topic, the ideal way to collect sample post are taking a random sample. [Boyd et al., 2004, Leskovec and Faloutsos, 2006, Wang et al., 2011] have presented sampling algorithms from large graphs. However, our preliminary results show that the sample contains a large proportion of spams and posts which do not relate to the chosen topic. Therefore, the topic searching functionality of the Weibo API is utilized to collect posts.

There are certain technical restrictions on API. Regarding topical searches, only the latest search results are returned. The maximum number of posts returned is 30 and our own searches typically returned about 18 posts. There is 30 seconds waiting time between consecutive searches. Overlapping among sets of search results is examined. Clearly the sample is time-sensitive. We preferred having a relatively short collection time so as to obtain a cross-sectional sample. Hence, the chosen topic for testing should have active discussion on Weibo.

It is well-known that China has strict Internet censorship compared to other countries. Even though the downfall of Bo Xilai³ was a hot topic at the time of our data collection, there were no meaningful posts on Weibo due to the censorship. The topic we finally chose is Han Han 韩寒.⁴ Born September 23, 1982, Han Han is a Chinese best-selling author, professional rally driver, and wildly popular blogger. He published his first novel *Triple Gate* 三重门 at the age of 17 and later dropped out of high school. Even though Han Han is famous for his criticism on social issues, he has not been heavily censored. A ghostwriting allegation against Han Han started in January 2012. Fang Zhouzi 方舟子,⁵ a scientific author and anti-fraud crusader, created widespread debate with Han Han on the Internet regarding the ghostwriting allegation. In response to the allegation, Han Han published a photocopied manuscripts set *Light and Upright* 光明与磊落 in April 2012, including the manuscript of his first novel *Triple Gate*.

Han Han received an online death threat on April 15, 2012. Our sampling took place immediately after this event on April 16 and 17, 2012. The data consist of 22,398 posts. The above proper nouns and words related to the death threat appeared frequently in our data set.

³For more information on Bo Xilai, see http://en.wikipedia.org/wiki/Bo_Xilai.

⁴For more information on Han Han, see http://en.wikipedia.org/wiki/Han_Han and *Han Han: China’s Literary Bad Boy* by Simon Elegant Monday, published on November 02, 2009, em Time Magazine, <http://www.time.com/time/magazine/article/0,9171,1931619,00.html>.

⁵For more information on Fang Zhouzi, see http://en.wikipedia.org/wiki/Fang_Zhouzi.

3 Processing

3.1 Encoding

Chinese documents use various encoding standards ([Wong et al., 2009]). *GB 国标*, abbreviation for “National Standard” in Chinese, is the standard used in mainland China, Singapore, and Malaysia. Since GB is unable to cover traditional Chinese characters, *GBK 国标扩展*, “Extension of National Standard,” is developed to support both simplified and traditional characters. Meanwhile, *Big5* is the encoding standard used by Taiwan, Hong Kong, and Macau. *Unicode* is an industry standard for the consistent encoding of text expressed in most of the world’s writing systems. *UTF-8* is the most commonly used encoding implementation of Unicode. All our data and related files use UTF-8 encoding for processing purpose.

3.2 Characteristics of Chinese Language

There has been many researches done on natural language processing in the English language, but not much on the Chinese language. This is due to the fact that the Chinese language contains unique characteristics that makes it difficult to do natural language processing on it well.

First, the Chinese language, unlike the English language, has no explicit delimiter between words. In the English language, spaces separate words and so to segment a sentence into its appropriate components is not too difficult since one could just separate based on spaces. The Chinese language however has no such delimiter between words.

An additional difficulty in segmenting a sentence into its appropriate components and words is that the Chinese language is made of separate characters where each character has a meaning of its own, but if you combine two or more characters together into a phrase, the meaning can be completely different. These types of ambiguities are typically easily resolved by humans reading the sentence and realizing what would make most sense in terms of the context of the sentence, but it is not so easy for a computer to perform those types of automatic word/phrase segmentations. For example, 他好吃 – this sentence could be separated in two ways. The first character means “he.” The other two words can be taken as two different phrases: “tastes delicious” or “loves to eat.” It is obvious that the phrase should be taken as “he loves to eat” because “he tastes delicious” does not make sense but it is difficult to have a computer automatically recognize which sentence makes more sense. Some words also have more than one meaning. For example, 打 can be used in different ways with different meanings. It can be used in the contexts of playing a sport, hitting a person or object, or playing a game.

Second, the Chinese language has many out-of-vocabulary (OOV) words . These are new phrases that are not found in dictionaries. These phrases typically result from cultural references, current hot topics, acronyms, abbreviations, names/nicknames, or informal and slang words. Specifically to the posts that we looked into, out of vocabulary words typically occurred in the case of cultural references. Many of these social media posts would contain nicknames created for some hot topic issue, person, or event of the week or simply popular slang or informal words and phrases used to convey an emotion. Since social media posts are more popular with the young adult generation, many of the words used in the posts are popular informal phrases that would not normally be found in established dictionaries. Knowledge of these types of OOV words comes out of knowing the current trends in Asian countries and being up to date with the typical language used by the younger generation.

Third, the Chinese language has two forms - traditional and simplified and it is not a one to one correspondence between the two forms which makes it difficult to convert between the two for translation or natural language processing.



Figure 1: An example of Weibo posting. The picture showed Han Han met with Ma Ying-jeou, the President of the Republic of China (Taiwan), on May 3, 2012.

3.3 Characteristics of Sina Weibo Posts

Our analysis takes not only the characteristics of Chinese language into consideration, but also the characteristics of Sina Weibo posts. The writing style of Weibo posts are generally informal. The users may not use standard punctuation marks for separation of sentences and parts of sentences. The most important features are described as follows:

Multiple Forms A post can be text, picture, and/or multimedia as shown in Figure 1. Weibo API is only able to return posts in text. For instance, we record a post in the following format:

1714080953 2012-04-16 10:01:59 //@ 風笑巨石: 那尊神容不得别人质疑? //@ 伯林 2011: 质疑派人士遭到人肉, 人身攻击, 甚至死亡威胁的时候, 从来没有污名化整个挺韩派, 也没有引起什么媒体关注, 相比之下, 韩寒如此炒作, 太无良了, 挺韩和批韩的双方本不至于如此撕裂

The post starts with the user identification number and time stamp. We ignore pictures and videos at this stage.

Reposting A user may repost another post. Reposting does not automatically imply agreement or liking. This type of post usually consists of two parts: the reposting user's comment and the post being reposted. The reposting user's comment may be empty or set as the default text "Repost" or "转发微博" ("Repost Weibo"). The reposted post itself may include multiple repostings. The reposting part usually starts with "//@" followed by a user name as demonstrated in the above example and Figure 1. The issue of keeping track of repostings and identifying agreements or disagreements to the topic reposted about can be a project in

itself. In our analysis, only the reposting user’s comment is kept for the sake of simplicity and time.

Spams There are a fair amount of spams on Weibo. Some spam posts are identical except for the URL. Hence, URLs are first removed and then we check for duplicated posts in the pre-tagging processing step.

Mentioning A user may mention other users in their post by inserting their usernames preceded by the @ symbol. The mentioned usernames may be integrated into the post. We define a set of topic-related usernames and substitute the mentioning of these usernames by the corresponding proper nouns. The other mentioned usernames are removed.

Emotion Symbols Sina Weibo provides users a set of emotion symbols which are denoted with a phrase indicating its corresponding meaning surrounded by square brackets. The users may use other emotion symbols, such as “:)” for a smile and “T_T” for a crying face. The two classes of graphic characters in CJK (Chinese, Japanese and Korean) computing,⁶ namely fullwidth and halfwidth, make emotion symbols more complicated. The punctuation marks in the emotion symbols can be in either fullwidth or halfwidth, such as fullwidth “。” and halfwidth “.” for period. Thus, there are various combinations, even for a limited set of commonly used emotion symbols. A dictionary for emotion symbols is constructed manually by the authors.

Internet Slangs Internet slangs are a large part of Out-of-Vocabulary (OOV) words. Some substitute the characters in a word with characters that have similar pronunciations, such as “蜀黍” (pronunciation: shu3 shu2) for “叔叔” (pronunciation: shu1 shu1, means “uncle”) and “围脖” (pronunciation: wei2 bo2, means “scarf”) for “微博” Weibo (pronunciation: wei1 bo2). Some are popular Internet interjections, such as “喵了个咪” (喵: “meow,” 了: past tense marker, 个: universal measure word, 咪: “mew”) which means “dog my cats.” Again, a dictionary for commonly used Internet slangs is constructed manually by the authors.

Topic Since the Chinese language has no explicit delimiter between words, topic words are surrounded by the pound signs # . The topic words can be integrated into the post.

3.4 Pre-tagging Processing

Our data set contains 22,398 posts. Due to the fair amount of spam posts and the issue of tracking down repostings described in Section 3.3, the data need to be cleaned before the constructing of the training and testing sets. A typical post looks like:

1165303315 2012-04-16 09:55:40 《韩寒收到网友死亡威胁》(来自 @ 新浪娱乐)
http://t.cn/zOprKap

1165303315 2012-04-16 09:55:40 《Han Han received death threat online》(from @ 新浪娱乐) http://t.cn/zOprKap

The pre-tagging processing consists of the following steps:

1. The user identification number and time stamp are removed.
2. Only the reposting user’s comment is kept. The reposted part is removed from further analysis. If the resulting string is empty, it will be eliminated as well.

⁶For more details, see http://en.wikipedia.org/wiki/Halfwidth_and_fullwidth_forms.

3. URLs are removed.
4. Duplicates are removed.

The resulting file consists of 13,070 unique posts. A set of randomly selected 3,500 posts, among which 3,000 are for training and 500 for testing, will be manually tagged.

3.5 Tagging

In all, we tagged a total of 3,500 Weibo posts. We made a search for the topic we were looking into and did this over time to get a total of 22,398 Weibo posts, with 13,070 posts left after pre-processing, from which we randomly picked out 3000 posts as our training set. We split up the posts to tag by hand. The final 500 posts were for our test set. We had a total of four different categories that we tagged the posts as – neutral/unidentifiable, positive, negative, and irrelevant (spam).

As we each tagged the posts, we encountered and realized some of the limitations involved. One limitation was in the fact that tagging these posts produced subjective responses. What one of us read as a negative response to the topic we chose, another may have read as a positive response. For example, the English phrase “that wasn’t too bad” could be taken as positive or negative. Positive – the experience was better than expected; negative – the experience wasn’t great. Hence, it is difficult sometimes to truly know whether the original author of the post had in mind a negative or positive reaction to the topic he was posting about.

Another limitation was that some posts we were not sure how to tag. Many posts consisted of just a quote by the author we were looking into. These could have been seen as positive posts since the writer may have liked the quote and so posted it. But at the same time, the author could have been neutral and was just merely using the quote to apply to a specific circumstance in his life at the time. We were not entirely sure of how to tag each of the posts that fell into this category, and so again, the subjectiveness of tagging the posts by hand comes into play as a limitation in the forming of our model. Another uncertainty that occurred in trying to manually tag the posts was that some of the posts didn’t talk about our chosen topic specifically, but a related topic. With our chosen topic, people who had negative responses towards 韓寒 (Han Han), were usually on the side of the opposing author who was discrediting him. Some of the posts did not directly mention 韓寒, but would instead show support for the opposing side. With these posts, we typically labeled as a negative response. However, an argument could be made for just throwing out those posts since they do not directly say anything about 韓寒, and they could possibly just be supporting the opposing author in his own literary works and not necessarily in his stance against 韓寒. Another uncertainty in tagging is what to tag posts that have no subject. There were a few posts that had nothing to do with the chosen topic at all, but there were also posts that had no subject but contained phrases such as “Keep it up!” and “always a supporter!” that could very well be taken as positive posts since our search is for our specific topic. But because there is no subject in the posts, this cannot be taken with 100% certainty. In these instances, where there is no subject in the posts, we marked them as irrelevant to be on the conservative side in our predictions.

These limitations in tagging will affect our model’s accuracy in predicting whether a post contains a positive or negative response to the topic at hand. These limitations also indicate to us that in general, models for predicting whether or not a social media post is negative or positive towards a chosen topic is limited greatly by the subjectiveness of the sentence’s meaning and interpretation which affects the training set used to form the model. This limitation could present a potential future question to look into on how we can improve tagging – whether we should just

remove posts that have too subjective of an interpretation to tag or if by studying these types of ambiguous posts, we can create certain priors or statistical models to help in the tagging process.

3.6 Pre-segmentation Processing

As discussed in Section 3.2, sentences in Chinese are normally strings of Chinese characters without spaces between words. Hence, word segmentation is crucial for our word-based analysis. According to the characteristics of Weibo posts described in Section 3.3, the following processing steps are preformed:

1. A set of topic-related usernames are defined. Any mentioning of these usernames are substituted by the corresponding proper nouns. The other mentioned usernames are removed.
2. A set of emotional symbols and Internet slangs are defined. They are then substituted by the corresponding word surrounded by square brackets.

3.7 Segmentation

汉语词法分析系统 ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System) is a well known Chinese word segmentation system developed by [Institute of Computing Technology, Chinese Academy of Sciences, 2011]. It offers the functionality of Chinese word segmentation, lexical tagging, named entity recognition, unknown words detection, and the user-defined dictionary. The current version is ICTCLAS 2011, which supports GB2312, GBK, UTF8 and several encodings and has a precision rate of 98.45%.

The Java version of ICTCLAS 2011 performs word segmentation and lexical tagging on a Linux 32-bit machine. A user-defined dictionary is provided. The entries in this dictionary contain proper nouns and Internet specific words. For instance, some users refer to Han Han as 韩少 (韩: Han Han's surname, 少: abbreviation of 少爷, which means "young master of the house"). Even with the user-defined dictionary, some appearances of Han Han's name 韩寒 can not be segmented and tagged correctly. Hence we account and correct for this directly using regular expression.

3.8 Conjunction Rules

[Lee and Renganathan, 2011] noted that special consideration should be given to the sentences whose parts are linked by contrasting transitional expressions. In particular, if a sentence contains conjunctions such as "although" and "but," only the part being emphasized will be kept and used to infer the sentiment polarity of the sentence. We considered these three cases:

1. Although (part A), (part B).
2. (Part A), but (part B).
3. Although (part A), but (part B).

For each case, only part B will be kept.

The four words for "although" are 虽然, 虽说, 虽, and 尽管. The words for "but" are 但, 但是, 不过, 可是, 然而, 只是, 可, 只, 然, and 却.

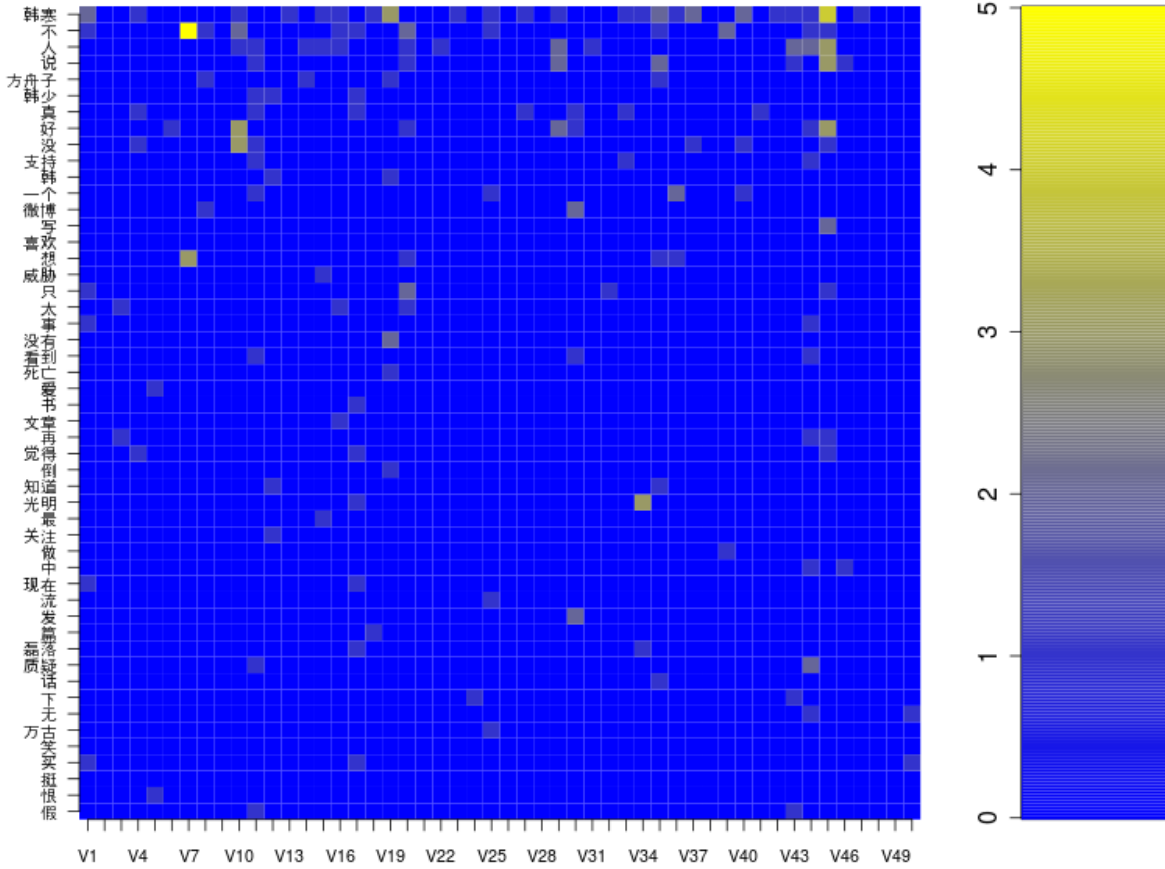


Figure 2: Plot of the transpose of the frequency matrix.

3.9 Stop Words and Punctuation Elimination

Moreover, stop words, non-text strings, and punctuation marks are eliminated based upon the lexical tags generated by ICTCLAS. The detailed process is as follows:

1. Remove prepositions, punctuation marks, English character strings, interjections, modal particles, onomatopoeia, and auxiliary words.
2. Remove pre-defined stop words and number strings.

Note that the pre-defined stop words do not contain the following six negation words: 不, 不是, 没有, 没, 无, and 别. These negation words will be used in sentiment score assignment.

4 Feature Analysis

It is of interest to study the relation between words or phrases in the posts. In what follows, we will base our analysis on the frequency matrix $X \in R^{n \times p}$, where the entry x_{ij} stores the frequency of occurrences of the j -th word(or phrase) in the i -th post. We include only words or phrases that

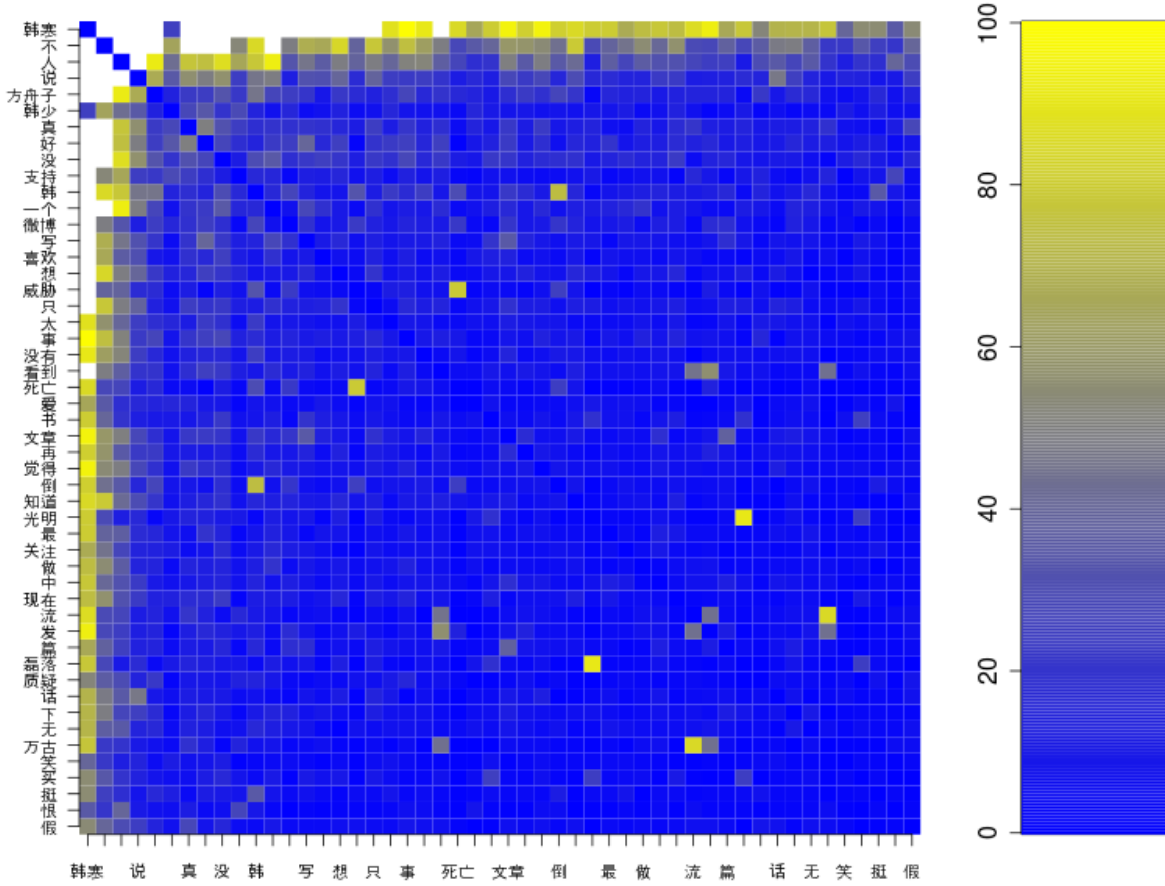


Figure 3: Plot of the word concurrence matrix C . Only top 50 most frequent words are shown.

have total occurrence of at least 10 times over all 3000 posts. Analyzing feature relations can help us better understand word usage and identify possible clusters in the feature space. Figure 2 shows the frequency of the top 50 most frequent words in the first 50 posts.

One possible way to visualize word usage relation is to look at the concurrence between any two pair of words. Denote by C the concurrence matrix, where the entry c_{ij} records the number of concurrencies of the i -th and the j -th word in the same post. Figure 3 and 4 give the matrix plot and the network plot of the top 50 most frequent words.

As can be seen from Figure 2, our data is quite sparse; most words do not appear too many times in each post. Due to the sparseness of our data, we chose methods of modeling our data accordingly (as will be described in further detail in later sections). We use high dimensional tools to take into account the sparseness of our frequency matrix. This frequency matrix will be our design matrix for our models.

Table 1: Sparse PCA results on the frequency matrix ($\lambda_1 = \lambda_2 = \lambda_3 = 10$).

1st PC top abs.coef.	2nd PC top abs.coef	3rd PC top abs.coef
韩寒 (Han Han)	方舟子 (Fang Zhouzi)	死亡 (death)
韩少 (Master Han)	够 (enough)	韩 (Han)
写 (write)	质疑 (suspect)	威胁 (Threat)
真 (truth)	韩 (Han)	倒 (oppose)
死亡 (death)	事 (event)	网友 (netizen)

It is surprising how the above results from sparse PCA are related to our prior knowledge of the subject. The first principal seems to be about the Han Han the person and perhaps his writing; the second column is related to the fact that the fraud-crusader Fang Zhouzi suspects the authorship of HanHan’s works; the third is about the death threat to Han Han. Thus all these three components seem to correspond to the three major aspects of Han Han that are known to us prior to our analysis. See Section 2 again.

4.2 Sparse Graphical Models

Undirected graphical models are commonly used in machine learning to study the relation between random variables ([Wainwright and Jordan, 2008]). Each node of the graph represents a random variable and an edge connecting two nodes represents the conditional dependency between the two random variables given all other random variables (and the missing of an edge indicates conditional independency). If, in addition, the joint distribution of the random variables is a multivariate normal with mean μ and covariance matrix Σ , then the i -th node and j -th node are conditionally independent (or, equivalently, missing an edge) if and only if $(\Sigma^{-1})_{ij} = 0$. Therefore, given data $x_1, x_2, \dots, x_n \in R^p$, to explore the relation between features, we can estimate the inverse covariance matrix by computing the MLE:

$$\max_S \left\{ \log \det(S) - \text{Tr}(\hat{\Sigma}S) \right\}$$

If the number of features p is large, certain regularization is needed to control the number of edges. Banerjee et al. ([Banerjee et al., 2008]) propose the following optimization problem to recover the sparse structure in a gaussian graphical model

$$\max_S \left\{ \log \det S - \text{Tr}(\hat{\Sigma}S) - \lambda \|S\|_1 \right\}$$

where $\|S\|_1 = \sum_{i=1} \sum_{j=1} |s_{ij}|$. To solve for S , Banerjee et al. ([Banerjee et al., 2008]) propose a block coordinate ascent method (updating one row and one column of S at one time). Their approach is exact but is time consuming. Meinshausen and Bühlmann ([Meinshausen and Bühlmann, 2006]) uses an approximation approach that is substantially faster. In our study, we adopt the fast and accurate graphical LASSO procedure (R package `glasso`, [Friedman et al., 2011]). Figure 5 shows the word usage network by fitting a sparse graphical model for $\lambda = 0.01$. Different values of the regularization parameters are tuned and more such plots can be found in Appendix A.

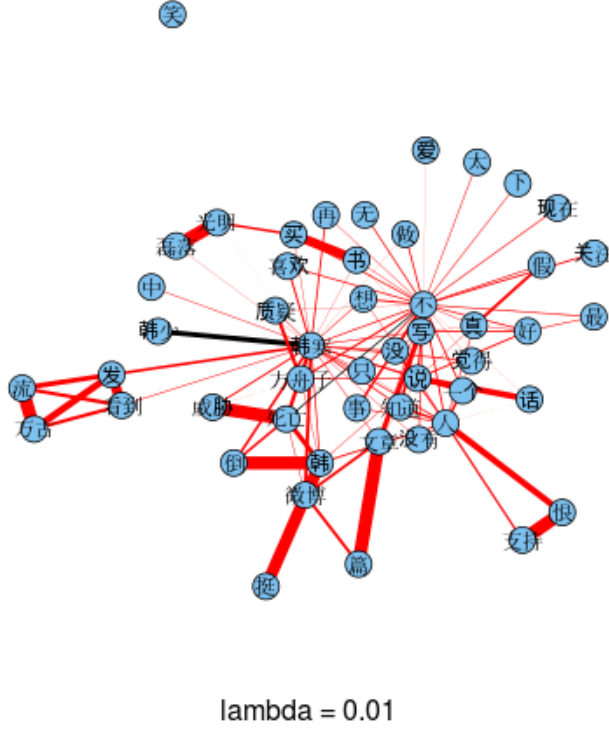


Figure 5: The network plot of the estimated inverse covariance matrix S , with $\lambda = 0.01$. As discussed, an edge (i.e. $S_{ij} \neq 0$) represents conditional dependency between two nodes given all the other nodes. Red edges indicate positive correlated occurrence $S_{ij} < 0$ (Given all other words, word i is more likely to occur if word j is observed) and black edges indicate negative correlated occurrence $S_{ij} > 0$. Edge width is proportional to $|S_{ij}|$, representing the strength of the tie. Again, due to space limit, only the top 50 most frequent words are shown.

We can make a comparison between the concurrence network (Figure 4) and sparse graphical model (Figure 5): the concurrence network is heavily influenced by usage frequency of words. For example, “不” (the negation word) and “韩寒”(Han Han) are strongly connected in the concurrence network, but this might not imply that there is a nontrivial relation between these two words. Sparse graphical models, on the other hand, give more interpretable results. For example, “不”(the negation word) and “韩寒”(Han Han) are not longer heavily connected. In addition, some interesting word combinations are revealed: the word “光明” (light) and “磊落” (upright) are words that constitute the name of the Han Han’s new book Light and Upright as mentioned in Section 2; and the clique form by “方舟子” (Fang Zhouzi), “怀疑” (suspect) and “韩寒” (Han Han) seems to reveal the fact that Fang suspects Han Han has a team of ghost writers. Other obvious relation revealed include “死亡” (death) and “威胁” (threat), “买” (buy) and “书” (book), “写” (write) and “文章” (article), and so forth.

It is also of interest to observe how the word usage network structure change as we change the regularization parameter λ . From Figure 5- 11 in Appendix A one can see that widths of the edges weaken and the number of edges decreases. Only strong conditional dependency between two words

can survive for the largest regularization parameter we try: $\lambda = 0.03$, e.g. word pairs like “死亡” (death) and “威胁” (threat), and “方舟子” (Fang Zhouzi) and “韩寒” (Han Han).

5 Classification

We are also interested in classifying the posts into different categories. Let $x_i \in R^p$ be the i -th row of the frequency matrix X (of the test data) and y_i the corresponding category. For simplicity, let’s assume that $y_i \in \{-1, +1\}$ is binary, where the “+1” can be used to label the following four categories: 1) positive opinion towards Han Han; 2) negative opinion towards Han Han; 3) neutral or unidentifiable opinion; 4) spam. Table 2 gives counts of posts of each of the above four categories in the training data set. We use “-1” to label the complement of the individual category (e.g., if “+1” means positive, “-1” would mean anything but positive. Note that the complement of positive is not negative, but rather, the union of negative, neutral and spam). For each of the above four cases, we apply LASSO and l_1 -norm support vector machine to classify the data points into “-1” and “+1.”

Table 2: Counts of posts in each of the four categories among the 3000 training posts.

Positive	Negative	Neutral	Spam
1097	308	1323	272

5.1 Sentiment Score Assignment

Dictionary-based sentiment score can provide us some intuitive understanding of the sentiment polarity of the posts. The dictionaries are obtained from HowNet [Dong and Dong, 2012]. HowNet is an online extralinguistic common-sense knowledge system for the computation of meaning in human language technology.

Each post is examined and the numbers of positive and negative words are recorded. A positive word contributes +1 to the sentiment score, whereas a negative word contributes -1. If there is a negation word among the three words before the positive/negative word, their combination will be treated as an entity and their updated contribution is -1 times the original contribution. The six negation words used are 不, 不是, 没有, 没, 无, and 别. The sentiment score of the post is the sum of all the contributions.

Also topic-related positive/negative words are added to the dictionaries. For instance, the users who refer Han Han as 韩少 (韩: Han Han’s surname, 少: abbreviation of 少爷, which means “young master of the house”) clearly have positive feelings about him.

This sentiment score can be used for classification purposes. A positive or negative score of a post indicates the corresponding category, whereas a post with zero score is considered as neutral. But spam category cannot be identified.

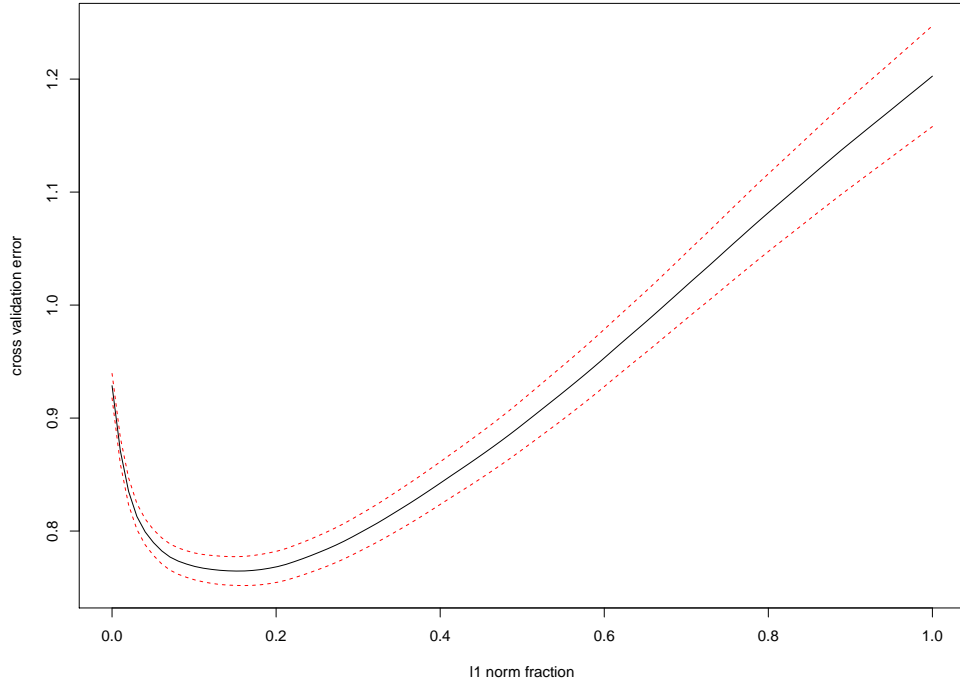


Figure 6: Cross-validation error plot for the positive v.s. nonpositive classification.

5.2 Sparse Regression with LASSO

LASSO ([Tibshirani, 1996]) is a sparse regression method which adds a l_1 -norm penalty to the linear least squares objective to promote sparsity in the regression coefficients:

$$\hat{\beta}(\lambda) = \arg \min_{\beta} \frac{1}{2} \|y - (\beta_0 + X\beta)\|_2^2 + \lambda \|\beta\|_1$$

In our study, we regress the class label vector y onto the word frequency matrix X , yielding the intercept $\hat{\beta}_0$ and the sparse regression vector $\hat{\beta}(\lambda)$. The classifier can be determined to be $f(x) = \mathbf{sign}(\hat{\beta}_0 + \hat{\beta}^T x) \in \{-1, +1\}$, where the resulting coefficient regression coefficient $\hat{\beta}$ has the following explanation: for each feature or word j , given all other feature or word variables fixed, the increase of the j -th word frequency by one leads to an increase in the regression function $\beta_0 + \beta^T x$ by an amount of β_j (if β_j turns out to be positive, this means the chance of classifying the data point into the $+1$ category is increased).

To look at which words were most relevant to each category we modeled (positive, negative, neutral, and spam), we looked at three sets of 20 words based on, respectively, the highest absolute β values, highest positive coefficient values and most negative coefficient values. The top 20 words based on the top 20 highest absolute β values tell us in general which words were most relevant to predicting that particular category. The top 20 words based on the top 20 highest positive β values tell us which words typically were common in posts that fell into the category modeled while the top 20 words based on the top 20 largest (in terms of magnitude) negative β values tell us which words typically were not in posts that fell into the category modeled but were rather more common in the categories outside of the one we were modeling.

Since the estimated β depends on the regularization parameter, we are left with the issue of choosing the “best” λ . A commonly used approach is to do a grid search for λ : for each value of λ , do a 10-fold cross validation; then choose the λ that yields the smallest cross validation testing sample error. For this method, we used the approach least-angle regression (LARS) by [Efron et al., 2004] to do the model selection. Their R package `lars` efficiently fits an entire lasso sequence with the least squares loss function ([Hastie and Efron, 2012]). Figure 6 shows the cross-validation error plot for the positive v.s. nonpositive classification and the other plots are in Appendix B.1.

The lists of top 5 most relevant words for positive and negative classification are in Tables 3–4. The complete list of top 20 most relevant words for each classification is in Appendix B.2 (Tables 9–12).

For the positive responses, the top 5 most relevant words (based on taking the absolute values of the betas) returned by our model are shown in Table 3. As can be seen, the top 5 most relevant words for the positive betas and the absolute value of betas were the same. The top 5 relevant words for the two categories have an assortment of fairly neutral or positive words. For example, words such as “mature,” “support,” and “keep up the good work” are very positive in nature and would certainly indicate a positive reaction to 韓寒. The top 5 most relevant words for the negative betas were not as easily classified since the phrases need to be placed in appropriate contexts to be able to tell the sentiment. We’d expect these words to indicate a dislike of Han han. An example of how “love” could indicate a dislike of Han Han is the fact that many opponents of Han Han mocked and teased him about his relationship with another author; they’d insinuate that the two authors were lovers.

For the negative responses, the top 5 most relevant words (based on taking the absolute values of the betas) returned by our model are shown in Table 4. As can be seen, the top 5 most relevant words for the positive betas and the absolute value of betas are the same again. These words clearly hold a negative sentiment. Words like “disgusting,” “hate,” and “liar” easily suggest that the post exudes a negative sentiment towards the author. The top 5 most relevant words for the negative betas are again a bit harder to classify though looking at the full list of the betas indicates a clearer picture of the fact that negative betas typically correspond to positive words. The word corresponding to the highest negative coefficient is “support” which easily indicates a support of Han han. What is interesting to note with these betas is the fact that these betas are all close to zero except for the first phrase.

Table 3: LASSO word images for the positive v.s. nonpositive classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
加油 (keep going)	0.820	加油 (keep going)	0.820	样子 (manner)	-0.396
韩少 (Master Han)	0.644	韩少 (Master Han)	0.644	恋 (love)	-0.344
成熟 (mature)	0.546	成熟 (mature)	0.546	发表 (announce)	-0.336
顶 (support)	0.533	顶 (support)	0.533	道理 (rational)	-0.336
宽容 (tolerant)	0.518	宽容 (tolerant)	0.518	利益 (benefit)	-0.335

Table 4: LASSO word images for the negative v.s. nonnegative classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
讨厌 (hate)	0.481	讨厌 (hate)	0.481	支持 (support)	-0.008
无耻 (shameless)	0.412	无耻 (shameless)	0.412	-	-
恶心 (disgusting)	0.395	恶心 (disgusting)	0.395	-	-
骗子 (liar)	0.380	骗子 (liar)	0.380	-	-
扁 (beat up)	0.353	扁 (beat up)	0.353	-	-

5.3 l_1 -Norm Support Vector Machine

The support vector machine (SVM) is another commonly used machine learning method to classify data points into two categories. Consider again the linear decision function $f(x) = \beta_0 + \beta^T x$ and the sign classifier $\text{class}(x) = \mathbf{sign}(f(x))$. The SVM minimizes the training misclassification rate and the margin of the decision boundary. Following the notations in [Hastie et al., 2004]:

$$\min_{\beta_0, \beta} \sum_{i=1}^n (1 - y_i(\beta_0 + \beta^T x_i))_+ + \frac{\lambda}{2} \|\beta\|_2^2,$$

where $z_+ = \max(0, z)$ (the function $h(z) = (1 - z)_+$ is also known as the hinge loss function). Similarly, the sparse version of SVM simply replaces the l_2 -norm by l_1 -norm (see, e.g. [Zhu et al., 2004]):

$$\min_{\beta_0, \beta} \left\{ \sum_{i=1}^n (1 - y_i(\beta_0 + \beta^T x_i))_+ + \lambda \|\beta\|_1 \right\}.$$

We repeat the same data analysis for the text data using l_1 -norm SVM as we did for the LASSO classification. To fit the sparse SVM, we use the matlab package `lpsvm` by [Fung and Mangasarian, 2004]. Again, 10-fold cross validations are performed in order to select the regularization parameter λ . Again the model seems to produce plausible results. Also, the top coefficients obtained by l_1 -norm SVM seems to be consistent with those obtained by fitting a LASSO.

The lists of top 5 most relevant words for positive and negative classification are in Tables 5–6. The complete list of top 20 most relevant words for each classification is in Appendix C.1 (Tables 13–16).

Table 5: l_1 -norm SVM word images for the positive v.s. nonpositive classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
加油 (keep going)	2.340	加油 (keep going)	2.340	铁证 (clear evidence)	2.305
铁证 (clear evidence)	2.305	家人 (family)	2.269	接受 (accept)	2.061
家人 (family)	2.269	韩少 (Master Han)	1.969	媒体 (media)	1.907
接受 (accept)	2.061	成熟 (mature)	1.806	默默 (quietly)	1.883
韩少 (Master Han)	1.969	顶 (support)	1.803	四娘 (GUO Jingming)	1.762

Table 6: l_1 -norm SVM word images for the negative v.s. nonnegative classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
扁 (beat up)	1.777	扁 (beat up)	1.777	脑子 (mind)	1.447
苦肉计 (the ruse of self-injury to win somebody's confidence)	1.708	苦肉计 (the ruse of self-injury to win somebody's confidence)	1.708	彻底 (completely)	1.290
恶心 (disgusting)	1.527	恶心 (disgusting)	1.527	送给 (give)	1.221
脑子 (asdf)	1.447	骗子 (liar)	1.301	感觉 (feel)	1.109
骗子 (liar)	1.301	公开 (open)	1.220	热点 (hot interest)	1.101

6 Test Set Misclassification Rates

To compare the classification performance, we use the best fitted models discussed above to classify the test set of 500 posts. Recall that the best models are selected to minimize the 10-fold cross validation error. Table 7 shows the frequency count of each category in the 500 posts and Table 8 summarizes the test results.

Table 7: Counts of posts in each of the four categories among the 500 test posts.

Positive	Negative	Neutral	Spam
172	36	287	5

Table 8: Misclassification rates of the three methods for each of the four categories.

	Positive	Negative	Neutral	Spam
Sentiment score assignment	78.6%	97.2%	78.8%	-
LASSO	22.6%	7%	33.6%	1%
l_1 -norm SVM	22.6%	8%	33.4%	1%

It can be seen that the two standard machine learning methods produce very similar results. Also notice the low misclassification rates in the negative and spam categories, which might be due to the fact that we do not frequently see posts that belong to those two categories in our test set.

It is also surprising to see how the sentiment score assignment method fails miserably in classifying the posts. As discussed in Section 5.1, this naïve approach is dictionary-based and only considers the individual words. Clearly an entity is not merely the sum of its parts. It is crucial to take context into account in sentiment analysis. For instance, a post containing many positive words may be praising Fang Zhouzi and hence against Han Han. Moverover, zero score has two scenarios: the post contains the same number of positive and negative words; there is no sentimental words presented. Even though these two cases can be distinguished, there is still little information to make reasonable inference. Since the dictionary itself is limited, the scores may not represent the whole picture.

7 Conclusion

In this project, we developed a framework for studying public opinions using Sina Weibo as a corpus for a given topic. Posts are collected from Weibo and then processed taking the characteristics of both the Chinese language and Weibo posts into consideration. Also the processing is topic-dependent since there are certain proper nouns related to a given topic. Word usage relation has been examined. Due to the sparsity of the word frequency matrix, both sparse principle component analysis (SPCA) and sparse graphical models are employed in our feature analysis. Two standard machine learning methods, namely sparse regression with LASSO and l_1 -norm SVM, have been applied to the training set of 3,000 posts and then the best fitted models are tested on a test set of 500 posts. These two methods have similar performance on our data set. Due to limitation in time we have not extended our study to a larger corpus. However, the sparse machine learning tools

we have used in this project produce human-interpretable results and reasonable misclassification rates.

Many future efforts remain to be done. One possible direction is to enlarge the scale of our study, in both time (i.e. extend the time period we do the sampling) and number of posts. In terms of sampling techniques, we would like to study random samples by sampling from large network as discussed in Section 2 after spams, OOV, and reposting can be handled properly. In terms of methodology, we would like to compare our LASSO and l_1 -norm SVM methods with the popular maximum entropy approach ([Lee and Renganathan, 2011, Nigam et al., 1999]). Also, as discussed in Section 2, one topic of interest is to track repostings using graphical models. Identification of OOV needs more statistical work as well. Clearly sentiment classifier is topic-dependent and there is no reason why we should restrict our analysis on just a few famous public figures. In the future, we can collect data on more and broader topics, preferably in other domains, to test our framework.

References

- [Banerjee et al., 2008] Banerjee, O., El Ghaoui, L., and d’Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516.
- [Boyd et al., 2004] Boyd, S., Diaconis, P., and Xiao, L. (2004). Fastest mixing markov chain on a graph. *SIAM review*, pages 667–689.
- [Cao, 2012] Cao, B. (2012). Sina’s Weibo outlook buoys Internet stock gains: China overnight. <http://www.bloomberg.com/news/2012-02-28/sina-s-weibo-outlook-buoys-internet-stock-gains-in-n-y-china-overnight.html>.
- [Dong and Dong, 2012] Dong, Z. and Dong, Q. (2012). 知网 HowNet. <http://www.keenage.com/>.
- [Efron et al., 2004] Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *The Annals of statistics*, 32(2):407–499.
- [El Ghaoui et al., 2011] El Ghaoui, L., Li, G.-C., Duong, V.-A., Pham, V., Srivastava, A., and Bhaduri, K. (2011). Sparse machine learning methods for understanding large text corpora. In *Proc. Conference on Intelligent Data Understanding*. Accepted for publication, July 2011.
- [Friedman et al., 2011] Friedman, J., Hastie, T., and Tibshirani, R. (2011). *glasso: Graphical lasso-estimation of Gaussian graphical models*. R package version 1.7.
- [Fung and Mangasarian, 2004] Fung, G. and Mangasarian, O. (2004). A feature selection newton method for support vector machine classification. *Computational optimization and applications*, 28(2):185–202.
- [Gawalt et al., 2010] Gawalt, B., Jia, J., Miratrix, L., El Ghaoui, L., Yu, B., and Clavier, S. (2010). Discovering word associations in news media via feature selection and sparse classification. In *Proceedings of the international conference on Multimedia information retrieval*, pages 211–220. ACM.
- [Hastie and Efron, 2012] Hastie, T. and Efron, B. (2012). *lars: Least Angle Regression, Lasso and Forward Stagewise*. R package version 1.1.
- [Hastie et al., 2004] Hastie, T., Rosset, S., Tibshirani, R., and Zhu, J. (2004). The entire regularization path for the support vector machine. *The Journal of Machine Learning Research*, 5:1391–1415.
- [Institute of Computing Technology, Chinese Academy of Sciences, 2011] Institute of Computing Technology, Chinese Academy of Sciences (2011). 汉语词法分析系统 ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System). <http://ictclas.org/>.
- [Lee and Renganathan, 2011] Lee, H. and Renganathan, H. (2011). Chinese sentiment analysis using maximum entropy. *Sentiment Analysis where AI meets Psychology (SAAIP)*, page 89.
- [Lee, 2012] Lee, M. (2012). China shuts down microblog accounts amid rumors of coup attempts. http://www.washingtonpost.com/business/china-shuts-down-microblog-accounts-amid-rumors-of-coup-attempts/2012/04/28/gIQAxYTJoT_story.html.

- [Leskovec and Faloutsos, 2006] Leskovec, J. and Faloutsos, C. (2006). Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 631–636. ACM.
- [Meinshausen and Bühlmann, 2006] Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462.
- [Nigam et al., 1999] Nigam, K., Lafferty, J., and McCallum, A. (1999). Using maximum entropy for text classification. In *IJCAI-99 workshop on machine learning for information filtering*, volume 1, pages 61–67.
- [Tan and Zhang, 2008] Tan, S. and Zhang, J. (2008). An empirical study of sentiment analysis for chinese documents. *Expert Systems with Applications*, 34(4):2622–2629.
- [Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.
- [Wainwright and Jordan, 2008] Wainwright, M. and Jordan, M. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1-2):1–305.
- [Wang et al., 2011] Wang, T., Chen, Y., Zhang, Z., Xu, T., Jin, L., Hui, P., Deng, B., and Li, X. (2011). Understanding graph sampling algorithms for social network analysis. In *Distributed Computing Systems Workshops (ICDCSW), 2011 31st International Conference on*, pages 123–128. IEEE.
- [Wong et al., 2009] Wong, K., Li, W., Xu, R., and Zhang, Z. (2009). Introduction to chinese natural language processing. *Synthesis Lectures on Human Language Technologies*, 2(1):1–148.
- [Zhu et al., 2004] Zhu, J., Rosset, S., Hastie, T., and Tibshirani, R. (2004). 1-norm support vector machines. *Advances in neural information processing systems*, 16(1):49–56.
- [Zou et al., 2006] Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286.

A Sparse Graphical Model Plots

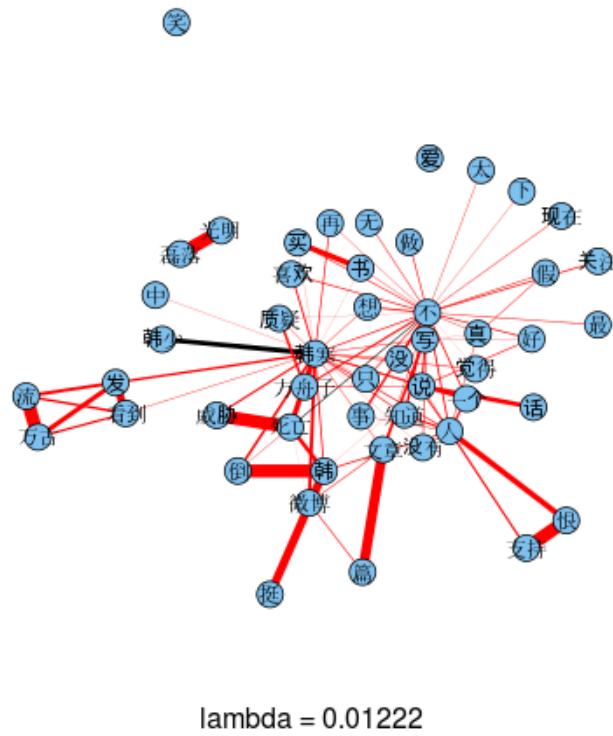


Figure 7: The network plot of the estimated inverse covariance matrix S , with $\lambda = 0.01222$.

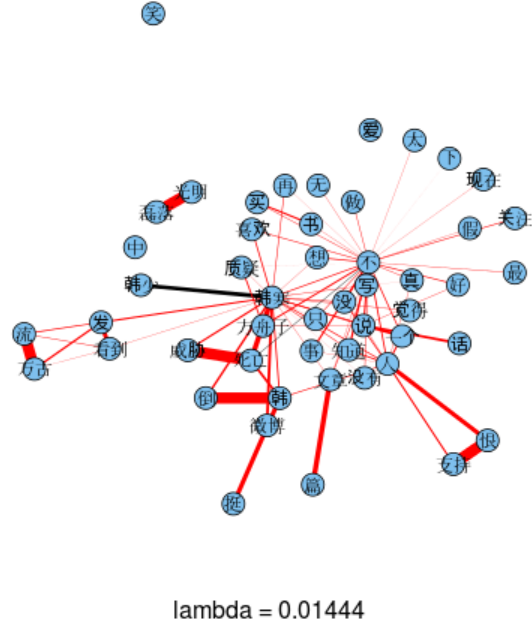


Figure 8: The network plot of the estimated inverse covariance matrix S , with $\lambda = 0.01444$.

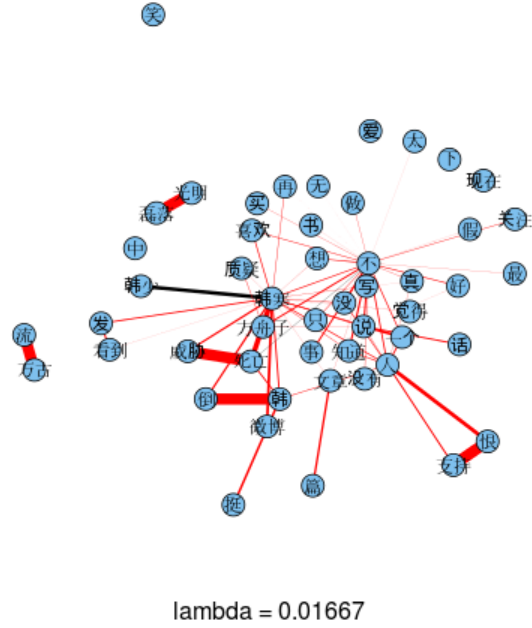


Figure 9: The network plot of the estimated inverse covariance matrix S , with $\lambda = 0.01667$.

B LASSO Results

B.1 Cross-Validation Error Plot

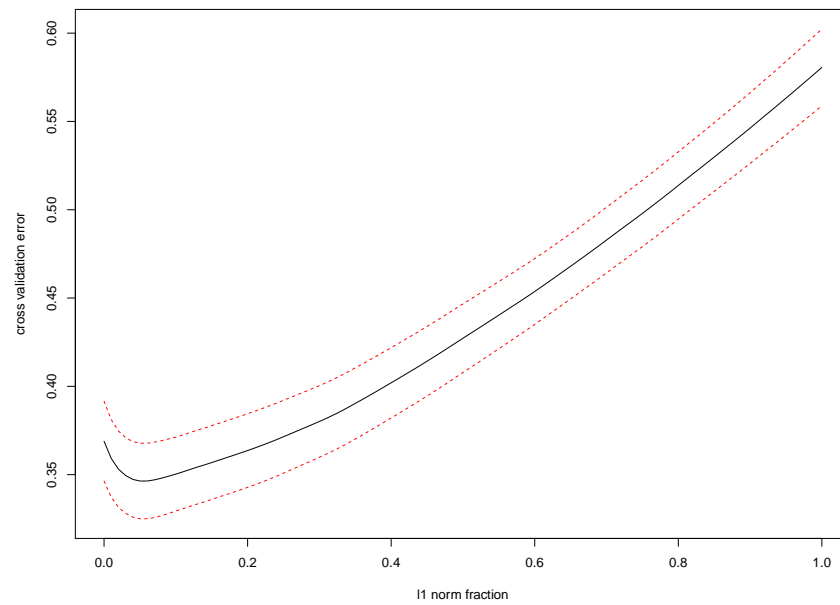


Figure 12: Cross-validation error plot for the negative v.s. nonnegative classification.

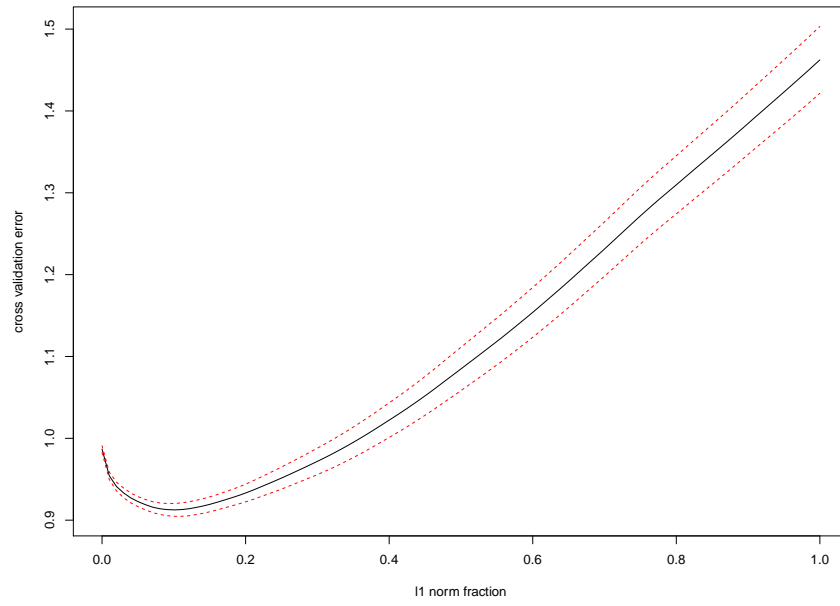


Figure 13: Cross-validation error plot for the neutral v.s. nonneutral classification.

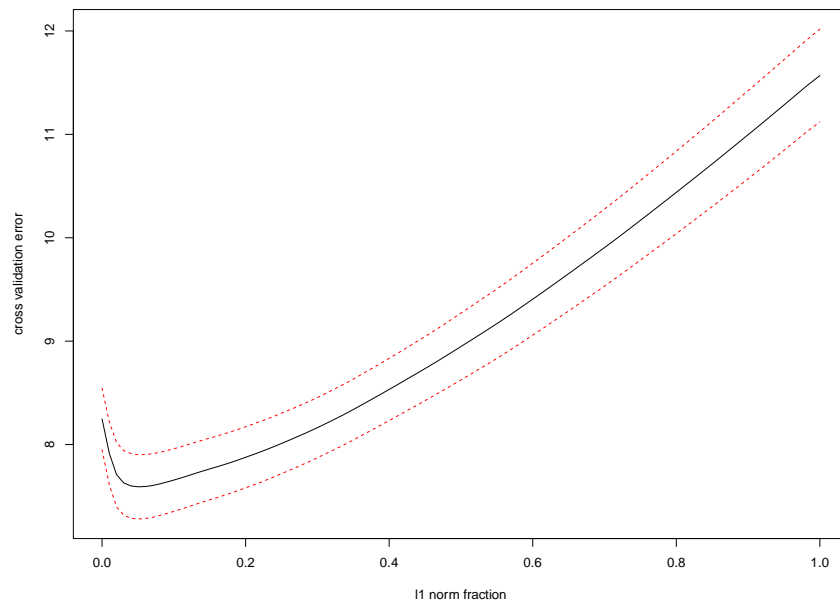


Figure 14: Cross-validation error plot for the spam v.s. nonspam classification.

B.2 LASSO Word Images

Table 9: LASSO word images for the positive v.s. nonpositive classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
加油 (keep going)	0.820	加油 (keep going)	0.820	样子 (manner)	-0.396
韩少 (Master Han)	0.644	韩少 (Master Han)	0.644	恋 (love)	-0.344
成熟 (mature)	0.546	成熟 (mature)	0.546	发表 (announce)	-0.336
顶 (support)	0.533	顶 (support)	0.533	道理 (rational)	-0.336
宽容 (tolerant)	0.518	宽容 (tolerant)	0.518	利益 (benefit)	-0.335
支持	0.477	支持	0.477	称	-0.323
家人	0.467	家人	0.467	遭受	-0.323
样子	0.396	尤其	0.395	媒体	-0.319
尤其	0.395	欣赏	0.383	翻	-0.314
欣赏	0.383	感动	0.381	铁证	-0.289
感动	0.381	影响力	0.370	骗子	-0.248
影响力	0.370	新书	0.327	上调	-0.248
恋	0.344	铁	0.316	投票	-0.234
发表	0.336	不错	0.309	女	-0.230
道理	0.336	终于	0.274	四娘	-0.226
利益	0.335	每个	0.274	关系	-0.215
新书	0.327	咬	0.261	广告	-0.210
称	0.323	文字	0.260	接受	-0.208
遭受	0.323	蛋	0.244	网	-0.204
媒体	0.319	纠缠	0.244	底	-0.196

Table 10: LASSO word images for the negative v.s. nonnegative classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
讨厌 (hate)	0.481	讨厌 (hate)	0.481	支持 (support)	-0.008
无耻 (shameless)	0.412	无耻 (shameless)	0.412	不 (no)	0.000
恶心 (disgusting)	0.395	恶心 (disgusting)	0.395	人 (people/person)	0.000
骗子 (liar)	0.380	骗子 (liar)	0.380	说 (say)	0.000
扁 (beat up)	0.353	扁 (beat up)	0.353	方舟子 (FangZhouZi)	0.000
装	0.321	装	0.321	韩少	0.000
选项	0.292	选项	0.292	真	0.000
苦肉计	0.290	苦肉计	0.290	好	0.000
利益	0.283	利益	0.283	没	0.000
全	0.261	全	0.261	一个	0.000
国家	0.247	国家	0.247	微博	0.000
智商	0.216	智商	0.216	写	0.000
告	0.198	告	0.198	喜欢	0.000
虚伪	0.192	虚伪	0.192	想	0.000
演	0.191	演	0.191	威胁	0.000
语	0.186	语	0.186	只	0.000
烦	0.178	烦	0.178	太	0.000
掉	0.142	掉	0.142	事	0.000
下去	0.141	下去	0.141	没有	0.000
公开	0.141	公开	0.141	看到	0.000

Table 11: LASSO word images for the neutral v.s. nonneutral classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
上调 (increase)	0.586	上调 (increase)	0.586	加油 (keep going)	-0.491
道理 (rational)	0.566	道理 (rational)	0.566	韩少 (Master Han)	-0.358
账号 (account)	0.534	账号 (account)	0.534	苦肉计 (the ruse of self-injury to win somebody's confidence)	-0.327
加油 (keep going)	0.491	铁证 (clear evidence)	0.459	支持 (support)	-0.290
铁证 (clear evidence)	0.459	称 (refer)	0.453	善良 (kind)	-0.268
称	0.453	想起	0.353	成熟	-0.263
韩少	0.358	杀	0.331	终于	-0.239
想起	0.353	最终	0.329	家人	-0.233
杀	0.331	意思	0.323	同意	-0.228
最终	0.329	遭遇	0.319	越来越	-0.220
苦肉计	0.327	金	0.308	欢乐	-0.217
意思	0.323	片	0.287	崇拜	-0.213
遭遇	0.319	应	0.278	讨厌	-0.202
金	0.308	变成	0.265	顶	-0.197
支持	0.290	有点	0.262	代笔	-0.194
片	0.287	之间	0.254	跳	-0.191
应	0.278	右边	0.250	真善美	-0.186
善良	0.268	民主	0.238	真正	-0.185
变成	0.265	郭敬明	0.237	欣赏	-0.181
成熟	0.263	久	0.226	无耻	-0.180

Table 12: LASSO word images for the spam v.s. nonspam classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
查看 (examine)	3.777	查看 (examine)	3.777	韩少 (Master Hanhan)	-1.033
抽 (win)	1.998	抽 (win)	1.998	韩寒 (Master Hanhan)	-0.716
每天 (everyday)	1.251	每天 (everyday)	1.251	别 (don't)	-0.232
往往 (often)	1.208	往往 (often)	1.208	支持 (support)	-0.217
外 (outside)	1.043	外 (outside)	1.043	这种 (this kind)	-0.202
韩少	1.033	征集	0.948	感	-0.196
征集	0.948	容	0.849	韩	-0.191
容	0.849	风	0.649	没有	-0.179
韩寒	0.716	票子	0.570	上调	-0.174
风	0.649	考	0.540	方舟子	-0.160
票子	0.570	主	0.438	光明	-0.141
考	0.540	性	0.430	一定	-0.137
主	0.438	总是	0.416	照妖镜	-0.132
性	0.430	儿	0.416	写	-0.132
总是	0.416	结论	0.405	觉得	-0.119
儿	0.416	后面	0.397	甚	-0.110
结论	0.405	法律	0.388	韩	-0.092
后面	0.397	机会	0.376	真相	-0.084
法律	0.388	公知	0.357	挺	-0.072
机会	0.376	中	0.357	不	-0.068

C l_1 -Norm Support Vector Machine Results

C.1 l_1 -Norm SVM Word Images

Table 13: l_1 -norm SVM word images for the positive v.s. nonpositive classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
加油 (keep going)	2.340	加油 (keep going)	2.340	铁证 (clear evidence)	2.305
铁证 (clear evidence)	2.305	家人 (family)	2.269	接受 (accept)	2.061
家人 (family)	2.269	韩少 (Master Han)	1.969	媒体 (media)	1.907
接受 (accept)	2.061	成熟 (mature)	1.806	默默 (quietly)	1.883
韩少 (Master Han)	1.969	顶 (support)	1.803	四娘 (GUO Jingming)	1.762
媒体	1.907	宽容	1.764	骗子	1.524
默默	1.883	人士	1.758	想起	1.519
成熟	1.806	支持	1.593	恋	1.491
顶	1.803	欢乐	1.315	韩寒和	1.476
宽容	1.764	影响力	1.237	关	1.415
四娘	1.762	诛	1.175	变成	1.411
人士	1.758	欣赏	1.138	圈	1.243
支持	1.593	幸福	1.126	曾经	1.238
骗子	1.524	纠缠	1.030	战	1.225
想起	1.519	代笔	1.025	网	1.219
恋	1.491	喜欢	0.943	发表	1.190
韩寒和	1.476	蛋	0.923	闲	1.190
关	1.415	明	0.919	小四	1.125
变成	1.411	终于	0.914	底	1.050
欢乐	1.315	咬	0.884	套	1.044

Table 14: l_1 -norm SVM word images for the negative v.s. nonnegative classification

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
扁 (beat up)	1.777	扁 (beat up)	1.777	脑子 (mind)	1.447
苦肉计 (the ruse of self-injury to win somebody's confidence)	1.708	苦肉计 (the ruse of self-injury to win somebody's confidence)	1.708	彻底 (completely)	1.290
恶心 (disgusting)	1.527	恶心 (disgusting)	1.527	送给 (give)	1.221
脑子 (asdf)	1.447	骗子 (liar)	1.301	感觉 (feel)	1.109
骗子 (liar)	1.301	公开 (open)	1.220	热点 (hot interest)	1.101
彻底	1.290	全	1.154	恶	1.077
送给	1.221	国家	1.149	青春	1.013
公开	1.220	讨厌	1.053	少	0.949
全	1.154	网	1.034	算	0.940
国家	1.149	烦	0.889	清楚	0.921
感觉	1.109	虚伪	0.843	愿意	0.920
热点	1.101	装	0.812	新书	0.868
恶	1.077	告	0.712	写作	0.840
讨厌	1.053	讨论	0.674	争论	0.827
网	1.034	难	0.636	地方	0.796
青春	1.013	时代	0.633	铁	0.789
少	0.949	天才	0.629	言	0.737
算	0.940	样子	0.626	子	0.727
清楚	0.921	选项	0.596	来自	0.705
愿意	0.920	喝	0.559	精彩	0.685

Table 15: l_1 -norm SVM word images for the neutral v.s. nonneutral classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
加油 (keep going)	2.233	铁证 (clear evidence)	2.054	加油 (keep going)	2.233
铁证 (clear evidence)	2.054	片 (piece)	1.896	成熟 (mature)	1.756
片 (piece)	1.896	至今 (so far)	1.884	同意 (agree)	1.725
至今 (so far)	1.884	战 (fight)	1.845	水 (water)	1.661
战 (fight)	1.845	意思 (meaning)	1.824	昨天 (yesterday)	1.611
意思	1.824	遭遇	1.729	韩少	1.604
成熟	1.756	儿子	1.701	路	1.568
遭遇	1.729	关系	1.648	国家	1.553
同意	1.725	生日	1.646	讨厌	1.533
儿子	1.701	道德	1.626	咬	1.485
水	1.661	称	1.610	人士	1.438
关系	1.648	杀	1.578	掉	1.322
生日	1.646	接受	1.567	小丑	1.311
道德	1.626	狂	1.508	偏执	1.282
昨天	1.611	账号	1.485	抽	1.257
称	1.610	理解	1.444	语	1.254
韩少	1.604	道理	1.341	声	1.252
杀	1.578	不幸	1.332	家人	1.235
路	1.568	加	1.293	一直	1.213
接受	1.567	上调	1.267	自由	1.198

Table 16: l_1 -norm SVM word images for the spam v.s. nonspam classification.

Word	Absolute Coef.	Word	Positive Coef.	Word	Negative Coef.
票子 (ticket)	1.000	票子 (ticket)	1.000	书 (book)	0.000
书 (book)	0.000	每天 (everyday)	0.000	围观 (surround to watch)	0.000
围观 (surround to watch)	0.000	抽 (win)	0.000	写 (write)	0.000
写 (write)	0.000	性 (sex)	0.000	骂 (curse)	0.000
每天 (everyday)	0.000	网 (Internet)	0.000	粉丝 (fans)	0.000
抽	0.000	分享	0.000	寫	0.000
骂	0.000	容	0.000	犯	0.000
性	0.000	公知	0.000	不错	0.000
粉丝	0.000	总	0.000	转	0.000
网	0.000	小	0.000	愤	0.000
分享	0.000	图	0.000	韩寒	0.000
寫	0.000	说	0.000	磊落	0.000
容	0.000	真	0.000	错	0.000
犯	0.000	支持	0.000	韩寒和	0.000
不错	0.000	韩	0.000	韩少	0.000
转	0.000	喜欢	0.000	方舟子	0.000
公知	0.000	想	0.000	觉得	0.000
愤	0.000	威胁	0.000	已经	0.000
韩寒	0.000	事	0.000	娱乐	0.000
总	0.000	看到	0.000	新	0.000