Predicting Censorship on Weibo

CSE 190: Assignment 2

Brian Tsay
A53029582
brtsay@ucsd.edu

John Kuk
A53032819
jskuk@ucsd.edu

ABSTRACT

Put abstract here.

CCS Concepts

• **Applied computing** → *Law, social and behavioral sciences;*

1. DATASET

The data used for this assignment is taken from the Weiboscope data collection and visualization project developed by the research team at the Journalism and Media Studies Centre, The University of Hong Kong [5]. The dataset consists of weibos (roughly the Chinese equivalent of tweets) collected in the year 2012. The authors created the dataset by first compiling a list of Weibo users with 1,000 or more followers and then getting their timelines, friends, and followers. Within the entire dataset, there are 226,841,122 tweets from 14,387,628 unique users. Of these tweets, 86,083 (about 0.03%) are censored.

Figure 1 shows the word clouds for the noncensored tweets (left) and for the censored tweets (right). Some terms are found often in both sets, namely 轉發微博 (retweet), 哈哈 (haha), 中国人 (Chinese people), and 越来越 (more and more).

However, censored tweets tend to be much more political. 钓鱼岛 (Diaoyu Islands) refer to the disputed island chains between China and Japan. 斯巴达 (Sparta) is actually a play on the fact that Sparta in Chinese (sibada) is nearly a homophone for 十八大 (shibada), the 18th National Party Congress where the previous president Hu Jintao stepped down and the current president – Xi Jinping – took his place.[1] There tend to be more people names in the censored tweets, such as 毛泽东 (Mao Zedong, former chairman of the Chinese Communist Party), 薄熙来 (Bo Xilai, disgraced Chongqing party chief), 王立军 (Wang Lijun, vice-mayor of Chongqing under Bo), and others. There are also references to the Chinese government, such as 领导人 (leadership) and 共产党 (Communist Party), and the masses (老百姓).

In comparison, the noncensored tweets appear much more benign. One of the most common terms is 奥特曼, which refers to the fictional Japanese superhero Ultraman, who is apparently quite popular among a certain generation in China. Many popular terms are simply astrology zodiac signs e.g. Leo (狮子座), Sagittarius (射手座), etc. The astrology most likely comes from a bot or person that tweets horoscopes every day.[2] The popularity of the term Ultraman can perhaps be attributed to the creation of a noodle-robot that looked like Ultraman in 2012. The conclusion would be that we should look for more overtly political words to classify censored tweets.

For the assignment, we kept all the censored tweets but kept only a small subsample (0.1%) of the noncensored tweets. Noncensored tweets were randomly sampled from the entire dataset such that the number of noncensored tweets would be roughly twice that as the number of censored tweets within our data.[3] Whereas 0.03% of tweets are censored in the entire dataset, 34% of tweets are censored in our small subsample. The training set was constructed such that half of the tweets would be censored while the other half would be noncensored. The validation and test sets were deliberately constructed to look like one another.

Table 1 shows some descriptive statistics for our dataset. It appears that more than half of the tweets in our subsample are retweets.[4] We also see that, on average, each user is producing 2.5 tweets in our dataset. This implies that many tweets are correlated with one another. A retweet is very likely related to whatever the original tweet was, and individual users will probably tweet similar things over time. These aspects of the data will be incorporated into our features as described in Section 2.

Figure 2 shows the distribution of tweets over time in our small subsample. While the number of noncensored tweets is somewhat constant over time, the number of censored tweets clearly vary greatly over time. There are few censored tweets in the beginning of the year, with a spike coming in mid-2012 and a huge increase near the end of 2012. The spike at

[1] Chinese internet users often rely on homophone tricks such as these to evade automatic keyword censoring. In this case, we can see that it was not 100% effective.

[2] Astrology is actually fairly popular in China.

[3] There was no particular reason that this ratio was chosen.

[4] Note that this is a function of how the original dataset was constructed. This is not an artifact of our random sampling strategy.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

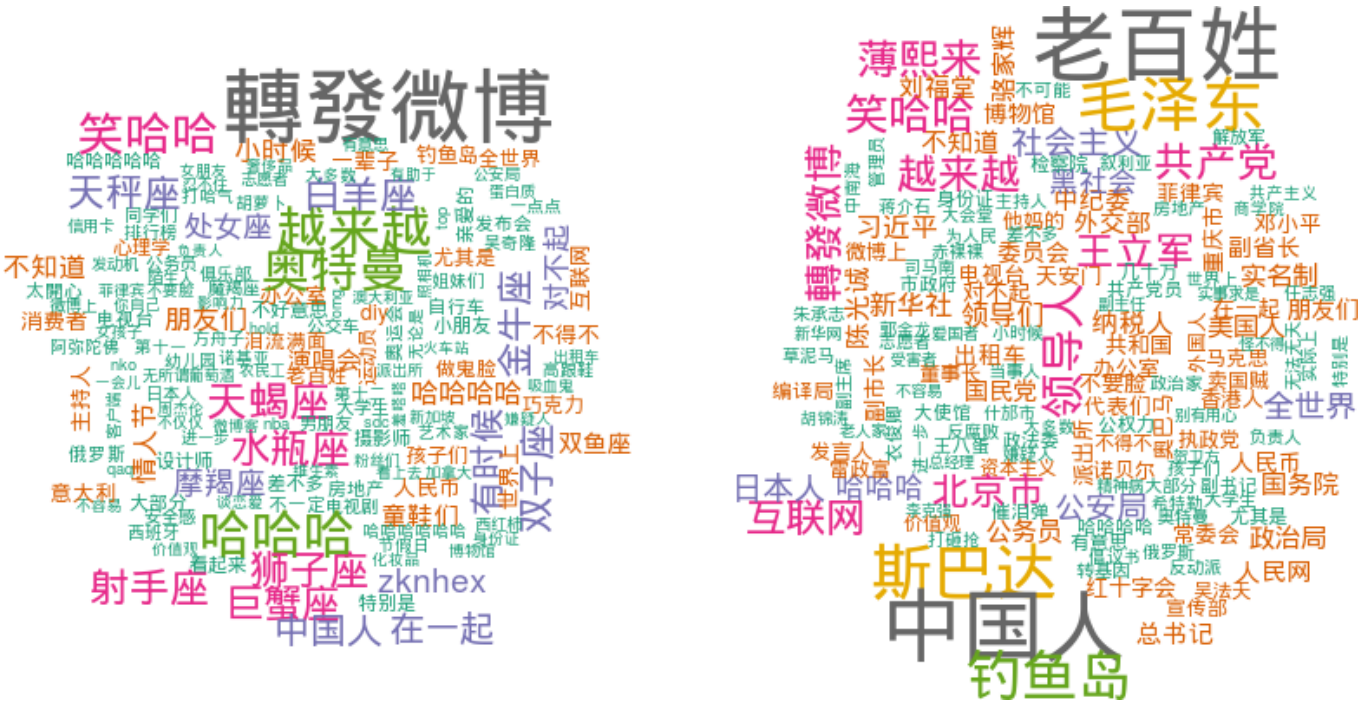

Figure 1: Noncensored (Left) and Censored (Right) Tweets Word Cloud

Statistic	Training Set	Validation Set	Test Set	Total
Retweets	47930	41285	41414	130629
Unique Users	39719	50446	50620	107907
Censored Tweets	43042 (50%)	21521 (25%)	21521 (25%)	86084 (34%)
Non-censored Tweets	43042 (50%)	63291 (75%)	63292 (75%)	169625 (66%)
Total Tweets	86084	84812	84813	255709

Table 1: Dataset Statistics

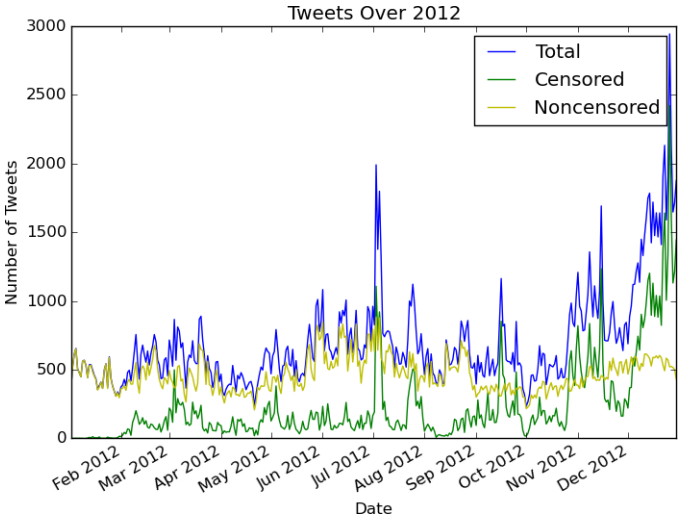

Figure 2: Tweets over Time

the end corresponds to the 18th Party Congress mentioned above. The spike in the summer of 2012 roughly corresponds to anti-Japanese protests held in multiple Chinese cities. A good predictive model should therefore require some way to account for the temporal aspect of censorship.

In other words, there are many factors at play in determining whether a tweet is censored or not. How we explicitly model these factors will be explained in the following section.

2. PREDICTIVE TASK

We would like to predict which tweets are censored in this dataset i.e. a binary-class prediction. To evaluate our model, we will simply use classification accuracy – the proportion of tweets that are labelled correctly. If we were Chinese government censors, then perhaps we would evaluate recall or F_β instead.[5] The idea is that a government censor would prefer flagging a "harmless" tweet for censorship over accidentally letting a "harmful" tweet through. We are [fortunately] not Chinese government censors, so we will stick with evaluating our model using classification accuracy.

Our data are split into training, validation, and test sets using the procedure described in Section 1. We purposely oversampled censored tweets (by including literally all of them) in our training, validation, and test sets. This oversampling makes sense for the training set, since we want a balanced sample to be used for training the model. This also certainly made the predictive task easier, since a trivial predictor – one that predicts no tweets are censored – on the entire dataset would have been 99.97% accurate while only 75% accurate on our test set.

The features we chose to include in our model can be categorized into two types: the actual text of the tweet and metadata about the tweet. For the text, we either used a simple bag-of-words representation with term frequency-inverse document frequency (tf-idf) weighting or a latent Dirichlet allocation (LDA) topic model to represent each tweet.

Processing Chinese text data is not exactly the same as processing English text. One of the major differences is that there are no spaces in Chinese text, so spaces must first be inserted between words in each document before a bag-of-words representation can be made. This process is known as segmentation.

Another major difference between Chinese and English is that Chinese is a ideogram-based language while English is an alphabet-based language. This creates problems for determining where to insert spaces. Consider the following example. The word 電 means electric and the word 腦 means brain. However, combining the two words yields 電腦, which means computer.

The segmentation task therefore becomes a machine learning problem where the model tries to predict what type of n-gram a word is. When the segmenter encounters a word like 電腦, it must decide whether this is a bigram (computer) or two unigrams (electric brain).[6] We had our hands full with our actual predictive task, so we used the basic Chinese segmenter provided in the Stanford Word Segmenter [3]. After segmentation, we removed punctuation and numbers.

One baseline we can then use is just the bag-of-words representation of each tweet, with all 229,334 terms and tf-idf weighting. The baseline accuracy for this simple representation is 76.23% on the validation set, only slightly better than the trivial predictor (75% accuracy).

Another potential baseline is to use the terms provided by the originators of the dataset [5]. In their paper, the authors do not explicitly go about trying to predict which tweets are censored, but they do provide a list of 30 terms within the paper itself that act as predictors for censorship.[7] The 30 terms can be found in the Appendix. These terms are acquired by applying something called the χ^2 selection algorithm to compare the relative frequency of keyword occurrence in censored posts and noncensored posts by the same author. We took these terms and applied a simple predictor that predicts a tweet is censored if any of the 30 terms appears in the tweet. This simple predictor achieves a classification accuracy of 57.6% on the training set (better than the trivial predictor) and 68.7% on the validation set (worse than the trivial predictor).

In an attempt to capture latent topics underlying tweets, we use LDA to express tweets as a mixture of topics. These topics do appear to have captured relevant terms. For example, with 50 topics, the three highest-weighted topics can be seen in table 2. As can be seen the highly-weighted topics correspond to sensitive or political terms.

We also incorporate metadata about the tweets (user information, time information, etc.) into our features. We created the following feature vector from **only** the training data:

1. The proportion of the user's tweets that have been censored.
2. The number of the times the user has been retweeted.
3. The number of the times the tweet has been retweeted.
4. The proportion of tweets that were censored that day.

[5]Recall that $F_\beta = (1+\beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \text{precision} + \text{recall}}$

[6]This example is somewhat stupid since it's unlikely for anyone to be talking about electric brains, but the basic point still stands.

[7]They provide a full list elsewhere, but that list has about 12000 terms in it.

1	2	3
link (holder to replace URLs)	人民 (people)	国家 (nation)
书记 (secretary)	政府 (government)	领导 (leader)
网友 (netizen)	网络 (network)	官员 ([government] official)
中央 (central)	重庆 (Chongqing)	政治 (politics)
要求 (demands)	警察 (police)	公开 (open)
调查 (inspect)	日报 (times [newspaper])	财产 (financial assets)
干部 (cadre)	法律 (law)	公布 (publicize)
信息 (information)	群众 (masses)	百姓 (common people)
发表 (issue)	行为 (behavior)	老百姓 (common people)
十八 (18)	公安 (public security)	体制 (system)

Table 2: Highest-Weighted Topics, k=50

5. The proportion of the retweeted user's tweets that have been censored, given that the tweet is a retweet.

There used to be a sixth feature that indicated the number of times that a retweeted message has been censored. However, since our subsample is only a tiny subset of the entire dataset, this value was always 0.

We create these features for the following reasons. A user who is censored more is likely to tweet things that are flagged for censorship (both in our dataset and in the future). Censors are more likely to care about popular tweets and users – which tend to mean a greater audience – than they care about unpopular tweets and users, who less people are likely to see. There are certain days (time periods) where there is some event that leads to greater discussion on Weibo and in turn greater censorship by the authorities.

When we train using only these features, we get a surprising baseline 93.19% accuracy on the validation set. This strange result will be discussed in Section 5.

3. MODEL

For this binary classification task, we didn't do anything fancy; we used L2-regularized logistic regression and SVM as implemented in the LIBLINEAR library [4]. For SVM, we did not use any kernel (like the ones provided in the LIBSVM library) since we wanted to keep training time short. A linear classifier is particularly appropriate for our bag-of-words representations, where both the number of instances and features are fairly large. With such a high-dimensional feature space, it is unlikely that mapping such data into even higher-dimensional space would have yielded increases in accuracy.

Ideally we would have used grid search or something more rigorous to tune our hyperparameters (the number of terms for bag-of-words and the number of topics for LDA). Instead, we optimized the number of terms and topics first while setting the cost $C = 1$. We then fixed the number of topics and terms to the "optimum" settings derived from the previous step and then allowed cost to vary.

When the bag-of-words representation for text was initially created, it contained 229,334 terms. We hypothesize that many of these terms might be completely useless or even detrimental to the model by inducing overfitting. We therefore attempted some dimension reduction via just dropping infrequent terms and by applying LDA (discussed later). Infrequent terms were removed based on their sparsity, using maximum allowed sparseness values of 0.99, 0.995, 0.999, 0.9999, 0.99999, and 1 (all terms included). These correspond to 48, 141, 1451, 12849, 66771, and 229334 terms,

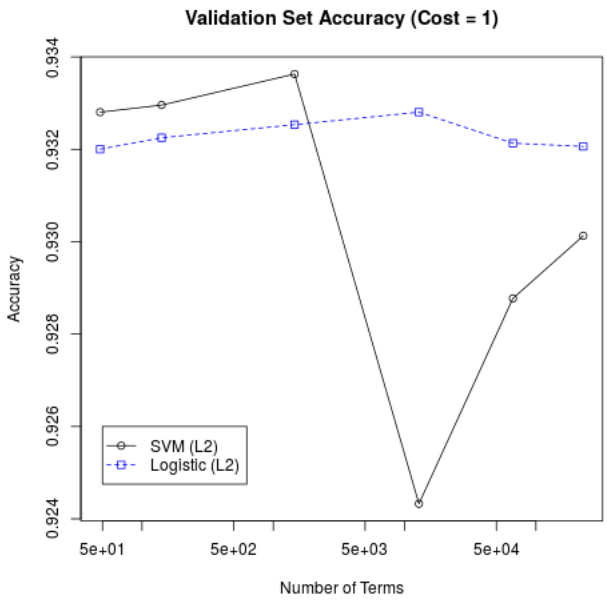

Figure 3: Tuning Bag-of-Words Sparsity

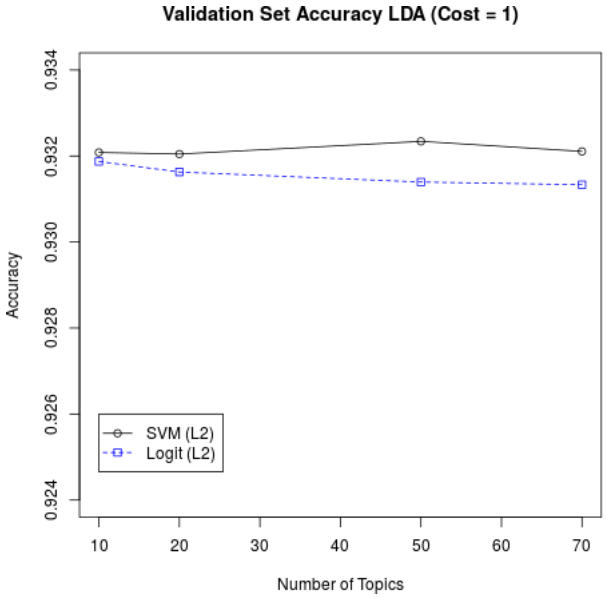

Figure 4: Tuning LDA Topics

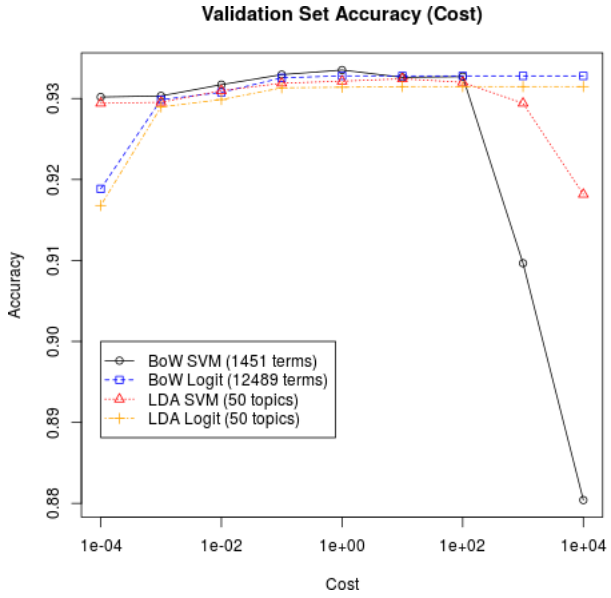

Figure 5: Cost Tuning

respectively. Figure 3 shows the results from tuning the number of terms to include in the basic bag-of-words representation of tweets.

Indeed, there is a small reduction in error on the validation set when we reduce the number of terms. The accuracy for SVM is maximized when we include 1,451 terms instead of the full 229,334 terms. The accuracy for logistic regression is maximized when we have 12,849 terms.

The results from tuning the number of topics k is shown in Figure 4. We tested the performance on four different k which are 10, 30, 50, 70. Selecting a proper number of topics can be based on human interpretability [2] or cross validation. Here, we will find what k improves the predictive power most. When we run logistic regression, $k = 10$ shows the best performance. Decreasing performance with higher k implies that too many dimensions in the topic model will cause over fitting in the predictive task. For SVM models, $k = 50$ shows the highest performance.

The costs are imposed on the logistic regression and SVM models to prevent overfitting. For both, the cost C is applied in the following optimization problem

$$\min_{w} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\xi(\mathbf{w};\mathbf{x}_i;y_i) \qquad (1)$$

where $\mathbf{w}$ refers to the weight vector, $(\mathbf{x}_i,\ y_i)$ is the set of instance-label pairs for $i = 1,\ldots,l$, and $\mathbf{x}_i \in R^n$ and $y_i \in \{-1, 1\}$. The loss function $\xi(\cdot)$ is $\max(1 - y_i\mathbf{w}^T\mathbf{x}_i, 0)^2$ for L2-SVM and $\log(1+e^{-y_i\mathbf{w}^T\mathbf{x}_i})$ for L2-logistic regression. For SVM, the cost C represents the tradeoff between maximizing the margin on the training set (overfitting?) and minimizing the classification error. For logistic regression, the cost C penalizes overly large (in magnitude) weight vectors.

The results from tuning the cost parameters are shown in Figure 5. Most of the models have very similar accuracies, which is unsurprising given the fact that much of the models' predictive power comes from the meta features. The best performance seems to be achieved for costs between 0.001 and 100, though the differences are slight. SVM tends to experience a relatively large dropoff in accuracy for costs of 100 and above, perhaps implying that accuracy is being sacrificed to maximize the margin. Logistic regression doesn't seem to experience these effects, perhaps implying that the weights were not growing large in magnitude to begin with. Lower costs also hurt logistic regression relatively more than SVM, perhaps implying that at least some overfitting on the training set is occurring. It should be noted that tuning the cost actually does very little in absolute terms, since the difference in accuracy rate is generally around 0.01%, corresponding to roughly 80 tweets in our validation set.

4. LITERATURE

The development of text mining has opened a new research topic in studying Chinese government censorship on social media. To study censorship existing studies use microblogs like Weibo [1, 5], Chinese language messages from Twitter [1], or blogs [7]. The main difference between existing studies and our study is the research goals. Existing studies try to answer what words [1, 5] or themes [7] are the targets of censorship. Even they collect every message they scrap, only deleted messages are the scope of their analysis. These studies find that politically sensitive words are likely to be deleted [1, 5]. Further analyzing what politically sensitive words and topics mean, it is found that government criticisms are relatively allowed compare to messages that have collective action potential [7]. The findings from existing studies imply using text would be powerful to predict what message is going to be censored, but they did not analyze prediction on censored / non-censored messages including other features in the model.

Existing studies could be done because of data collection strategy. All of these studies revisit their target websites multiple times to detect government censorship. To compare the relative frequency of each keyword occurrence in censored and uncensored messages, χ^2 feature selection algorithm to calculate the relative frequency is used [5]. Using

Model	Test Accuracy
BoW 1451 + Meta (SVM, C=1)	93.33%
BoW 12891 + Meta (Logit, C=1)	93.28%
BoW all only (Logit, C=1)	73.42%
LDA 50 + Meta (SVM, C=1)	93.19%
LDA 10 + Meta (Logit, C=1)	93.14%
LDA 10 only (Logit, C=1)	74.44%
Meta only (Logit, C=1)	93.10%
Trivial predictor	75%
Fu 30 baseline	68.8%

Table 3: Results

keywords to predict censorship is intuitive, but this method lacks contexts for each keyword. To overcome this problem, classifying documents by content area using supervised model is the state-of-the art method in this area of research. To classify massive of amount of data, multiple human coders classify a sample of documents and make the model aggregate and classify the remaining documents [7].[8] Our paper was motivated by the fact that the newest research on this topic considers content area and topic seriously.

5. RESULTS

Results are seen in Table 3. Every model that includes meta features show better performance than the trivial predictor. Including the meta features improve the predictive performance by 18.1%. The main issue is the fact that our baseline using only the meta features of a tweet is so high. One potential reason for this is the fact that we included too many censored tweets in our training set. By including 50% of all the censored tweets in the entire 220 million tweet dataset within the training set, we may have given the model too much meta information to work with. The major drivers behind the high accuracy are the first (proportion of user's tweets that have been censored) and fifth (proportion of retweeted user's tweets that have been censored) features. In terms of online/real-time learning, these features are slightly unrealistic, since they allow us to sort of "see into both the past and future", since these tweets cover the entirety of 2012. It could be more realistic if we change the first and fifth features to binary variables to see if there is at least one incidence of past censorship that can predict the result. Interestingly, this transformation still shows high performance (92.09%).

Including the Bag of Words (BoW) features on top of the meta features only improve around 0.2% of performance. If we include only the BoW features without meta features, the model performance is even worse than trivial predictor. Comparing this result with past studies emphasizing key words that are likely to deleted, it implies that choosing specific words in the features is better for prediction than simple bag-of-words representation. LDA features also show minimal improvement (around 0.04-0.09%). Since one tweet can only contain 140 characters, topic models do not perform well than longer documents. The low level of performance reflects this weakness of topic models on short documents. Another problem is that LDA only catches 2-5 relevant topics that are likely to be censored. Other topics do not include useful information, but increase dimensionality of the feature vectors.

It is unclear how generalizable our results are, though they are in all likelihood not particularly generalizable. The original dataset itself does not consist of a random sample of Chinese Weibo users, so our current results might only apply to popular Weibo users (or Weibo users that were popular in 2012). In addition, one might say that our accuracy results are highly dependent on the construction of our meta features and on the richness of our training set (half of all censored tweets in the entire dataset). For example, the day feature (proportion of tweets censored that day) cannot be used for real-time prediction. However, it is possible to use the basic idea of including temporal data within a real-time classifier. One potential solution is to have one feature be a sort of rolling window that measures how many tweets within the past x hours have been censored. While there would be some lag, this feature would be able to capture the increased activity of censors during certain events.

In terms of the metadata on the user, while it is perhaps unlikely that we can capture precise measurements on what proportion of a user's tweets have been censored before, it should be an easier task to determine whether a user has been censored at all before. The fact that transforming the proportion measurement into a simple binary variable provides some preliminary evidence that user metadata can be incorporated into a real-time prediction service in a similar fashion to the way described in this assignment.

In conclusion, the finding from this prediction task inform us that red-flagged users and tweets that are more popular have higher probability to be deleted. Users that have been censored in the past are more likely to be censored again, either because they simply like tweeting about sensitive topics or because they have attracted the attention of China's censors (or both). Collecting user past tweeting behavior and current retweet status can be used for real-time prediction task. Does this finding mean text information is useless for censorship prediction? The result here implies that including a relevant text feature is important for prediction. Better selection of keywords is one way to improve the text feature. Another potential way is to train a classifier on censored and uncensored documents on train set in a supervised fashion, and make the classifier to classify documents on validation and test sets. In other words, hand-code an initial tags or labels on a collection of tweets and use these tags in future classification tasks.

[8]King et al. 2013 [7] uses ReadMe [6] method which is a non-parametric approach for characterizing distribution of classes. It requires fewer assumptions than other methods. Especially, it does not need random sample of documents. Without making Naive Bayes like assumption, this algorithm examines joint distribution of characteristics and focuses on distributions makes this analysis possible.

6. REFERENCES

[1] D. Bamman, B. O'Connor, and N. Smith. Censorship and deletion practices in chinese social media. *First Monday*, 17(3), 2012.

[2] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei. Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems*, pages 288–296, 2009.

[3] P.-C. Chang, M. Galley, and C. D. Manning. Optimizing chinese word segmentation for machine translation performance. In *Proceedings of the Third*

Workshop on Statistical Machine Translation, StatMT '08, pages 224–232, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[4] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, June 2008.

[5] K.-w. Fu, C.-h. Chan, and M. Chau. Assessing censorship on microblogs in china: Discriminatory keyword analysis and the real-name registration policy. *Internet Computing, IEEE*, 17(3):42–50, May 2013.

[6] D. J. Hopkins and G. King. A method of automated nonparametric content analysis for social science. *American Journal of Political Science*, 54(1):229–247, 2010.

[7] G. King, J. Pan, and M. E. Roberts. How censorship in china allows government criticism but silences collective expression. *American Political Science Review*, 107(02):326–343, 2013.

APPENDIX

A. CODE

Code for this project can be found on our Github page.

B. BASELINE 30 TERMS

These 30 terms are provided by the original authors of the datset and can be seen in Table 4 [5].

Term
重庆
光诚
陈光诚
两会
骆家辉
辟谣
代表
薄
日报
公布财产
北京日报
薄熙来
人大代表
骆家辉公布
财产
转发
转
王立军
求证
转发微博
请骆家辉
书记
转么
鬼子转
公布
求辟谣
删
陈
微博
养老不”

Table 4: 30 Terms for Censorship Prediction