Work-in-progress: Automated Named Entity Extraction for Tracking Censorship of Current Events

Antonio M. Espinoza
Computer Science Department
University of New Mexico
amajest@cs.unm.edu

Jedidiah R. Crandall
Computer Science Department
University of New Mexico
crandall@cs.unm.edu

Abstract

Tracking Internet censorship is challenging because what content the censors target can change daily, even hourly, with current events. The process must be automated because of the large amount of data that needs to be processed. Our focus in this paper is on automated probing of keyword-based Internet censorship, where natural language processing techniques are used to generate keywords to probe for censorship with. In this paper we present a named entity extraction framework that can extract the names of people, places, and organizations from text such as a news story. Previous efforts to automate the study of keyword-based Internet censorship have been based on semantic analysis of existing bodies of text, such as Wikipedia, and so could not extract meaningful keywords from the news to probe with.

We have used a maximum entropy approach for named entity extraction, because of its flexibility. Our preliminary results suggest that this approach gives good results with only a rudimentary understanding of the target language. This means that the approach is very flexible, and while our current implementation is for Chinese we anticipate that extending the framework to other languages such as Arabic, Farsi, and Spanish will be straightforward because of the maximum entropy approach. In this paper we present some testing results as well as some preliminary results from probing China's GET request censorship and search engine filtering using this framework.

1 Introduction

There are many open questions about Internet censorship, including how effective it is, what makes it effective, what kinds of targeted activities it is effective (or is not effective) at stopping, and so forth. A first step toward answering any of these questions is to collect enough data to understand how censorship is applied and what kinds of activities are targeted by the censors. This implies automated probing that is broad and carried out over a long period of time, because censorship within a single country can vary from province to province, company to company, and technology to technology and what content is targeted can change daily, even hourly.

1.1 Related work

Our focus in this paper is on keyword-based Internet censorship, and for the preliminary results we present we are interested specifically in China. Keyword-based Internet censorship in China has been studied by several groups of researchers, but is not well understood. An anonymous government official writing as "Mr. Tao", in a report published by Reporters Without Borders [7], described three types of keywords: masked words, sensitive words, and taboo words. According to Mr. Tao, a keyword list is produced and updated by the Information Office of the State Council. He adds, "each site adds key-words to its own filters in order not to run the risk of being criticised, punished or, worse still, closed down."

One of the more thoroughly studied forms of keyword-based Internet censorship is GET request filtering at the router level, where GET request packets containing blacklisted keywords cause routers in the backbone of China's Internet to forge reset packets and attempt to reset the TCP connection between the offending client and server. In contrast to HTML response filtering, which appears to have not been effective and may have been discontinued [8], GET request filtering is very effective in terms of the ratio of offending connections that are reset and is still pervasive on China's Internet today. The methods of China's HTTP keyword filtering were first published by the Global Internet Freedom Consortium [4]. Clayton *et al.* [2] published a more detailed study of this mechanism. The ConceptDoppler project [3] found that HTTP keyword filtering in China is not peremptory and is not strictly implemented at the border of the Chinese Internet, with a significant amount of filtering occurring in the backbone. The ConcpetDoppler project also used latent semantic analysis [6] to cluster words from the Chinese-language version of Wikipedia around sensitive concepts and then probe with these potentially sensitive words to see if they are censored by the GET request router-based mechanism. ConceptDoppler initially produced a list of 122 words, and has produced two more lists since.

Software that runs on servers in China, such as blogging software, also implements keyword-based censorship. One snapshot of a blacklist from a blog site is available in a Human Rights Watch Report [5, Appendix II] from 2006, for example. Client-side programs such as chat clients also implement keyword censorship. A blacklist for QQChat is available in the same report [5, Appendix I], and Villeneuve [9] gives a high-level analysis of topics censored by the chat client that is part of TOM-Skype.

Note that all of these lists are one-time-only snapshots. Some of the lists are very different from others, suggesting they come from different sources. Furthermore, our preliminary results indicate that the HTTP GET request blacklists that are used by routers in the backbone of China's Internet do not change on a daily, weekly, or even monthly basis. Existing systems that can continuously probe, such as ConceptDoppler, are based on document summary techniques that cluster words based on concept and therefore are not suitable for finding the named entities that are relevant to current events. Such document summary techniques can only compare documents and terms to an existing corpus of text based on the semantics that are latent in term and document frequencies, while named entity extraction gives additional semantic information about what a document is about based on its use of named entities. Because of the lack of data about Internet censorship and appropriate methods for gathering the data broadly and over a long period of time we have developed a named entity extraction framework, which we present in this paper.

1.2 Structure of the rest of the paper

We discuss the implementation of our framework in Section 2. Then we explain our experimental methodology for our preliminary results in Section 3 followed by the results in Section 4 and some concluding remarks.

2 Implementation

We implemented a *named entity extraction* (NEE) framework by means of *maximum entropy* (ME) machine learning. Borthwick *et al.* [1] demonstrated that an ME approach to NEE allows for flexibility in the choice of features to train on, since the interactions among features are not as important as they would be in other approaches such as Hidden Markov Models or Maximum Likelihood. We focused on three types of named entities: names of people, names of places, and names of organizations.

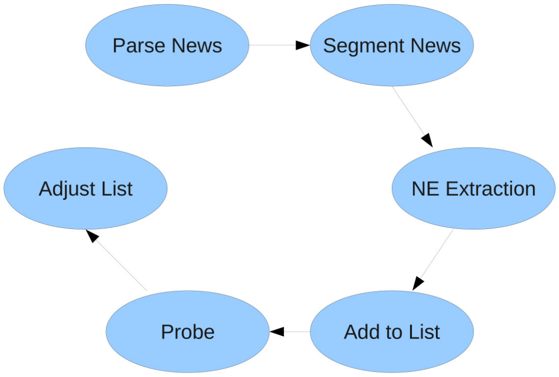

Figure 1: **The high-level workflow of our implementation.**

Our NEE framework requires a training corpus that has existing *labels*. That is, every word in the training corpus should be labeled with one of four labels: as a name of a person, name of a place, name of an organization, or not any of these types of named entities. The first three groups are then subdivided into complete, beginning, middle or end of the type of label. This is done so that it is possible for a named entity to span multiple segments after segmentation. We used the Chinese-language version of Wikipedia as our training corpus. When people, places, or organizations appear in Wikipedia, the reference is often a link to that person, place, or organization. In addition to the labeled data the ME framework also requires a feature vector for each word in order to build a model of *features* that correlate with particular labels. A feature is a property of the labeled word. One example feature is whether the word contains any characters that are common Chinese surnames. Another example feature is if the word is followed by a possessive such as Chinese 的 (de). The following table is a complete list of features used.

Feature	**Test**
Is place?	Does the word translate to a known place?
Has a name character?	Does the word contain a common name character?
Has punctuation?	Does the word contain any punctuation?
What punctuation (if any)?	What punctuation does the word contain?
Is month?	Does the word contain the character 月?
Has capital letters?	Does the translated word contain roman characters that are capitalized?
Has number?	Does the word consist of only roman numerals?
Has a Chinese number character?	Does the word contain a Chinese number character (一，二，三...十).
Has de?	Does the word contain the symbol 的?
Is a dictionary term?	Is the word in a Chinese dictionary?
Parts of Speech	All the parts of speech the translated word has.
Number characters	The number of characters in the word.

We used several heuristics to treat the Wikipedia corpus as a labeled data set. We assumed the link was a label for a name of a person if the document that was linked to had 年出生 (year of birth), 年逝世 (year of death), or 年逝世人物列表 (year of death person list) among its categories. We assumed the link was a label for a name of a place if the document that was linked to had GPS coordinates associated with it or contained one of the following infoboxes: country, city, cncity (cn=Chinese), prc provence (prc=Peoples Republic of China), or university. Lastly, we assumed the link was a label for an organization if it linked to an article that contained a company or organization infobox.

In all of the experiments in this paper, we trained on one third of the Wikipedia corpus using the above labeling scheme, and then tested on a different third of the corpus. For both training and testing, we applied Chinese text segmentation to the entire corpus to divide it into words (because the Chinese written language does not use spaces to divide sentences into words). Then we assigned a feature vector to each word based on the word itself, as well as the word that precedes and follows it. We trained and tested each of the three types of named entities separately.

Using an ME toolkit, we then assigned conditional probabilities to each word for each sublabel conditioned on its feature vector. Because the probabilites given were for a word being the begining 'label', middle 'label', end 'label', complete 'label', or not a 'label' (where 'label' is person, place, or organization), we had to find the highest probable legal path through the output. In order to be a legal path sublabels must be in correct order, for example end 'label' cannot precede middle 'label' legally. Similary beginning, middle, and end 'label' cannot be surrounded by not 'label' on both sides. In order to accomplish this we used the fact the output of the ME tookit can implicitly be thought of as a directed acyclic graph. Therefore we were able to preform a topological sort to find the highest probable legal path.

For testing or for the actual probing experiments, we take an unlabeled corpus of text (or a test set where the labels are withheld), and then assign a label to each word based on the ME model of the training set. We scale the conditional probabilities in the model linearly so that we get a desirable fraction of labeled words.

See Figure 1 for a high-level workflow of our implementation. For probing, we have written parsers for seven Chinese-language news websites. Our framework downloads the news from these websites every day and performs named entity extraction based on the model that was created using Wikipedia. For any word that is labeled as a named entity, we include that word in our list of keywords to probe with on that day. Our probing infrastructure performs two kinds of probes, it tests twelve servers for HTTP GET request filtering based on forged RSTs, and it tests two search engines to see if the word elicits a legal message in Chinese stating that entries have been removed from results for the search query. Our probing infrastructure has multiple priority levels, with levels with lower numbers being higher in priority for testing. If a word is ever interpreted to be blacklisted, it is placed in priority 0 so that it will be tested every 12 hours for the remainder of the probing. Words enter the probing infrastructure at level 1. Every 12 hours level 0 words are probed, followed by level 1 words, then level 2, and so on. If a word does not appear to be blacklisted, it is moved down one priority, except if it is in priority 0 in which case it remains in priority 0. There are 15 priorities, with the lowest being 14. After a word has been probed 14 times and does not appear to be censored, it falls off the bottom of the list.

In order to get search engine results that are independent of GET request censorship, we divide GET requests for the two search engines we test against into separate packets that will be reconstructed by the server but will evade GET request filtering. For testing for forged RSTs, we wait at least 100 seconds between each query for any pair of IP addresses. As a special consideration, the search engine results do not affect the priorities of keywords, because we found this to cause many words that were not actually targets of censorship to be in priority 0. We record a traceroute to each server every hour, so that any major changes in the keyword censorship that might be due to changes in routing can be explained.

3 Expiremental Metholodogy

For the preliminary results we present in this paper, there are two experiments that we performed. One experiment is to measure the specificity and recall of the NEE framework on a different third of Wikipedia than the training set. This gives us baseline numbers to see how well the NEE framework is performing. The other experiment for which we have preliminary results is a test run of approximately two months (with some downtime) in which we ran the entire NEE and probing framework and obtained some results that are censored topics from the news.

For the first experiment, we focus on specificity instead of precision because of the context of probing with keywords. Precision is the probability that a word labeled as a named entity actually is a named entity. Since there are no human consumers of the output of our NEE framework, precision is not as relevant. Any word that is not a named entity but is labeled as such will be probed with, perhaps unnecessarily, but this is relatively acceptable compared to missing named entities. What we wish to consider instead is how much extra probing we have to do to ensure that we label a good fraction of the named entities as named entities. Thus, recall and specificity are better indicators of performance in our context than recall and precision. Recall is the probability that an actual named entity is labeled as a named entity. Specificity is the proportion of words that are not named entities that are not labeled as named entities. Thus, as long as the specificity remains high enough that NEE is saving us about an order of magnitude of probing compared to just probing with every word, we can trade off precision for recall and achieve a high recall while greatly reducing the amount of necessary probing.

For the second experiment, our initial two months of running the entire infrastructure includes downloading and parsing the news from seven sources every day, labeling the named entities, and probing for both GET request and search engine censorship. This data has various issues such as downtime and the need to remove some polluted data manually, but gives promising anecdotal evidence that censorship of current events can be detected using NEE. We provide a summary of the types of words we found to evoke censorship and how the different forms of censorship seem to vary with the news, with the caveat that these are preliminary results and no certain conclusions can be drawn from them at this time.

4 Results

In this section we present both sets of results: results from testing for the specificity and recall by withholding labels from the Wikipedia dataset, and preliminary results from an initial two-month run of the entire infrastructure.

4.1 Specificity and recall

For labeling the names of people, we obtained the following results:

- Specificity: 83.44%
- Recall: 89.63%
- Precision: 0.42%

A precision of 0.42% is usually not considered to be very good for a named entity extractor, but remember that our context is different. One way to interpret these results is that we can label only 16.6% of the words in our dataset as names of people (thus reducing the amount of probing necessary by nearly an order of magnitude), and include 89.63% of the actual names of people in our probing by doing so.

The results for names of places are as follows:

- Specificity: 69.80%
- Recall: 96.3%
- Precision: 0.77%

And, finally, the results for names of organizations are as follows:

- Specificity: 88.40%
- Recall: 87.56%
- Precision: 0.28%

4.2 Initial two-month run

One of the more surprising results from our initial two months of data is that the HTTP GET request blacklist appears to be fairly static. That is, words do not seem to be added to or removed from this particular censorship blacklist on a daily, weekly, or even monthly basis. We remind the reader that these are preliminary results and our data has some downtime and other issues. However, our data was taken during a time of many reports of arrests and censorship related to the Jasmine Revolution protests in China in 2011, and despite many of these current events being censored in search engines we probed with these keywords for HTTP GET request censorship and saw none that was related to any current event. Nor did we see any evidence of any keyword being added or removed in our preliminary results.

We did notice that current events evoked censorship in search engines, however. Specifically, certain words caused the search engine to return a warning that results had been removed due to local laws. Note that this probably means that a website was removed that contained the word we probed with and was highly ranked, it does not mean that the word itself is on any keyword blacklist. This is an important distinction. We determined that this is probably the case with the following experiment. We searched for both "fuck" and "fuck you" in both search engines that we used for probing. The word "fuck" causes the message saying results have been removed to appear, while "fuck you" does not cause the message to appear. This suggests that this form of censorship is more topical and not based solely on a certain byte string appearing in the query. However this does not preclude a blacklist for search engine censorship.

We witnessed search engine censorship of certain words from the news that we assert is definitely censorship based on current events because of the words themselves. Some of the words are:

- 茉莉花 (Jasmine Flower): related to the Jasmine Revolution protests.
- 诺贝尔 (Nobel Prize), 刘先生 (Mr. Liu), LIU, Xiao, Liu, and 挪威 (Norway): these are all related to Liu Xiaobo winning the Nobel Prize while imprisoned by China.
- 七十七, 77, 七七 (all mean the number 77): these elicited censorship at a time when China's President was being criticized by many Chinese citizens. He had visited a woman at her home during a live newscast, and asked her how much she pays for rent. She replied that she paid 77 RMB, or approximately 12 dollars, per month. It is likely that she is on a government program and this is the small portion of the actual rent that she pays, but many felt that she had been coerced by the government to claim that her rent was very low to make the government look good.
- 王府井 (Wangfujing): this is an area in Beijing where some of the Jasmine Revolution protests happened.

5 Concluding Remarks

In conclusion, our preliminary results are promising in terms of building an infrastructure that can probe censorship with words from current events. Our NEE algorithm gives a good specificity and recall, and we demonstrated that this infrastructure can produce instances of censorship that are related to current events.

6 Acknowledgments

We would like to thank the anonymous FOCI reviewers for their insightful comments. We would also like to thank the many people who helped us improve our translations and gave feedback on other aspects of the paper. Fletcher Hazlehurst, Veronika Strnadova, Leif Guillermo, and Ronald Garduno helped with various aspects of implementation or understanding of the maximum entropy framework and features. This material is based upon work supported by the National Science Foundation under Grant Nos. #0844880 and #1025442. Antonio Espinoza and Jedidiah Crandall are also supported by the DARPA CRASH program.

References

[1] Borthwick, A., Sterling, J., Agichtein, E., and Grishman, R. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In *In the Proceedings of the Sixth Workshop on Very Large Corpora* (1998), pp. 152–160.

[2] Clayton, R., Murdoch, S. J., and Watson, R. N. M. Ignoring the Great Firewall of China. *I/S: A Journal of Law and Policy for the Information Society 3*, 2 (2007), 70–77.

[3] Crandall, J. R., Zinn, D., Byrd, M., Barr, E., and East, R. ConceptDoppler: a weather tracker for Internet censorship. In *Proc. of 14th ACM Conference on Computer and Communications Security (CCS)* (2007).

[4] The Great Firewall Revealed. Whitepaper released by the Global Internet Freedom Consortium in December of 2002.

[5] "Race to the Bottom": Corporate Complicity in Chinese Internet Censorship. In *Human Rights Watch* (August 2006). http://www.hrw.org/reports/2006/china0806.

[6] Landauer, T. K., Foltz, P. W., and Laham, D. Introduction to latent semantic analysis. *Discourse Processes 25* (1998), 259–284.

[7] Mr. Tao. China: Journey to the heart of Internet censorship. Investigative report sponsored by Reporters Without Borders For Freedom and Chinese Human Rights Defenders, Oct 2007.

[8] PARK, J. C., AND CRANDALL, J. R. Empirical study of a national-scale distributed intrusion detection system: Backbone-level filtering of html responses in china. In *Proceedings of the 2010 IEEE 30th International Conference on Distributed Computing Systems* (Washington, DC, USA, 2010), ICDCS '10, IEEE Computer Society, pp. 315–326.

[9] VILLENEUVE, N. Breaching trust: An analysis of surveillance and security practices on China's TOM-Skype platform. Available at `http://www.infowar-monitor.net/breachingtrust/`.