English Dataset For Automatic Forum Extraction

Jakub Sido & Miloslav Konopík & Ondřej Pražák

NTIS – New Technologies for the Information Society,
Faculty of Applied Sciences, University of West Bohemia, Technická 8, 306 14 Plzeň
Czech Republic
WWW home page: http://nlp.fav.zcu.cz

Abstract. This paper describes the process of collecting, maintaining and exploiting an English dataset of web discussions. The dataset consists of many web discussions with hand-annotated posts in the context of a tree structure of a web page. Each post consists of username, date, text, and citations used by its author. The dataset contains 79 different websites with at least 500 pages from each. Each web page consists of a tree structure of HTML tags with texts taken from selected web pages. In the paper, we also describe algorithms trained on the dataset. The algorithms employ basic architectures (such as a bag of words with an SVM classifier and an LSTM network) to set a baseline for the dataset.

1 INTRODUCTION

In the last years, a significant portion of human social lives moved into the Internet, and many social networks have appeared since. Long ago, it became clear that these networks contain a lot of valuable information. Nowadays, the most significant social networks are heavily monitored and analyzed by various autonomous algorithms. For big social networks, it makes sense to create a dedicated crawling script to gather the data it contains. However, small networks appear and disappear on a daily basis, and such effort is not profitable. Nevertheless, a lot of valuable information may be gathered when all the small networks are combined.

Every social network is unique to some extent. Crawling scripts would have to be manually created (or at least customized) for every network. Such an effort is not very profitable for small networks. Therefore, we propose to create an automated extraction algorithm instead. In this paper, we present a dataset dedicated to training such algorithms.

Our primary motivation to target small social networks consists in monitoring potentially harmful, unwanted or dangerous activities on the web. This would allow early prevention of such activities (suicides, crimes against society, sexual abuse cases). We believe that small networks are very prone to bad activities in general because they are not currently so well monitored.

In the paper, we describe algorithms trained on the dataset. The algorithms employ basic architectures (such as a bag of words with an SVM classifier and an LSTM network) to set a baseline for the dataset. Even with these simple architectures, we reach a promising accuracy of data extraction. Still, we believe that there is a big room for further improvement.

2 RELATED WORK

There are a few other works which are related to this topic. We can split them into two categories: datasets and tools. The first group deals with automatic data extraction from diverse sources. However, there are also some systems that operate on the forum data, too.

2.1 Datasets

Several datasets appeared in the last decade with increasing interest in the automatic data extraction. The datasets exist primarily for English. However, there are some datasets for other languages as well.

Automatic Extraction of Informative Block from webpages This dataset was created in 2005. A group from the Pennsylvania State University took 11 news websites and random news from every section. Altogether, it was 5911 pages. This dataset is not publicly available(Debnath et al., 2005).

Shallow Information Extraction from Medical Forum Data This project(Sondhi et al., 2010) deals with searching for answers on the medical discussion forum. The authors took 175 posts in 50 threads from the Healthboards forum. The dataset is publicly available[1].

Exploiting thread structures to improve smoothing of language models for forum post retrieval This project (Duan and Zhai, 2011) used a dataset created by the authors from the CNET Computer Help forum for searching for relevant posts in the context of a query. They made the hard copy of 29 413 threads with 135 752 posts.

Learning Online Discussion Structures by Conditional Random Fields This work(Wang et al., 2011) deals with searching relevant answers on discussion forums. The authors used a dataset they collected from three different online discussion forums. Altogether it contains more than 180 000 posts in 31838 threads. The dataset is publicly available.[2]

2.2 Existing Systems

Several systems allow users to search through web discussions:

- https://webhose.io
- http://boardreader.com
- http://omgili.com/

Many of them are operated commercially. Some of them offer free usage with a restricted number of queries. However, none of them provides raw data for any further analysis.

[1] http://sifaka.cs.uiuc.edu/ir/index.html
[2] http://sifaka.cs.uiuc.edu/\textasciitildewang296/Data/index.html

3 DATASET DESIGN

3.1 Target Language

The presented dataset consists of websites in the English language. In the future, we will extend the dataset to other languages.

3.2 Annotation process

During the annotation process, the annotators downloaded the web pages with discussions and labelled by hand using XPath. For each forum, they created manually the XPaths which specifies the informative field's (e. g. author names, dates, and texts) position in an HTML tree. Annotator could use XPath extraction tool (for example developer tools in a web browser), but XPath had to be generalized by hand. The task of the annotators was to define an XPath for each class for every annotated site.

Every useful part of the post is classified into one of six classes:

- *author*,
- *date*,
- *text*,
- *citation author*,
- *citation date*,
- *citation text*.

In order to make evaluations of the prospective future systems more accurate, we labelled only those elements that contain the desired text information[3] directly. It is, of course, possible to mark whole subtrees of these tags as relevant but then we could get potentially textless tags as class-labeled.

You can see the annotation output example on Figure 1.

```
{
    "author": "//a[@class='bigusername']",
    "date": "//td/div[@class='normal'][2]",
    "text": "//tr/td[@class='alt1']/div[1]",
    "citation_author": "//tr/td[@class='alt2']/div[1]",
    "citation_date": "/NONE",
    "citation_text": "//tr/td[@class='alt2']/div[2]"
}
```

Fig. 1: Annotation output example

[3] e.g., post text or nickname of an author

Afterwards, every web page is automatically transformed into a representation in an easily machine-readable format for further processing. We will describe the transformation later in the section 4. This format preserves the tree structure of a web page but makes further analysis much easier. However, the original pages with XPaths are available, too.

3.3 Design Decisions

Many forum pages do not conform to any HTML markup standard (they are not valid). In some other cases, they are valid, but they introduce other problems. For example, the information is not present in any structure on some forums. This means that all the data is written in one HTML element. We also have to deal with citations, data deletion and other issues. In the following sections, we describe our solution to these problems.

Citations And Responses Citations or responses are used often on the web forums. As the original idea of creating this dataset is to be able to detect and extract posts, we do not try to keep the tree of interactions between users as the authors of (Debnath et al., 2005) do. We want to mark all users and their posts. Thus, every post was grabbed regardless of whether it is a reply. At the same time, this structure can be derived from the tree structure of the web page, which we include in the dataset as well.

Example of response:

```
John : Is anybody interested in...
  Luke : Yes, I am.
  Pete : I am not.
```

Only the citations which are often used directly in users' texts should be separated from the posts. In some cases, the citation does not contain the whole content or the date of the original post. Thus, it is necessary to create new classes for them.

Example of citation:

```
Luke : "Is anybody interested
    in..."(John)
  Yes, I am !
```

Template Information On some forums, posts are marked as deleted instead and they are still present in the HTML tree. Usually, the date of publication of such post stays in its position, but the text is replaced with some template text. In some cases, the original element stays on its position, and only the text is substituted for example with one of the following:

- This post was deleted by the author.
- Deleted by admin one day ago.

Sometimes the original subtree with the post is replaced with another one e.g.

```
<div class="post">
   <h4>John</h4>
   <p>Text of john's post</p>
</div>
```

is replaced by:

```
<div class="deleted">
   <span>Deleted</span>
</div>
```

In both cases, it is up to the prospective user of this data set to handle it.

Missing Data Citations are mentioned with the dates of original publication on some forums, while on other forums they are not. For this reason, it is possible that the information about the date of publication of the cited post is missing.

Mixed elements During creating of the data set, we found several forums that have different pieces of semantic information mixed within one element. In most cases, it was the date of publication and the author's name that was mixed. Like:

```
<div>
   Posted by John one day ago
</div>
```

Considering the fact we want to keep the data set as simple as possible and to the number of such forums, we decided not to include these sites in the data set.

4 DATA FORMAT

We designed the dataset taking into account the tree structure of the web pages. It is naturally possible to walk through the tree-structured page using the standard graph algorithms. The elements in the data set files are ordered as the preorder search finds them.. Each element contains the id of its parent to keep the tree structure.

Each node is represented by a separate line in the format:

```
ID PARENT_ID TAG_NAME CLASS TEXT
```

Where class is defined by mapping in Table 1.

Class	Id
other - everything other like ads, e.g.	0
nick - a username used by an author	1
date - a date of publication of the post	2
text - a text of the published post	3
citation author - a username of the cited author	4
citation date - a date of the original cited post	5
citation text - a text of the citation	6

Table 1: Mapping of classes on id number

Lets suppose we have a structure like this:

```
<div>
   <div>
       <p>Posted by John</p>
    </div>
    <div>
       <span>
           One day ago
        </span>
        This is my favourite place
        in Roma.
        <img />
        On Saturday I'm flying to
        Italy.
    </div>
</div>
```

This structure will be transformed into the following format:

```
0;-1;div;0
1;0;div 0
2;1;p;1;Posted by John
3;0;div;3;This is my favourite place in Roma.
          On Saturday Im flying to Italy.
4;3;span;2;One day ago
5;3;img;0
```

In this, way the dataset is converted into the format suitable for other processing.

5 CLASSES COUNTS AND STATISTICS

The Table 3 shows some basic dataset statistics.

Class	Relative frequency
nick	1.8%
date	1.8%
text	2.2%
citation author	1.9%
citation date	0.8%
citation text	1.5%
others	90%

Distribution of classes

Total number of forums	79
– training part	50
– testing part	29
Total number of pages	65 242
Minimum pages per forum	501
Maximum pages per forum	2317
Average number of pages per forum	826

Statistics of counts of downloaded pages

Table 3: Dataset statistics

We can notice the imbalance between the class ”other” and the remaining classes. However, it is an expected result provided that the number of tags with the target information is very low.

6 FIRST EXPERIMENTS

In order to set a baseline on the dataset, we conduct first experiments of data extraction from the forums. The experiments consist of *preprocessing*, *feature extraction* and *classification*. We describe all the steps in the following sections.

6.1 Preprocessing

The forums have some specific properties regarding the employed language and vocabulary. For example, author's names (nicks) are typically composed of a mixture of (usually artificial) names, numbers and special characters. Such words create a lot of low-frequency vocabulary items that are hard to classify. Therefore, we transform some specific groups of characters into predefined symbols. The groups of our interest are:

- Capital letters
- Small letters
- Numbers
- Non-alphanumeric characters

Thanks to the above-described transformations, usernames like *Jack59*, *Frank41* or *Stephan235* are projected onto the same word representation for the subsequent processing. The same happens with the dates like *2013-08-04* and *2001-02-07*. Other examples can be found in Table 4.

Next, we use other standard preprocessing techniques such as tokenization based on a regular expression and lower casing. The lower casing is performed after the above-mentioned transformation.

Infrequent words	Mapped on
Jack59 Frank41 Stephan235	Aa1
john_23 george-4	a-1
2012-04-03 2009-03-12	1-1-1

Table 4: Example of replacing groups of characters

6.2 Classification Classes

Our dataset contains some classes (see Table 1) that are relatively similar. The similar classes form the following pairs: *nick – citation author*, *date – citation date*, *text – citation text*.

We expect that these pairs would create problems for the classifier. Therefore, we have decided to merge the pairs of similar classes. In our results, we show scores for both reduced four classes dataset and the original seven classes dataset.

6.3 Features

In order to keep our classification architectures simple and straightforward, we use the following basic features:

- K-most frequent words (the K is depended on the classifier – see section 6.4).
- Character masks – created by the transformation described in section 6.1.
- HTML tags (67 different HTML tags such as *div*, *img*, p etc.).

6.4 Classifiers

We employ two classifiers: the SVM classifier and the LSTM (Hochreiter and Schmidhuber, 1997) classifier.

The *SVM classifier* uses bag-of-words features based on a dictionary created from 200 most frequent words and 530 masks (section 6.1) from the training part of the dataset. HTML tags are in the form of one hot vector.

The *LSTM classifier* is a recurrent neural network that takes sequences of words as input. In our approach, we use randomly initialized embeddings to convert words into low dimensional vectors of real numbers that are fed on the input of the LSTM network. In this approach, we consider words that occur at least 15 times in the training portion of the dataset and the same set of 530 masks. We use the following hyper-parameters: *dimension of word embeddings*: 300, *LSTM hidden dimension*: 256, *dropout rate*: 0.5, *learning rate*: 0.0001, *optimization algorithm*: Adam.

6.5 Experiment results

Following tables summaries the results of the different configurations of experiments.

Classifier	Text + Mask	HTML Tags	F1
SVM	✓	✗	47.45
	✗	✓	36.52
	✓	✓	54.92
LSTM	✓	✗	62.74
	✓	✓	**65.17**

Table 5: 4 class classification - authors, dates and texts classes are merged see 6.2

Classifier	Text + Mask	HTML Tags	F1
SVM	✓	✗	34.28
	✗	✓	27.72
	✓	✓	40.36
LSTM	✗	✓	45.35
	✓	✓	**48.14**

Table 6: 7 class classification

The dataset divided to train, test and validate parts will be freely available on our website as well as the implementation of the experiment.

7 Future Work

Dataset Extension We plan to extend the dataset by adding more web-pages and different languages. With a bigger dataset and better algorithms (see the next paragraph) we indent to increase the dataset automatically. First, we will run the automatic extraction algorithm on a set of new web-pages. Then, we will generalize the XPaths of the extracted elements to generate correct XPaths for the web-pages. This process would have to be supervised at first, but we expect that at later phases, the whole process can be fully autonomous.

Advanced Algorithms A few issues showed up during the first experiments. Replacing of groups of characters brought some improvement, and when combined with the LSTM the results look promising. In the following research, we intend to focus on the tree structure of the web pages. The trees of web-pages are much larger than trees that parse trees. We expect that the task of dealing with the structure of web-pages will be fairly challenging.

Discussion contributions (posts) appear in a repetitive pattern on forums. We aim to design algorithms that would capture these patterns and use them to improve the classification accuracy.

8 Conclusion

The result of the above-described work is the new data set containing more than 30 000 web pages with forum discussions from 79 different web servers. This dataset can be used for training of algorithms for automatic extraction of forum posts from diverse sources. Concerning this purpose, it is designed to contain lots of different web servers with various layouts. This data set will be publicly accessible from our departmental web server.

Acknowledgement

This work has been partly supported from ERDF ”Research and Development of Intelligent Components of Advanced Technologies for the Pilsen Metropolitan Area (InteCom)” (no.: CZ.02.1.01/0.0/0.0/17_048/0007267) and by Grant No. SGS-2019-018 Processing of heterogeneous data and its specialized applications.Computational resources were provided by the CESNET LM2015042 and the CERIT Scientific Cloud LM2015085, provided under the programme ”Projects of Large Research, Development, and Innovations Infrastructures”

[Debnath et al., 2005]Debnath, S., Mitra, P., and Giles, C. L. (2005). Automatic extraction of informative blocks from webpages. In *Proceedings of the 2005 ACM Symposium on Applied Computing*, SAC '05, pages 1722–1726, New York, NY, USA. ACM.

[Duan and Zhai, 2011]Duan, H. and Zhai, C. (2011). Exploiting thread structures to improve smoothing of language models for forum post retrieval. In Clough, P., Foley, C., Gurrin, C., Jones, G. J. F., Kraaij, W., Lee, H., and Mudoch, V., editors, *Advances in Information Retrieval*, pages 350–361, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Hochreiter and Schmidhuber, 1997]Hochreiter, S. and Schmidhuber, J. (1997). Lstm can solve hard long time lag problems. In *Advances in neural information processing systems*, pages 473–479.

[Sondhi et al., 2010]Sondhi, P., Gupta, M., Zhai, C., and Hockenmaier, J. (2010). Shallow information extraction from medical forum data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1158–1166, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Wang et al., 2011]Wang, H., Wang, C., Zhai, C., and Han, J. (2011). Learning online discussion structures by conditional random fields. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 435–444, New York, NY, USA. ACM.