# Extraction of Citation Data from Websites based on Visual Cues

**Master's Thesis Exposé**

Tim Repke

March 8, 2016

## 1 Introduction

Every scientific article, be it an essay or a paper in a journal, provides citations and a bibliography to support its arguments. Publishers and universities require their academics to format their bibliography using predefined styles like APA, MLA or Harvard, to name just a few.

Obviously it is a tedious job to do that by hand, which is why there is a multitude of software to format the references following style guides, provided the required information is present in a structured format. Some services (citation generators), like CiteThisForMe[1], EasyBib[2] or RefME[3] even try to provide this information automatically, so the user only needs to enter an identifier like an ISBN, DOI or URL to add a specific work to the bibliography.

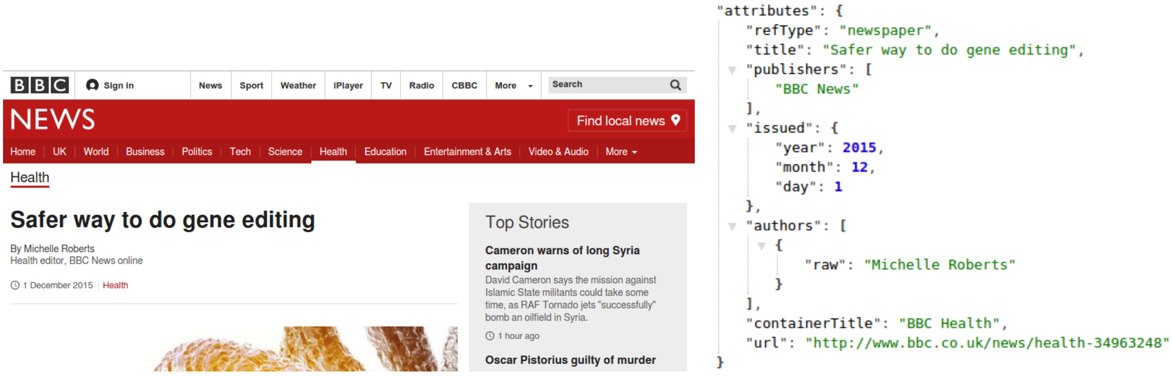
This thesis focuses on the case, where a web resource needs to be referenced. One would have to open the website and find the relevant information, such as the title, author, date of publication and publisher, needed to reference this specific resource. In the scope of this work, this information will be called *citation data*. The aforementioned citation generators automate this process, so that one only needs to enter a URL. The software tries to extract data required to cite this article.

For example, to reference a news article on BBC online, the citation generator 1) would open the URL, 2) extract the title, author, date of publication and publisher, and 3) finally return a properly formatted reference according to the selected style.

In my work for RefME, a start-up creating a citation management platform, I created and enhanced the underlying *scraper service*, which, given a URL, fetches the HTML code of that website and, based on hard-coded identifiers, extracts citation data. This method is very sensitive to the way a website's code is written, which inspired a different approach.

Based on the experience gathered, it seems that using visual cues might improve the precision of the citation data extraction. The primary advantage is, that those are independent

[1] `https://www.citethisforme.com/` [2] `http://www.easybib.com/` [3] `http://www.refme.com/`



Safer way to do gene editing

By Michelle Roberts
Health editor, BBC News online

1 December 2015 Health

Top Stories

**Cameron warns of long Syria campaign**

David Cameron says the mission against Islamic State militants could take some time, as RAF Tornado jets "successfully" bomb an oilfield in Syria.

1 hour ago

**Oscar Pistorius guilty of murder**

```
"attributes": {
    "refType": "newspaper",
    "title": "Safer way to do gene editing",
    "publishers": [
        "BBC News"
    ],
    "issued": {
        "year": 2015,
        "month": 12,
        "day": 1
    },
    "authors": [
        {
            "raw": "Michelle Roberts"
        }
    ],
    "containerTitle": "BBC Health",
    "url": "http://www.bbc.co.uk/news/health-34963248"
}
```

(a) Screenshot of a BBC article

(b) Extracted citation data

Figure 1: Example for citation data extraction of a BBC article

from the mark-up structure or identifiers. For example, incorporating the position and font size or colour of an element, relative to the rest of the website, could increase the accuracy of the extraction process.

## 2 Problem description

In the thesis, a machine learning model will be designed, implemented and evaluated to fulfil the information extraction task mentioned before. As an example, figure 1 shows the desired result of the extraction task for a BBC news article.

This model will be based on the study of the hypothesis, that visual cues yield useful features to label and extract relevant elements from websites. Relevant visual cues along with other features have to be found and evaluated with the objective of improving the precision for arbitrary domains over the previously mentioned current approach. The challenge of this thesis would be to find and compare meaningful features that can be used to train a model.

## 3 Related Work

Related to this work is research on information extraction using machine learning.

Whereas this thesis focuses on extracting citation data from websites, the GROBID project [1] aims to do just that for scholarly articles in the form of PDFs. This open source software implements an approach suggested by Peng et al. [2], that utilises layout information and syntactical features to train Conditional Random Fields *(CRF)*, and extends the Wapiti toolkit [3]. The work at hand is inspired by that approach by extending the usage of visual cues.

Lipinski et al. compared different machine learning approaches for header extraction and found that CRFs, as used in GROBID, had the best accuracy [4]. Those results are backed by similar findings in the previously mentioned paper by Peng, reaching an accuracy of just over 98% for randomly selected scholarly articles, partly from CiteSeer data [5] and the Cora project [6].

These approaches used linear chain CRFs, which assume a linear input sequence. A research by Microsoft proposed 2D-CRFs to represent the neighbouring relations of elements on the two dimensional plane of a rendered website [7].

Documents in the web, written in a structured mark-up (HTML), enable the development of different approaches, since they are not as loosely represented as PDFs. Cai et al. [8] found a promising approach to extract the semantical structure of a webpage, which led to more specific applications. For example content extraction [9], which makes it possible to hide all menus and sidebars to focus on the actual content of the page.

Research on information extraction from websites often focuses on only creating so-called wrappers, that are domain specific. A wrapper is considered to be a procedure that extracts or labels data for similar pages, for instance the title and author for all news articles on the BBC website. Ferrara et al. [10] provide a comprehensive overview of literature and general techniques in the field of web data extraction. These mainly aim towards (semi-) automatically generating or maintaining domain specific wrappers for a set of websites.

## 4 Creating a Gold Standard

In order to train a machine and evaluate the model, a set of annotated websites is needed. In this work the annotation of a website is considered to consist of the title, author and the issued date (as formatted on the website and in a normalised format).

To create a representative real-world dataset of websites that students and academics actually reference in their essays or papers, RefME offered access to their database. The selection is based on URLs that were saved in projects within a two week period.

This set needed to be scaled down to a manageable size. It was assumed, that websites within the same domain are served by a content management system and therefore use the same template. Stratified sampling was used to select 1000 URLs with a wide variety of domains but still representing their relative frequency [11]. Those were already annotated by hand, saved and parsed into a useful structured representation.

For further large scale evaluation of the trained model, the annotated set could be used to calculate the accuracy of bibliographical information that RefME's users entered. Obviously that is not part of the gold standard, but this way the precision of the model could be measured by applying it to thousands of websites. Based on the accuracy of the user data, a confidence factor needs to be applied to the extended results.

## 5 Machine Learning Approach

The approach implemented by the GROBID project will be considered as a rough guideline for this thesis. Obviously key components have to be reconsidered or even handled completely different. The processing chain will logically be separated into three parts as described in more detail below.

In the first part the website is preprocessed using PhantomJS[4], a headless browser with a JavaScript API. This way it is possible to get a rendered representation of the website including all the relevant information for visual features like fonts and the position of elements. The objective of this part is to split the website into blocks and enrich those with styling information (like position, size, font, etc).

It will be subject to experiments to run so-called boilerplate removal algorithms in order to get rid of irrelevant blocks. However, this might not be necessary, since the trained model could already consider those sections of the website irrelevant anyway. Furthermore, applying such an algorithm yields the chance of accidentally removing relevant information.

During the training phase, annotation information is represented as part of the enriched information. Peng et al used a series of syntactical and language features. The only layout information used was the text alignment. However, as mentioned before, their work was done in the context of scholarly articles (PDFs), where the citation data is usually contained within the first page, having a fairly similar layout following a few globally accepted rules.

Further features need to be introduced. For example positioning information, font size and weight or margins/paddings. Apart from those visual features, it seems reasonable to consider a lexicon based feature, which tries to match the class of a block to a list of known (i.e. title) class names.

The great advantage of CRFs is, that neighbouring elements are taken into account while classifying. Since a linear chain CRF expects a linear sequence as an input, the importance of the processing chain, that generates this sequence has to be emphasised. The raw input is either a two dimensional plane, when the page is rendered, or a hierarchical structure formed by the HTML document.

The naive approach is to flatten the structure based on the order of appearance of elements in the HTML document. In a more sophisticated approach the rendered layout could imply the order of blocks in the sequence. Either way, the objective should be to keep blocks as close together in the sequence, as they are in the rendered layout. Alternatively, the neighbourhoods could be modelled in a graph to be used for a general CRF (or more specifically in a 2D-CRF).

## 6 Evaluation

To evaluate the trained model several experiments will be conducted. All results will be averaged over multiple runs, while each run uses a different subset of the gold standard for training and testing (based on cross-validation). A subset of the gold standard will not be used in any training or testing until the final evaluation of the best model. Typical measures are used to assess the quality of assigned labels, like precision, recall and F1-Score.

Each extraction task for a website will be validated based on both the complete accuracy (all information for one website fully extracted) and the by-word accuracy (how many of the

[4] `http://phantomjs.org/`

words were correctly labelled). Thus a more sensible evaluation is possible, that is soften the penalty for slight discrepancies (i.e. a missed or additional word in a title).

In case it turns out that results heavily differ, it might be interesting to group the websites of the gold standard into categories like *news*, *blog* or *company website* and compare the independent results for each of those.

Further experiments will focus on more in-depth performances. For example the impact of features towards the result. This is especially interesting with regard to layout based features to highlight the subject of this thesis: visual cues in data extraction.

Since CRFs are used in this thesis, the effect of the order (number of transitions) should be analysed. Another important influence can be made with regularisation by penalising previous distributions of features (i.e. Gaussian/Laplacian Prior) [12]. Since that should be implemented anyway as it prevents over-fitting, tests should be made to find the best approach/parameters.

## References

[1] GROBID GitHub project page. https://github.com/kermitt2/grobid. Accessed: 2015-12-03.

[2] Fuchun Peng and Andrew McCallum. Accurate information extraction from research papers using conditional random fields. *Information Processing Management*, 42(4):963–979, 2006.

[3] Thomas Lavergne, Olivier Cappé, and François Yvon. Practical very large scale CRFs. In *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 504–513. Association for Computational Linguistics, July 2010.

[4] Mario Lipinski, Kevin Yao, Corinna Breitinger, Joeran Beel, and Bela Gipp. Evaluation of header metadata extraction approaches and tools for scientific pdf documents. 2013.

[5] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000.

[6] Kristie Seymore, Andrew McCallum, and Roni Rosenfeld. Learning hidden markov model structure for information extraction. In *AAAI-99 Workshop on Machine Learning for Information Extraction*, pages 37–42, 1999.

[7] Jun Zhu, Zaiqing Nie, Ji-Rong Wen, Bo Zhang, and Wei-Ying Ma. 2d conditional random fields for web information extraction. In *Proceedings of the 22nd international conference on Machine learning*, pages 1044–1051. ACM, 2005.

[8] Deng Cai, Shipeng Yu, Ji-Rong Wen, and Wei-Ying Ma. Extracting content structure for web pages based on visual representation. In *Web Technologies and Applications*, pages 406–417. Springer, 2003.

[9] Dandan Song, Fei Sun, and Lejian Liao. A hybrid approach for content extraction with text density and visual importance of dom nodes. *Knowledge and Information Systems*, 42(1):75–96, 2015.

[10] Emilio Ferrara, Pasquale De Meo, Giacomo Fiumara, and Robert Baumgartner. Web data extraction, applications and techniques: A survey. *Knowledge-Based Systems*, 70:301–323, 2014.

[11] Carl-Erik Särndal, Bengt Swensson, and Jan Wretman. *Model assisted survey sampling*, chapter Stratified Sampling, pages 100–109. Springer Science & Business Media, New York, 2003.

[12] Fuchun Peng and Andrew McCallum. Information extraction from research papers using conditional random fields. *Information processing & management*, 42(4):963–979, 2006.