Web Mining Framework: Croatian Patents Case Study

Goran Popović
B2match GmbH, Croatia

Abstract

Patents are one of the most valuable sources of technical and commercial knowledge. Although patents are public and can be easily searched on the Web, for most countries, there is no easy way to download bulk patent data. The purpose of this paper is to create a web mining framework used to extract patent data from the Croatian State Intellectual Property Office. Even though framework was created for the purposes of extracting Croatian patents, it can be reused for other web mining cases. The architecture of the proposed framework combines the use of web crawler and big data tools, in order to provide a complete and flexible solution for building general-purpose web mining application. The biggest limitation of this framework comes from the programming knowledge required to implement it. Therefore, this framework is only available to the small number of researchers. Data extracted with web mining methods is only as good as the algorithm used to extract data. Nevertheless, the data from official sources should always be preferred to the one retrieved using web mining methods.

Keywords: web mining, data mining, crawling, patent mining
JEL classification: L86, O34

Introduction

The patent system is the most prolific and up-to-date source of information on applied technology (EC, EPO, 2007). Patents contain important research data that can be used to show technological details and relations, reveal business trends, inspire novel industrial solutions or help make investment policy (Tsenga et al., 2007).Data retrieved from the patents often cannot be found anywhere else. According to EPO estimates, up to 80% of the current technical knowledge can be found only in patent documents.

Although patents are, by their nature, public documents, until recently it was hard and costly to retrieve a large number of patents for the research purposes. Only since 2010, the US Patent and Trademark Office (SUPTO) has offered its patents online through a bulk download service (https://bulkdata.uspto.gov). Even though all US patents are now available for download, some countries, such as Croatia have still not provided a way to download all patent data for research purposes. Also, the barrier to entry patent processing remains high because of large costs incurred by computer resources required to process millions of patents.

The State Intellectual Property Office of the Republic of Croatia is a government agency responsible for the registration of patents, trademarks and design in Croatia. Intellectual property office offers the ability to freely search and download patent data but does not offer a way to export all patents in a way that US Patent and Trademark Office and European Patent office do. Even though the State Intellectual Property Office of Croatia does not provide a way to export all patents, they can be retrieved using web mining methods.

Web mining is the use of data mining techniques to automatically discover and extract information from web documents and services (Etzioni, 1996). According to Arotaritei and S. Mitra, web mining can be broadly categorized as: a) web usage mining typically generated by user's interaction with the web, b) web structure mining of inter-document links, provided as a graph of links in a site or between sites and c) web content mining of multimedia documents, involving text, hypertext, images, audio and video information. Since the patent data is searchable and displayed on a web page, using web content mining methods, patent data can be extracted and saved to the database. Content web mining consists of two main tasks: 1) download relevant web pages using crawlers, 2) process web pages using popular processing frameworks such as Hadoop tools (MapReduce technique) or the new Google Dataflow service.

A web crawler, robot or spider is a program or suite of programs that is capable of iteratively and automatically downloading web pages, extracting URLs from their HTML and fetching them(Chun, 1999). MapReduce is a programming model and an associated implementation for processing and generating large data sets (Dean and Ghemawat, 2004). Users specify a map function that processes a key/value pair to generate a set of intermediate key/value pairs and a reduce function that merges all intermediate values associated with the same intermediate key (Dean and Ghemawat, 2010).

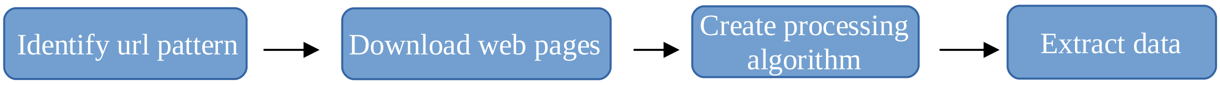
The purpose of this paper is to present a content web mining approach used to extract patent data from The State Intellectual Property Office of the Republic of Croatia. There are four parts in this paper. The framework is presented in the second part and validated through a case study reported in the third part. Finally, the last part concludes this paper.

Web mining framework

The aim of the proposed framework is the extraction of large amounts of data from web sites. The framework was proposed based on multiple years of personal experience in web mining and data processing. Some of the available technologies used in this paper are: Python and Go programming languages, Google BigQuery storage and Google Dataflow service. Google's Go language is a compiled, statically typed language with garbage collection, limited structural typing, memory safety features and CSP-style concurrent programming features added (Metz, 2011). Go is used for building a high performance concurrent crawler responsible for downloading thousands of web pages. Python is a high-level programming language that over the last two decades has grown very popular in the scientific computing community (Mortensen and Langtangen, 2016). Python is used as a processing language inside Google Dataflow Big Data service. Google Dataflow is a fully-managed cloud service and programming model for batch and streaming big data processing (Google, 2016). Google BigQuery is used for the storage purposes. BigQuery is a scalable, interactive ad hoc query system for analysis of read-only nested data (Melnik et al., 2010).

Framework consists of the following steps: 1) identify URL pattern for the relevant web pages, 2) download web pages based on the previously recognized URL pattern, 3) extract fields and structure from the web pages and create processing algorithm, 4) process web pages and extract useful structured data. The whole process can be seen on Figure 1.

Figure 1
Web content mining framework



Source: Author

In the first step, the web site is being analysed for the purposes of finding the URL pattern which can be used to download targeted entity pages containing wanted data. In the most cases that can be achieved by making several random searches and looking for the changes in the entity URLs. After the pattern is discovered, downloading algorithm is built in to the crawler.

In the next step, crawler is used to download relevant web pages based on the previously discovered URL pattern. Web pages are being stored to the Google BigQuery service. Other storages, such as Google Cloud Storage and Amazon S3 Storage can also be used.

The complexity of the required processing algorithm depends on the types of field and data stored in the web pages. In most of the cases, each entity will contain all fields, therefore allowing us to easily test and create a processing algorithm after analysing just a single entity. In some cases, different entities will have different type of field which requires creating an algorithm for the purposes of finding unique fields and types of data. When all fields and types of data are recognized, a main processing algorithm used for extracting data can be created. The algorithm is written in Python, but can be written also in Java or any other programming language supported by big data tools.

In the last step, previously made processing algorithm is used to extract relevant data from the web pages using Dataflow service. Once again, processed data is saved to the distributed storage, Google BigQuery.

Case Study

Web site of The State Intellectual Property Office of the Republic of Croatia contains thousands of Croatian patent data. Searching and downloading patent data is allowed but there is no way to export all patents. To download all patents, previously presented web mining framework is used.

In the first step, Intellectual Property Office web site is analysed with the purposes of finding out the URL pattern used to download patent web pages. After several random searches, it can be observed that all patent web page URLs have this form "http://it-app.dziv.hr/Pretrage/hr/p/Detaljno.aspx/{ID}". It can also be noticed that the ID of all checked patents consists of 8 numbers. Without going into further optimisation, downloading all IDs would be long and costly since there are 99.999.999 combinations. After more analysing, a pattern emerged. Patent page ID consists of year(4 digits) and ID(left padded 4 digits). For an example, patent number 1 in year 1996 would have an ID: 19960001, patent number 55 in year 2015 would have an ID 20150055 and so on. This means that each year has at most 9999 patents and since Croatian intellectual property office displays only patents since the year 1992 only 249.975 web pages will need to be downloaded. Compared to the initial number of combinations this comes down to only 0.25%. In cases like this, with a large number of possible ID combinations it is crucial to narrow down the numbers as much as possible in order to speed up the process and lower its costs.

After identifying the URL pattern of the patent web pages, the custom crawler is written in Go programming language. Responsibility of the crawler is to download and save all web pages based on an algorithm that can be seen on Figure 2. The crawler will save only those web pages for which the web server returned status code 200 indicating that the content exists. This does not necessarily mean that the patent exists on that URL, only that the web server returned a valid response.

Figure 2
Pseudo code of downloading and saving patent web pages

```
for year in range(1992, 2016):
    for id in range(1, 9999):
        url = “http://it-app.dziv.hr/Pretrage/hr/p/Detaljno.aspx/” + year + leftpad(id, 4, “0”)
        page = http.fetch(url)
        if page.status_code == 200:
            storage.save(page)
```

Source: Author

On the 23rd of June, 2016, crawler was deployed and successfully started downloading patent web pages using algorithm that can be seen on Figure 2. Almost 250.000 http requests were sent, of which 18.203 received web responses were valid and saved to the BigQuery storage. Combined, 293 MB of html data including headers was saved.

Table 1
Crawler statistics

Number of http requests	249.975
Number of saved web pages	18.203
Storage size	293 MB
Average web page size	16 KB

Source: Author

The third step consists of working out the patent structure to be able to create algorithm used to extract data from the web pages. Since there are multiple types of patent data, each entity can have different fields. Therefore, to be able to recognize a patent structure and create a processing algorithm all fields and types of value need to be extracted from the downloaded web pages. The algorithm can be seen in Figure 3. For every web page stored by the crawler, all unique fields are extracted. After all unique fields from all web pages are extracted, the system is randomly taking two patents for each unique fields.

Figure 3
Pseudo code of field extraction

```
  # extract all fields from the patent web pages
  for webpage in patent_webpages:
      fields = webpage.xpath(“//table/field”)
      for field in fields:
          yield (field, webpage.id)

   # group by field and return 2 random patents for each field
   sample = fields.fixedSizePerKey(2)

storage.save(sample)
```

Source: Author

The extracted fields and patent samples are then used to create the algorithm used to process html pages and extract relevant patent data. After processing all downloaded web pages, 28 different types of field were found.

In the last step, once again Google Dataflow service is being used, this time to finally process and extract patent data. Simple map and parse function is being used which can be seen in Figure 4. Parsed data is saved to the BigQuery storage.

Figure 4
Pseudo code of patent data extraction

```
  for webpage in patent_webpages:
     patent = parse_patent(webpage)
storage.save(patent)
```

Source: Author

Out of the 18.203 downloaded web pages, 18.201 patents were extracted taking 20.1 MB of space or just 6.86% of raw downloaded data.

Conclusion

This paper has shown that even though The State Intellectual Property Office did not provide a way to download bulk patent data, patent data can still be retrieved using web mining methods. The framework presented in this paper can be reused for similar cases. Even though patent data can be retrieved using data mining methods, researchers need to be careful because quality and correctness of extracted data depends on the quality of the used processing algorithm. Therefore if possible, researches should always use the official data provided by registers. Biggest limitation of the presented data mining framework is the required knowledge of programming. Because of that, web mining will be limited to only researchers with fairly good programming knowledge.

References

1. Arotariteia, D., Mitrab, S. (2004), “Web Mining: a Survey in the Fuzzy Framework”, Fuzzy Sets and Systems, Vol. 148, pp. 5-19.

2. Chun, T.Y., (1999), "World Wide Web Robots: an Overview", Online & CD-ROM Review 23, Vol. 3, pp. 135-142.
3. Dean, J., Ghemawat, S. (2004), "MapReduce: Simplified Data Processing on Large Clusters", Communications of the ACM - 50th anniversary issue: 1958 – 2008, Vol. 51 No.1, pp. 107-113.
4. EC, EPO, (2007), "Why Researchers Should Care about Patents", European Commission and European Patent Office.
5. Etzioni, O., (1996), "The World Wide Web: Quagmire or Gold Mine", Communications of the ACM, Vol. 39, pp. 65-68.
6. Melnik, S., et al., (2010) "Dremel: Interactive Analysis of Web-Scale Datasets", Proc. Of the 36th International Conference on Very Large Data Bases, pp. 330-339.
7. Metz, C. (2011), "Google Go Boldly Goes Where No Code Has Gone Before", available at: http://www.theregister.co.uk/2011/05/05/google_go/ (06/08/2016)
8. Mortensen, M., Langtangen H. P. (2016), "High performance Python for direct numerical simulations of turbulent flows".
9. Tsenga, Y., et al., (2007), "Text mining techniques for patent analysis", Information Processing and Management, Vol. 43, pp. 1216-1247.

About the author

Goran Popović is currently employed as a programmer at B2match GmbH in Vienna, Austria. He received his bachelor's degree in economics (major Business Economics) in 2013 and graduated (major Managerial Informatics) in 2014 at the Faculty of Economics & Business – Zagreb where he is currently enrolled in a doctoral program. His interests include data mining, cloud infrastructure, big data, databases and distributed systems. The author can be contacted at **goranpopovic@gmail.com.**