Web Scraping: Applications in Infrastructure Planning

Rafid Morshedi
Data Analytics and Automation Engineer, WSP
Rafid.Morshedi@wsp.com

Ben Chu
Graduate Engineer, WSP
Ben.Chu@wsp.com

Eric Huang
Graduate Data Scientist and Engineer, WSP
Eric.Huang@wsp.com

Lucy Ivers
Graduate Engineer, WSP
Lucy.Ivers@wsp.com

ABSTRACT

There is an abundance of data available to the modern engineer tasked with planning new infrastructure. However, this data is not always in a readily accessible format or may be scattered across various locations. Furthermore, the volume of data is often too large to be processed manually in a timely manner. When planning major infrastructure, it is critical to understand potential impacts from 3rd party developments that may impact upon the planned works. In New South Wales this information is in the public domain but scattered across various local and state government websites which must be manually searched (often by junior staff) to detect 3rd party developments that present a risk to the alignment. This paper describes how this process was automated for a railway project in Sydney. Web scraping bots were written to mine pertinent development application information from government websites, which were then automatically risk-rated using natural language processing and machine learning methods. In order to mine the information effectively and geolocate the data, several publicly available spatial datasets were integrated, including lot and administrative boundary information from the Digital Cadastral Database (DCDB) combined with the Geocoded-National Address File (G-NAF). Many machine learning techniques were tested and ultimately a boosted tree algorithm (XGBoost) was used to build a model that allocated a preliminary risk rating to development applications. The automated system drastically reduced the time taken to search through development applications. The speed allowed for high-risk developments to be identified as they were submitted for planning approvals and new assessments could be made rapidly upon changes to the alignment. The use of the human mind was then targeted towards verification.

KEYWORDS: *Web scraping, machine learning, natural language processing, development applications, G-NAF, DCDB.*

1 INTRODUCTION

There is a large amount of publicly available information that needs to be analysed to construct new infrastructure. Development Applications (DAs) are one such piece of publicly available information. During the planning and design phase of new infrastructure, particularly

underground infrastructure, it is required that designers are aware of any developments that are planned near the new piece of infrastructure. In the case of public transport infrastructure, this may present a significant financial, legal and program risk to the project if land take is required. From an engineering context, these developments can place constraints on underground infrastructure. Thus, it is necessary to constantly track new DAs.

Development Applications must be published publicly in New South Wales under the requirements of the Environmental Planning and Assessment Act 1979. Most local government and state agencies publish this information on their respective websites. Planners and designers must manually click through hundreds of webpages to filter through DAs and flag any developments that may pose a risk to a project. This takes a significant amount of time and due to the continual submission of new applications is prone to omissions.

This paper describes a process developed to automate this process. The process automatically mines the information from the various government websites and then uses machine learning methods to automatically classify developments as high or low risk. To do this, a large number of disparate datasets need to be joined in order to mine the data effectively and produce meaningful results.

2 PROCESS

2.1 Overview

An overview of the process the authors have designed is shown in Figure 1. The process can be separated into four main components, which are described in detail here.

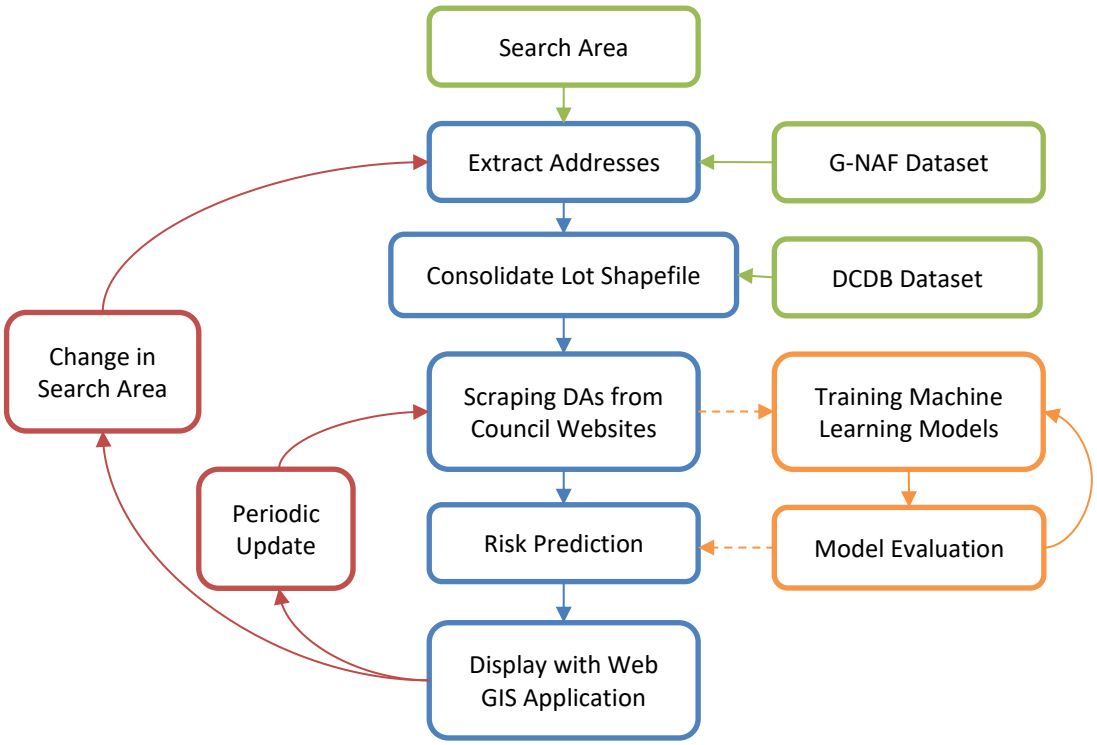

Figure 1: Flowchart of the process.

Firstly, addresses within the area of interest are extracted. These are then used to search for Development Applications from council websites. A machine learning algorithm predicts the risk rating of each application, and the resulting information is displayed on a web-based Geographic Information System (GIS) application. This process can be repeated for a periodic update or when a change has occurred in the search area.

2.2 Scraping

Scraping is the process used to locate and retrieve DA data from council websites: addresses are searched for on the DA trackers and relevant information is saved. The council websites that are scraped are served in either Hypertext Markup Language (HTML) or embedded in a JavaScript. These two require different scraping approaches, as described below.

The Geocoded-National Address File (G-NAF) dataset (PSMA, 2018) is used to extract addresses. This data is used in conjunction with the Digital Cadastral Database (DCDB), which contains property shapes and boundaries.

2.2.1 Scraping HTML Pages

HTML is the standard mark-up language for creating webpages (W3C, 2017). Elements on a HTML page are presented in a hierarchical order and each has a unique tag. These tags are used by a HTML parser, such as *Beautiful Soup* (Richardson, 2019), to accurately locate specific elements of interest on a webpage. These elements can then be recorded for further processing.

It should be noted that some council DA tracker websites have a very similar architecture. However, due to slight variations, a slightly different scraper had to be used for each website created using HTML.

2.2.2 Scraping JavaScript Pages

JavaScript is another core technology used to power webpages. Pages embedded with JavaScript are rendered as the page is loaded and are interactive programs on a webpage, which makes it harder to scrape compared to HTML. The Selenium WebDriver (Selenium, 2019) has been used to automate the process of interacting with the JavaScript and subsequently retrieving DA information.

2.3 Risk Prediction

An integral part of this process is the prediction of risk ratings, which utilises Natural Language Processing (NLP) and machine learning. NLP techniques are used to extract features from written text in scraped DAs for input into machine learning algorithms. Features are measurable variables of each item that is being analysed, and in this case they are words from each DA.

The process of feature extraction utilised by the authors is described as follows. Firstly, text cleaning is performed, where all non-alphanumeric characters are replaced with whitespaces and all capitalised letters replaced with their lower-case counterparts. Secondly, stop words are removed. Stop words are commonly used words that carry no significant information for our purposes, such as *a*, *the* and *are*. In the next step, lemmatisation is performed, which groups together different forms of a word, such as *organise*, *organises* and *organising*. There are also words with similar origin and meaning, such as *democracy*, *democratic* and *democratisation* which are also lemmatised into a single feature. From here, n-grams are extracted, and their

frequencies counted. An n-gram is a sequence of *n* consecutive words found in a corpus. The authors have chosen to extract 1- and 2-grams for their work.

Several machine learning models, including naïve Bayes, neural network, k-nearest neighbour and Extreme Gradient Boosting (XGBoost) (Chen and Guestrin, 2016), were trained and tested using the extracted features. XGBoost was chosen for the final implementation of the process due to its superior performance over the other models.

2.4 Display

The information that has been obtained is then displayed using a GIS application for ease of viewing. Lots with DAs rated as high risk are highlighted as bright red, while low-risk lots are shown as green. This allows engineers tasked with planning major infrastructure projects access to regularly updated information.

3 RESULTS AND DISCUSSION

The combination of web scraping and machine learning led to significant savings in the time taken to analyse and assess new Development Applications. Firstly, new DAs could be monitored much more regularly, and the process was less prone to human errors.

The consolidation of information into a single GIS platform reduced the cognitive load on designers and planners looking to identify impacts on transport infrastructure. The authors could scrape all the websites of authorities whose jurisdictions intersected the corridor for the planned transport infrastructure.

It is important to note that humans were not removed from the process but were rather used in a verification and validation role. The risk ratings produced by the machine learning algorithms are checked by engineers and planners with the results being fed back into the machine learning training process to improve future risk predictions. It was found that humans were much more effective in this role than in manually checking all the DAs.

As a side benefit, the collation of DAs allowed engineers from other disciplines, such as transport planning, to rapidly find developments near the alignment that could have traffic impacts on the local area.

3.1 Performance of Risk Predictor

The performance of the machine learning risk predictor is presented here, as it is integral to the usefulness of the process. By leveraging the prediction threshold, performance metrics such as accuracy and recall can be optimised according to the use case. Accuracy measures the ratio of correct predictions out of all predictions and recall measures the ratio of correctly recognised high-risk DAs out of all high-risk DAs. In the authors' application, the optimisation goal is to maximise recall while retaining an acceptable trade-off to accuracy (Figure 2). A recall of 0.95 and accuracy of 0.86 was chosen by the authors.

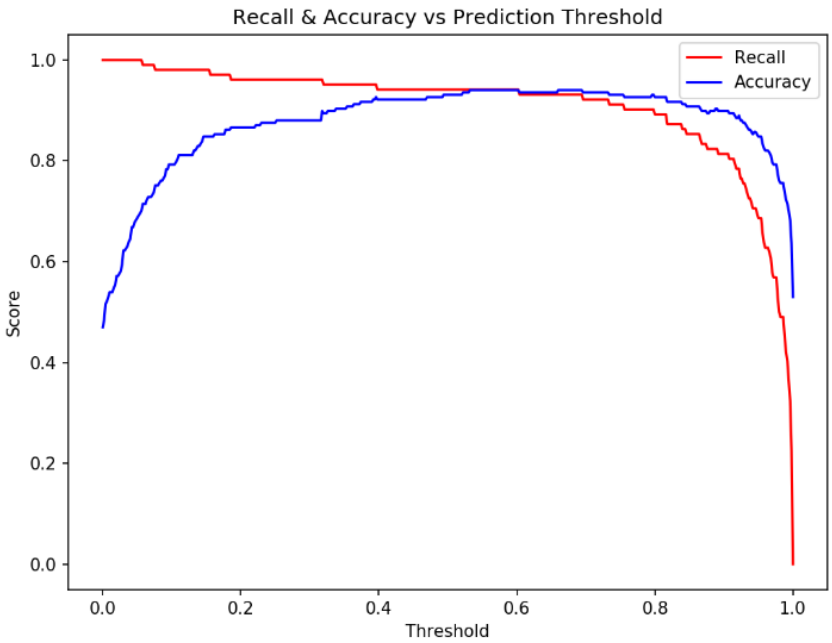

Figure 2: Trade-off between recall and accuracy.

4 DRAWBACKS

It should be noted that there are several drawbacks of the process presented in this paper.

4.1 Currency of Datasets

The G-NAF and DCDB datasets are utilised to identify DAs of interest and present these to the user. Where the G-NAF dataset is out of date, DAs with new addresses within the search area will not be identified and therefore missed. The G-NAF dataset has a quarterly update frequency, which means that it is outdated by one and a half months on average. Due to the nature of DAs, they are also more prone to having an address that is not listed on the G-NAF. Where the DCDB dataset is out of date, older lot shapes are rendered, which decreases the value of the information. Possible solutions include utilising online datasets, increasing the update frequency of datasets, and geocoding DAs such that address extraction is no longer a necessary step.

4.2 Quality of Description

The process is dependent on the description of the DA in predicting the risk rating. The machine learned algorithm can only perform if the description accurately describes the development. This problem could be overcome by including more relevant information, such as estimated costs (this is currently only shown for some councils) and estimated time.

4.3 Council Website

The process is dependent on some aspects of the council websites not changing, and as such requires close scrutiny to identify changes that occur. The changes are, by their nature, not advertised beforehand. Appropriate adjustments to the scraper must be made should changes occur.

4.4 Discrepancies Between Datasets

During the development of this automated process, the authors have noticed that there are discrepancies between the G-NAF and DCDB datasets. Lots exist in the DCDB dataset that do not have a corresponding address on G-NAF, which means that these lots are not searched for. There are also inaccuracies between the two datasets causing DA information to fall into another lot in close proximity to where they are.

5 CONCLUDING REMARKS

The combination of web scraping and machine learning techniques proved to be an effective means of conducting preliminary analysis of Development Application data for assessing 3rd party risk to planned infrastructure. Such techniques were enabled by readily available foundational datasets such as G-NAF and DCDB, which were fundamental in adding context to the data that was scraped. Without these datasets the entire process would have been heavily flawed.

The development and use of such techniques represents a new approach to risk assessments for infrastructure planning and allows for rapid appraisals of new options. Moreover, it highlights how digital tools may be used to analyse the increasingly large datasets that are being made available to engineers, surveyors and planners. These datasets can be analysed even if the datasets are not made available in a structured format.

ACKNOWLEDGEMENTS

The work described in this paper was funded by WSP and would not have been possible without the support of Sam McWilliam who came up with the initial idea and Stuart Allabush who supported us throughout the development process and continues to support us as the software is being improved.

This work would not be possible without foundational datasets such as G-NAF and the DCDB being made freely available and for that we would like to thank PSMA and the NSW Department of Finance, Services and Innovation.

REFERENCES

Chen T. and Guestrin C. (2016) XGBoost: A scalable tree boosting system, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, 13-17 August, 785-794.

PSMA (2018) Geocoded National Address File (G-NAF), May 2018 version, https://data.gov.au/dataset/19432f89-dc3a-4ef3-b943-5326ef1dbecc (accessed May 2018).

Richardson L. (2019) Beautiful Soup, https://www.crummy.com/software/BeautifulSoup/ (accessed March 2019).

Selenium (2019) Selenium WebDriver, https://www.seleniumhq.org/projects/webdriver/ (accessed March 2019).

W3C (2017) HTML 5.2, https://www.w3.org/TR/html52/ (accessed March 2019).