Modernizing Census Bureau Economic Statistics through Web Scraping

Brian Dumbacher[1], Carma Hogue[1]
Brian.Dumbacher@census.gov, Carma.Ray.Hogue@census.gov
[1]U.S. Census Bureau, 4600 Silver Hill Road, Washington, DC 20233

Abstract
For economic surveys conducted by the U.S. Census Bureau, useful data such as respondent or equivalent-quality data can sometimes be found online. The Census Bureau is researching the use of web scraping public sites to improve existing economic survey collection and processing as well as sampling frames. We will discuss our efforts to build a tool called SABLE (Scraping Assisted by Learning), which uses a combination of web crawling, web scraping, and machine learning to discover, collect, and process data from the web. We will also describe past, current, and future Census Bureau efforts to scrape state government tax revenue data, public pension data, building permit data, and other information to enhance data relevance, reduce respondent and analyst burden, and increase the quality of sampling frames. Concerns and challenges associated with these efforts are also described.

Key Words: U.S. Census Bureau, web scraping, official statistics, economic statistics, passive data collection

1. Introduction

1.1 Big Data Context

The Economic Directorate of the U.S. Census Bureau has been researching various Big Data methodologies for improving all components of the Survey Life Cycle, or stages in the survey production process (Snijkers *et al.*, 2013, p. 132), from frame development and data collection through estimation. The Economic Directorate has used administrative data for many years, but more recently, it has examined the possible use of third-party data, system-to-system collection, and web-scraped data (Dumbacher and Hanna, 2017). Each Big Data methodology or data source has positive and negative considerations for its use. After conducting research to evaluate feasibility and quality, other factors such as cost, timeliness, transparency of methods, skillsets needed, sustainability, and risks are considered when making final decisions about what methodology or data source to adopt.

Disclaimer: Any views expressed are those of the authors and not necessarily those of the U.S. Census Bureau.

When cost is a factor in the decision-making process, third-party data acquisition often becomes infeasible. Sometimes the acquisition process itself is a hindrance to obtaining timely data although new procedures could be adopted to eliminate some of the difficulties of obtaining data sources quickly. For example, in the case of a system-to-system transfer of data from the respondent's computer system to the Census Bureau's system, the complexity of the surveys and the lack of harmonization of data across all of the Economic Directorate's appropriated surveys make it difficult to set up a system-to-system structure that will reduce respondent burden by an appreciable amount.

Instead, those working on Big Data projects in the Economic Directorate have found that web scraping and web crawling (Mitchell, 2015), machine learning (Hastie, Tibshirani, and Friedman, 2009), and Big Data methodological improvements such as "tableplots" for editing large datasets (Puts, Daas, and de Waal, 2015; Tennekes, de Jonge, and Daas, 2013) are the most beneficial methods available in the near future because of the low cost, the quality of some of the available datasets, and the skillset that the Economic Directorate is developing. This paper focuses on web scraping and web crawling and describes some of the Economic Directorate's past, current, and future efforts in this area.

1.2 Outline

The rest of the paper is organized as follows. Section 2 covers additional background information and motivation regarding web scraping. In Section 3, we present the SABLE tool (which stands for Scraping Assisted by Learning) that was developed to house the software needed to perform certain web scraping and web crawling tasks, some of which are assisted by machine learning. Sections 4 and 5 cover applications of SABLE to the Economic Directorate's public sector programs, namely scraping state government tax revenue collections and specialized pension statistics. A SABLE-related project on scraping Securities and Exchange Commission (SEC) filing metadata is covered in Section 6. Section 7 describes a Big Data project on scraping building permit data in support of construction surveys. A couple efforts to use web scraping to improve sampling frames and universes are mentioned in Section 8. Finally, in Section 9 we examine future web scraping steps including SABLE's use inside the Economic Directorate.

2. Web Scraping Background

2.1 Motivation

For many economic surveys conducted by the Economic Directorate, respondent data, equivalent-quality data, and relevant administrative records can sometimes be found online. Useful online data sources include respondent websites, Comprehensive Annual Financial Reports (CAFRs) and other publications on state and local government websites, public filings with the SEC, and Application Programming Interfaces (APIs). Going directly to these types of online sources and collecting the data passively and in a manner much more automated than currently done has the potential to reduce respondent and analyst burden, save costs, and enhance the efficiency of data collection operations while maintaining the quality of data products.

Web scraping is the process of collecting data from online sources automatically. It involves finding and extracting data and contextual information from web pages and documents. In order to scrape data from some documents, such as ones in Portable Document Format (PDF), they might first have to be converted to a format that is easier to analyze. In this paper, web scraping also includes making queries to and fetching information from APIs. Although collecting data through APIs does not involve dealing with unstructured data and unstandardized terminology that most web scraping projects typically do, it does capture the overall spirit of using online sources.

Related to web scraping is web crawling, which is the automated process of systematically visiting and reading web pages. Web crawlers, also known as spiders or bots, are typically used to build search engines but have shown to be useful in finding new data sources that support economic programs.

2.2 Policy Issues

An important policy issue regarding web scraping for official statistics deals with informed consent, i.e. whether "respondents" have to be notified that their websites are being scraped. For the Census Bureau, this as-yet unresolved policy issue basically results from ambiguity on whether informed consent is needed when crawling and scraping data from the public websites of private businesses and companies (as opposed to public sector websites such as websites of state and local governments). Currently, analysts in the Economic Directorate manually scrape data from websites without giving notice that they are doing so.

Statistics Canada, on the other hand, does inform their respondents of intentions to scrape their websites. Statistics Canada's "About us" page (Statistics Canada, 2018), which is located at https://www.statcan.gc.ca/eng/about/about, informs data users and respondents what web scraping is and that their websites could be scraped. Crawling a website is more intrusive and burdensome than scraping from specific documents and web pages. As described in greater detail in Section 3, SABLE informs a website when it has crawled and provides background information about itself.

3. Scraping Assisted by Learning (SABLE)

3.1 Overview

SABLE is a collection of tools developed by researchers in the Economic Directorate for performing three main tasks: web crawling, web scraping, and text classification. Text classification models are used to predict whether documents contain useful data and to map scraped data to standardized terminology and classification codes. Table 1 describes these tasks in greater detail (Dumbacher and Diamond, 2018). Not all three tasks may be relevant to a given application. For example, data sources may already be determined, so it may not be necessary to perform web crawling. In this case, the problem would consist of just scraping and classifying data from known websites and documents.

Table 1. Three Main Tasks Performed by SABLE

Web Crawling
• Scan websites • Discover documents • Compile a training set of documents for building classification models
Web Scraping
• Find the useful data in a document using the frequencies and locations of important word sequences • Extract numerical values and contextual information such as data labels
Text Classification
• Predict whether a document contains useful data • Map scraped data to the Census Bureau's terminology and classification codes using data labels associated with the scraped data

3.2 Software

SABLE is based on two key pieces of open-source software: Apache Nutch, which is a Java-based web crawler (Apache, 2018), and Python. To run Nutch, one supplies a list of seed URLs, or starting points, and sets crawling parameters related to politeness and depth. Politeness refers to how frequently Nutch jumps from one web page to another. Visiting pages too frequently can burden websites' servers. To avoid this, web crawlers can incorporate a delay as they crawl. Websites provide politeness parameters such as this delay as well as instructions on what directories are off limits through a file called "robots.txt." Nutch is configured to always obey "robots.txt." Depth refers to how many levels of links to follow. A deeper crawl will map a website more extensively but will take longer to run. Nutch also has filters that one can apply to limit crawling to certain website domains and file types. Nutch first visits the seed URLs and then iteratively follows links down to the specified depth. It stores information about the pages and documents it comes across such as date and time stamps and whether links are duplicates, are broken, or redirect to other URLs.

As Nutch crawls, it leaves a short description explaining SABLE's purpose: "U.S. Census Bureau research to find alternative data sources and reduce respondent burden." Typically this information would only be looked at by server administrators in the case of unusual website visit activity, but it is good practice nonetheless. Also provided to the websites is a link to the SABLE repository on the Census Bureau's publicly accessible GitHub account: https://www.github.com/uscensusbureau/SABLE. This repository currently contains documentation, two Python programs for converting PDFs to TXT format and for fitting and evaluating text classification models, supplementary files, example text data scraped from websites, and example output.

Python is used to scrape text and data from documents, process the scraped data, perform text analysis, and fit and evaluate classification models. There are three main Python modules: scikit-learn, the Natural Language Toolkit (NLTK), and PDFMiner. Scikit-learn is a commonly used machine learning module with many options for classification (Pedregosa *et al.*, 2011). NLTK is used to process and analyze text and also has some machine learning capability (Bird, 2006). The NLTK and scikit-learn modules have complementary features that make it easy to fit classification models for text. Lastly, PDFMiner converts PDFs to TXT format and is used in many SABLE applications (Shinyama, 2013).

3.3 Architecture Design

The architecture design for SABLE is fairly simple and is illustrated in Figure 1 (Dumbacher and Diamond, 2018). SABLE resides on a Linux server behind the Census Bureau's firewall and crawls and scrapes data from external public websites. Apache Nutch is self-contained and consists of the application itself, parameter files for customizing crawls, and directories for storing crawl results. The Python programs are located in a separate folder. Supplementary files consist of lists of common "stop" words that are useful for text analysis. For some problems involving PDF-to-TXT conversion and the classification of entire documents, additional folders are used to organize documents according to file format and class.

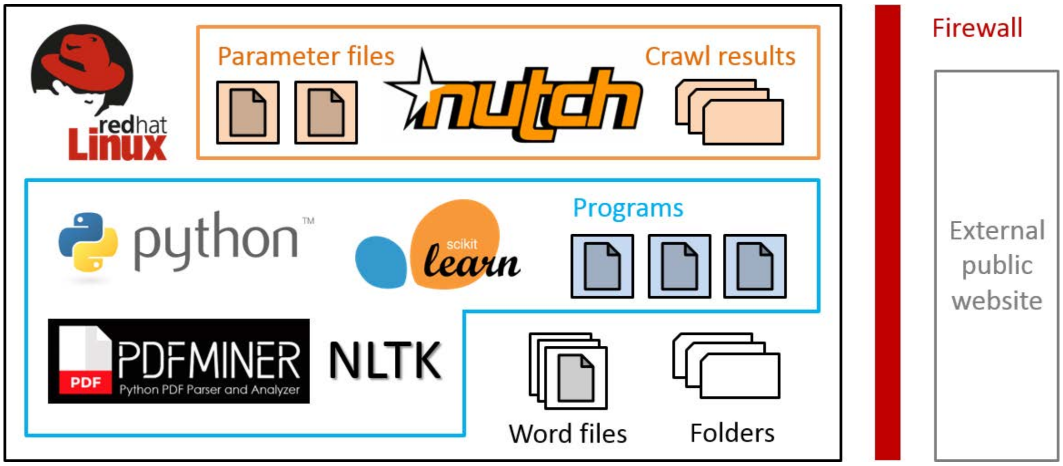
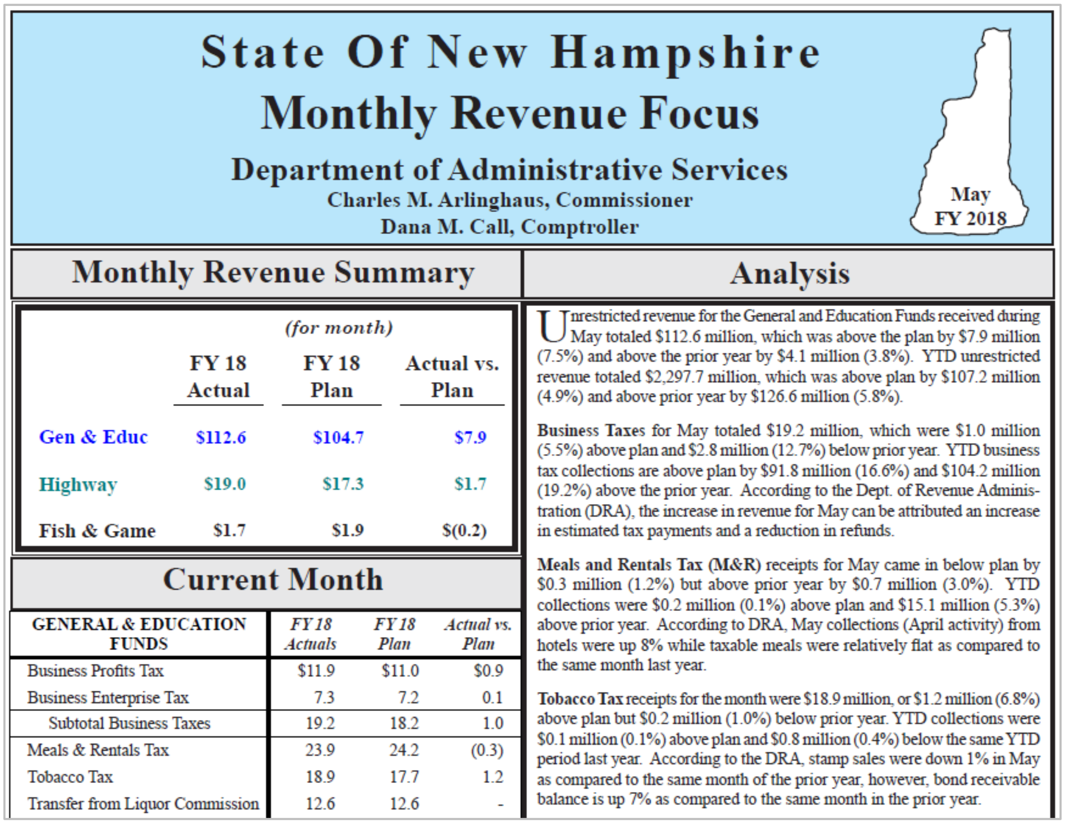


Figure 1. SABLE architecture design. SABLE resides on a Linux server behind the Census Bureau's firewall and crawls and scrapes data from external public websites. Apache Nutch and Python are the two key pieces of software.

4. State Government Tax Revenue Collections

4.1 Background

The Economic Directorate conducts surveys that collect state and local government tax data on a quarterly, annual, and quinquennial basis. The data are usually published later than data users would like to have the information. For some governments, state tax tables are available to the public monthly through statistical reports and other publications, most of which are in PDF format. Figure 2 is a screenshot of a monthly revenue report from the State of New Hampshire Department of Administrative Services website and is an example of a useful online data source. In general, some important taxes include general sales and gross receipts tax, individual income tax, and corporate net income tax. The first application of SABLE was to the Quarterly Summary of State and Local Government Tax Revenue (QTax). The goals were to crawl state government websites, discover potential new sources of tax revenue information, and build a classification model for predicting whether a PDF contains tax revenue data (Dumbacher and Capps, 2016).



State Of New Hampshire
Monthly Revenue Focus

Department of Administrative Services
Charles M. Arlinghaus, Commissioner
Dana M. Call, Comptroller

May
FY 2018

Monthly Revenue Summary

	FY 18 Actual	FY 18 Plan	Actual vs. Plan
	(for month)		
Gen & Educ	$112.6	$104.7	$7.9
Highway	$19.0	$17.3	$1.7
Fish & Game	$1.7	$1.9	$(0.2)

Current Month

GENERAL & EDUCATION FUNDS	*FY 18 Actuals*	*FY 18 Plan*	*Actual vs. Plan*
Business Profits Tax	$11.9	$11.0	$0.9
Business Enterprise Tax	7.3	7.2	0.1
Subtotal Business Taxes	19.2	18.2	1.0
Meals & Rentals Tax	23.9	24.2	(0.3)
Tobacco Tax	18.9	17.7	1.2
Transfer from Liquor Commission	12.6	12.6	-

Analysis

Unrestricted revenue for the General and Education Funds received during May totaled $112.6 million, which was above the plan by $7.9 million (7.5%) and above the prior year by $4.1 million (3.8%). YTD unrestricted revenue totaled $2,297.7 million, which was above plan by $107.2 million (4.9%) and above prior year by $126.6 million (5.8%).

Business Taxes for May totaled $19.2 million, which were $1.0 million (5.5%) above plan and $2.8 million (12.7%) below prior year. YTD business tax collections are above plan by $91.8 million (16.6%) and $104.2 million (19.2%) above the prior year. According to the Dept. of Revenue Administration (DRA), the increase in revenue for May can be attributed an increase in estimated tax payments and a reduction in refunds.

Meals and Rentals Tax (M&R) receipts for May came in below plan by $0.3 million (1.2%) but above prior year by $0.7 million (3.0%). YTD collections were $0.2 million (0.1%) above plan and $15.1 million (5.3%) above prior year. According to DRA, May collections (April activity) from hotels were up 8% while taxable meals were relatively flat as compared to the same month last year.

Tobacco Tax receipts for the month were $18.9 million, or $1.2 million (6.8%) above plan but $0.2 million (1.0%) below prior year. YTD collections were $0.1 million (0.1%) above plan and $0.8 million (0.4%) below the same YTD period last year. According to the DRA, stamp sales were down 1% in May as compared to the same month of the prior year, however, bond receivable balance is up 7% as compared to the same month in the prior year.

Figure 2. Screenshot of a monthly revenue report from the State of New Hampshire Department of Administrative Services website. Reports such as this contain useful tax revenue data. Source: https://das.nh.gov/accounting/FY%2018/Monthly_Rev_May.pdf

4.2 Methodology and Results

Researchers first created a list of seed URLs of home pages of state government departments of revenue, taxation, and finance. Nutch was used to crawl these websites to a depth of three and discovered approximately 60,000 PDFs. To create a training set for use with machine learning, a simple random sample of 6,000 PDFs was selected, where the sample size was chosen based on an estimate of how long it would take to classify the PDFs manually. Next, a PDF-to-TXT conversion algorithm based on the PDFMiner module was applied to scrape text and put it in the simple format of a single string of words separated by spaces. The text in this format could then be used as input to classification models. About 1,000 PDFs could not be converted to TXT format for various reasons. For the approximately 5,000 PDFs that could be converted, researchers manually classified them as positive (contains useful data on tax revenue collections) or negative. Lastly, these 5,000 PDFs were randomly divided into training and test sets.

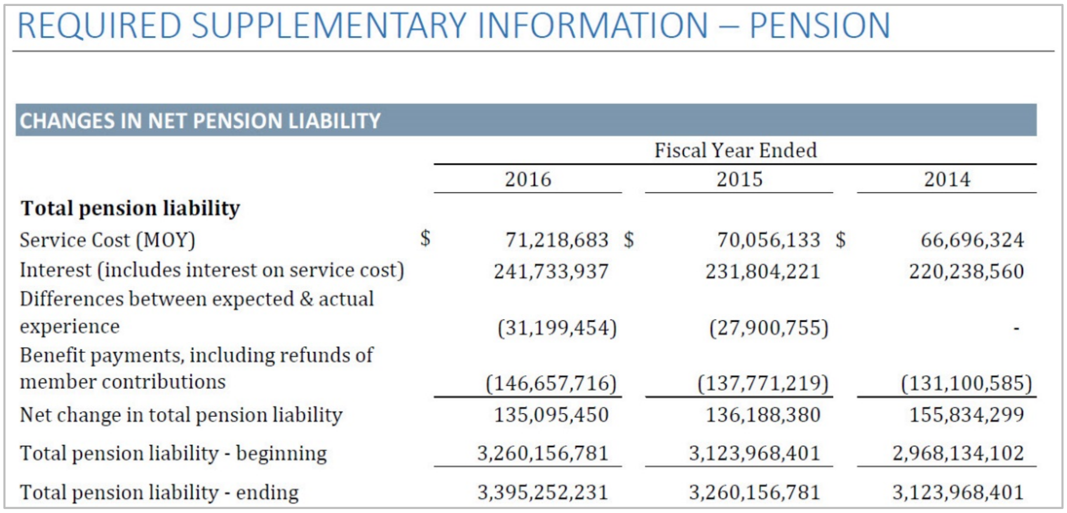
Different machine learning models using various sets of features were fit on the training set and evaluated on the test set. The best performing model achieved an accuracy of 98 percent and an F_1 score of 0.89, which is a measure that balances recall and precision (Tan, Steinbach, and Kumar, 2006, p. 297). Such a model could be used to classify future PDFs discovered through more extensive web crawling.

5. Public Pension Statistics

5.1 Background

Several of the largest state and local government pension plans are available on public sector websites in CAFRs that are readily available to the public. These CAFRs are usually in PDF format and comprise the financial report of a state, city, county, or other governmental entity that complies with Governmental Accounting Standards Board requirements (Governmental Accounting Standards Board, 2018). The majority of the public sector financial data that Census Bureau surveys request is available in a CAFR. Often, the respondent asks the Census Bureau to gather the requested data from their CAFR. Analysts are currently manually transferring the information to the questionnaires.

For state and local government pensions, researchers are examining the feasibility of scraping CAFRs for a new pensions product for use by the Bureau of Economic Analysis. Specifically, there is interest in scraping specialized content not currently collected in a Census Bureau survey from the CAFRs of the 300 largest state- and local-administered pension plans. The main pension statistics for this product are service cost and interest. Figure 3 is a screenshot of the CAFR of the Santa Barbara County Employees' Retirement System showing pension statistics for fiscal years ended June 30, 2014-2016. In general, there is no standardization in CAFRs across governments, but the pension terminology is fairly consistent across government entities and throughout time.



REQUIRED SUPPLEMENTARY INFORMATION – PENSION

CHANGES IN NET PENSION LIABILITY

	Fiscal Year Ended		
	2016	2015	2014
Total pension liability			
Service Cost (MOY)	$ 71,218,683	$ 70,056,133	$ 66,696,324
Interest (includes interest on service cost)	241,733,937	231,804,221	220,238,560
Differences between expected & actual experience	(31,199,454)	(27,900,755)	-
Benefit payments, including refunds of member contributions	(146,657,716)	(137,771,219)	(131,100,585)
Net change in total pension liability	135,095,450	136,188,380	155,834,299
Total pension liability - beginning	3,260,156,781	3,123,968,401	2,968,134,102
Total pension liability - ending	3,395,252,231	3,260,156,781	3,123,968,401

Figure 3. Screenshot of the CAFR of the Santa Barbara County Employees' Retirement System showing specialized pension statistics for fiscal years ended June 30, 2014-2016. Source: http://cosb.countyofsb.org/uploadedFiles/sbcers/benefits/SBCERS 6-30-2016 CAFR With Letters.pdf

5.2 Methodology

A two-stage approach to scraping service cost and interest is currently being considered. After converting the CAFRs from PDF to TXT format, models are applied that are based on the locations of important word sequences in order to identify tables containing the pension statistics. For example, the phrases "required supplementary information" and "changes in net pension liability" tend to indicate the beginnings of tables, the phrases "service cost," "interest," and "differences between expected and actual experience" indicate table content, and the phrase "total pension liability – ending" typically indicates the end of useful content. In the second stage, the identified tables are parsed, and regular expressions are used to scrape service cost and interest data.

At the same time, researchers try to scrape information on what units the figures are in (for example, dollars or thousands of dollars), the names of the pension funds, and the corresponding time period. It is challenging dealing with tables that have complicated structures. It may make sense to group the tables according to structure and build a separate scraping model for each structure type. Another approach that could be explored in the future involves using machine learning to predict structure type. Model features could be based on the x- and y-coordinates of individual characters.

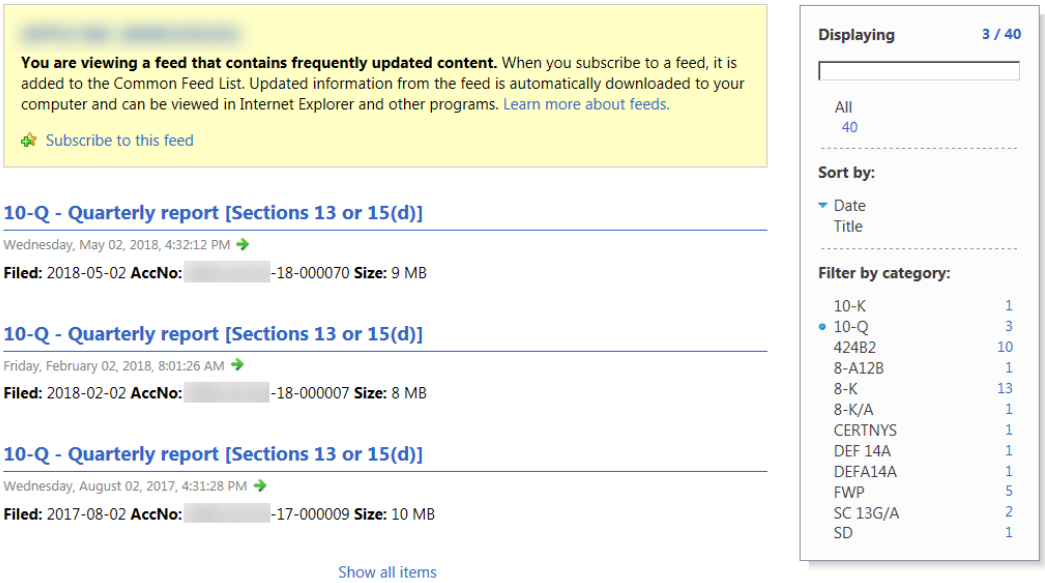
6. Securities and Exchange Commission Filing Metadata

6.1 Background

The EDGAR (Electronic Data Gathering, Analysis, and Retrieval) database on the SEC website contains financial filing information for publicly traded companies. EDGAR is used often by Census Bureau analysts to impute missing values and validate responses for many economic surveys. In particular, the 10-K and 10-Q reports provide valuable annual and quarterly financial information, respectively. For the most part, the process of going into EDGAR or visiting company websites to find out when new 10-K and 10-Q reports are available and then obtaining the data is manual. An ideal process would be automated.

6.2 Methodology

To this end, researchers have started using Python to scrape filing metadata from the EDGAR database. EDGAR has a Really Simple Syndication (RSS) feed, which can be accessed to obtain recent filing information. Figure 4 is a screenshot of the RSS feed.



[redacted]

You are viewing a feed that contains frequently updated content. When you subscribe to a feed, it is added to the Common Feed List. Updated information from the feed is automatically downloaded to your computer and can be viewed in Internet Explorer and other programs. Learn more about feeds.

Subscribe to this feed

10-Q - Quarterly report [Sections 13 or 15(d)]

Wednesday, May 02, 2018, 4:32:12 PM

Filed: 2018-05-02 **AccNo:** [redacted]-18-000070 **Size:** 9 MB

10-Q - Quarterly report [Sections 13 or 15(d)]

Friday, February 02, 2018, 8:01:26 AM

Filed: 2018-02-02 **AccNo:** [redacted]-18-000007 **Size:** 8 MB

10-Q - Quarterly report [Sections 13 or 15(d)]

Wednesday, August 02, 2017, 4:31:28 PM

Filed: 2017-08-02 **AccNo:** [redacted]-17-000009 **Size:** 10 MB

Show all items

Displaying 3 / 40

All
40

Sort by:

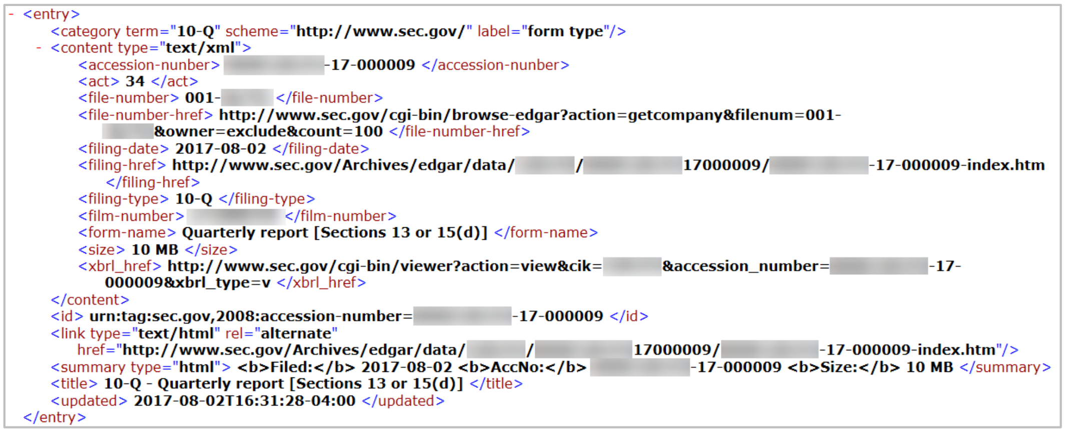
Date
Title

Filter by category:

10-K	1
● 10-Q	3
424B2	10
8-A12B	1
8-K	13
8-K/A	1
CERTNYS	1
DEF 14A	1
DEFA14A	1
FWP	5
SC 13G/A	2
SD	1

Figure 4. Screenshot of the EDGAR RSS feed as displayed in a web browser. The most recent 10-Q filings for a specific company are listed.

The same information can be fetched by querying the RSS feed from Python, for example. In order to do so, one needs to supply the desired filing type (for example, 10-K or 10-Q) and the Central Index Key of the company, which is a unique filer identifier. A Python script based on the Beautiful Soup module (Crummy, 2017) is used to submit a query to the feed and fetch results in Extensible Markup Language (XML) format. The XML format is structured and based on standardized tags. Figure 5 shows part of an XML file created using this method that contains recent 10-Q filings for a specific company.



```
- <entry>
    <category term="10-Q" scheme="http://www.sec.gov/" label="form type"/>
  - <content type="text/xml">
      <accession-nunber> [redacted]-17-000009 </accession-nunber>
      <act> 34 </act>
      <file-number> 001-[redacted] </file-number>
      <file-number-href> http://www.sec.gov/cgi-bin/browse-edgar?action=getcompany&filenum=001-
        [redacted]&owner=exclude&count=100 </file-number-href>
      <filing-date> 2017-08-02 </filing-date>
      <filing-href> http://www.sec.gov/Archives/edgar/data/[redacted]/[redacted]17000009/[redacted]-17-000009-index.htm
        </filing-href>
      <filing-type> 10-Q </filing-type>
      <film-number> [redacted]</film-number>
      <form-name> Quarterly report [Sections 13 or 15(d)] </form-name>
      <size> 10 MB </size>
      <xbrl_href> http://www.sec.gov/cgi-bin/viewer?action=view&cik=[redacted]&accession_number=[redacted]-17-
        000009&xbrl_type=v </xbrl_href>
    </content>
    <id> urn:tag:sec.gov,2008:accession-number=[redacted]-17-000009 </id>
    <link type="text/html" rel="alternate"
      href="http://www.sec.gov/Archives/edgar/data/[redacted]/[redacted]17000009/[redacted]-17-000009-index.htm"/>
    <summary type="html"> <b>Filed:</b> 2017-08-02 <b>AccNo:</b> [redacted]-17-000009 <b>Size:</b> 10 MB </summary>
    <title> 10-Q - Quarterly report [Sections 13 or 15(d)] </title>
    <updated> 2017-08-02T16:31:28-04:00 </updated>
  </entry>
```

Figure 5. Screenshot of an XML file containing information on recent 10-Q filings for a specific company. This XML file was created in Python using information scraped from the RSS feed.

The XML file can be parsed easily using regular expressions (Mitchell, 2015, p. 22) or methods in Beautiful Soup to scrape filing dates and, in turn, determine whether a filing was made recently. Other useful information contained in the XML file include the URL to the corresponding report and an indicator for whether the filing is an amended version. The scraped metadata can be output to an Excel file that is easier for analysts to work with.

The natural next step is to scrape actual financial information from the reports pointed to by the filing metadata. To this end, the SEC recommended that the Economic Directorate consider the Arelle software (Arelle, 2018). Arelle understands the Extensible Business Reporting Language (XBRL) format in which the financial data are contained. A Civic Digital Fellow who worked in the Economic Directorate during the summer of 2018 investigated Arelle and other approaches using Python. A Python module called lxml was found to be more convenient to parse the financial data. The Fellow developed a pipeline to automate the process of querying the RSS feed, parsing the financial data, and applying machine learning to map the XBRL tags to the Economic Directorate's terminology and data items. Discussions will continue with the SEC regarding what methods and technology of theirs could be used by the Economic Directorate.

7. Building Permit Data

7.1 Background

New construction data collected by the Census Bureau are used by government agencies and policy analysts to measure and evaluate size, composition, and change occurring within the construction sector. To measure new construction, the Census Bureau conducts the Building Permits Survey (BPS) and the Survey of Construction (SOC), and to evaluate the private nonresidential frame for Construction Spending (U.S. Census Bureau, 2018), the Census Bureau conducts the Nonresidential Coverage Evaluation (NCE). These three programs use permit authorization information from the building permit jurisdictions. As with many economic programs, costs are increasing, response rates are decreasing, and respondents are feeling burdened. Authorization information on new, privately owned construction is available online for some building permit jurisdictions, and it makes sense to explore the feasibility of scraping these data.

7.2 Methodology

In October 2015, research began on examining issues regarding incorporating publicly available building permit data into construction surveys. The initial stage of research focused on two building permit jurisdictions, Chicago, IL and Seattle, WA, whose data are publicly available through APIs. During this stage, data from these two jurisdictions were analyzed to determine advantages, limitations, and implications surrounding incorporation of these new potential data sources. The result of this initial research was a promising first step as the new sources appeared to provide timely and valid data with respect to corresponding BPS data.

In mid-2016, work continued on the project along two fronts. The first front consisted of researching publicly available data for building permit jurisdictions across the U.S. with a focus on jurisdictions that issued large numbers of residential permits in 2015. Here information was discovered in different formats. Other than APIs, publicly available building permit data were also obtainable via downloadable reports, Excel files, database queries, and other media.

The second front consisted of additional research into the Chicago and Seattle data sources. Through validation, researchers noticed differences in classifications and definitions from one jurisdiction to another. For example, the term “living space” versus “finished floor space” when reporting residential square footage data. Also, publicly available building permit data do not seem to provide complete information on new construction. Information on housing units and specific physical characteristics is generally lacking at the level of detail needed for estimation. In many cases, these new data sources will only provide broad construction information.

By the end of 2017, building permit jurisdictions for Nashville, TN; Las Vegas, NV; Mesa, AZ; San Jose, CA; Austin, TX; Cary, NC; and Boston, MA were included in the research because they appeared to provide information found lacking above. This research showed that classification (residential versus nonresidential) was becoming a standard data item. Greater and more accurate reporting was evidenced in the research. Next, attention turned to the availability of housing unit information as a key point of focus. Housing unit information is the most critical item required of a new data source. Unfortunately, it is one of the least reported. Consistency and definitional differences that affected the usefulness of the data source were noted.

7.3 Challenges and Future Work

Many challenges in incorporating publicly available building permit data from the web into construction surveys are related to the Big Data concerns of representativeness and consistency of the data source. Building permit data will likely be available for areas where new construction activity is large or increasing. Areas where new construction is minimal or limited may not be willing to invest necessary resources to make their information available online. Unfortunately, although classification information is improving, housing unit information is not. Lastly, these data are available in many different formats. A viable solution might have to be able to extract information from APIs, reports, and databases.

Prior to formally incorporating publicly available building permit data into new construction surveys, a number of consistency issues must be addressed. Important issues involve dealing with certain key characteristics such as construction classification that use various terms and definitions across jurisdictions. Finally, successful ongoing validation against corresponding BPS, SOC, and NCE data will also be required to ensure appropriate coverage and completeness of building permit information. Currently, a couple third-party data sources are being examined as a possible solution: Zillow, an online real estate database company, to complement collection of housing characteristic data and Construction Monitor, a compiler of building permit information for use by businesses, suppliers, and contractors, to obtain monthly permit data.

8. Efforts to Improve Sampling Frames

Web scraping can also be used to improve sampling frames and universes and to assess their coverage. For an example in the context of agriculture surveys, see Young (2018). There are a couple efforts to scrape location and contact information of various entities to improve universe coverage for programs in the Economic Directorate. The first effort involves the surveys of prisons, jails, and correctional facilities that the Economic Directorate conducts on behalf of the Bureau of Justice Statistics to collect information on the inmate population and confinement conditions. Another Civic Digital Fellow who worked in the Economic Directorate during the summer of 2018 used web crawling, web scraping, and text classification methods similar to SABLE’s to find previously unknown juvenile facilities.

Another effort involves public sector surveys of state and local governments. There is an idea to use SABLE to crawl government websites and create a more accurate frame of tax collectors, which would assist QTax. Something to keep in mind for this potential project are the various arrangements that states have regarding how taxes are collected and on what government websites this information may be found. For some states, a single county official collects the taxes for every taxing jurisdiction (e.g., municipalities) in the county. In other states, every taxing jurisdiction may do the collection itself.

9. Next Steps with Web Scraping

In the Economic Directorate, overall there is interest in making greater use of online sources and automating web scraping and web crawling processes that currently are manual. In terms of next steps for SABLE, a main goal is to release a data product, either to the Bureau of Economic Analysis or the public in general, that is based on scraped data. The public pensions statistics project has made the most progress in this direction, and the public sector area in general seems like an ideal setting for other data products.

Analysts in the Economic Directorate currently use SEC's EDGAR database and are interested in seeing how the metadata project and related efforts progress. A couple key next steps are to work with various survey teams to see how they can best use this information and incorporate web scraping into their production cycles. Lastly, the Census Bureau has formed a Bureau-wide working group to address policy issues related to web crawling and web scraping. Having policies in place would provide a needed framework in which to scrape, crawl, and use APIs in a way that informs and respects respondents and does not place undue burden on website servers.

Acknowledgments

The authors would like to thank Carol Caldwell, Michael Kornbau, and Rebecca Hutchinson for reviewing this paper and Selvaratnam Sridharma, Angela Delano, Diane Willimack, Demetria Hanna, and Stephen Owens of the U.S. Census Bureau and Civic Digital Fellows Suryabrata Dutta and Kevin Wu for their contributions and help.

References

The Apache Software Foundation. (2018). Apache Nutch. <http://nutch.apache.org>. Accessed June 18, 2018.

Arelle. (2018). Arelle: Open Source XBRL Platform. <http://arelle.org/>. Accessed July 13, 2018.

Bird, S. (2006). NLTK: The Natural Language Toolkit. *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions.* Sydney, Australia: Association for Computational Linguistics, 69–72.

Crummy. (2017). Beautiful Soup. <https://www.crummy.com/software/BeautifulSoup/>. Accessed June 18, 2018.

Dumbacher, B. and Capps, C. (2016). Big Data Methods for Scraping Government Tax Revenue from the Web. *2016 Proceedings of the American Statistical Association, Section on Statistical Learning and Data Science*. Alexandria, VA: American Statistical Association, 2940–2954.

Dumbacher, B. and Diamond, L.K. (2018). SABLE: Tools for Web Crawling, Web Scraping, and Text Classification. *2018 Federal Committee on Statistical Methodology Research Conference*.

Dumbacher, B. and Hanna, D. (2017). Using Passive Data Collection, System-to-System Data Collection, and Machine Learning to Improve Economic Surveys. *2017 Proceedings of the American Statistical Association, Business and Economic Statistics Section*. Alexandria, VA: American Statistical Association, 772–785.

Governmental Accounting Standards Board. (2018). GASB: Governmental Accounting Standards Board. <https://www.gasb.org/home/>. Accessed June 18, 2018.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Second Edition). Berlin, Germany: Springer.

Mitchell, R. (2015). *Web Scraping with Python: Collecting Data from the Modern Web*. Sebastopol, CA: O'Reilly Media, Inc.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Puts, M., Daas, P., and de Waal, T. (2015). Finding Errors in Big Data. *Significance*, The Royal Statistical Society, June 2015.

Shinyama, Y. (2013). PDFMiner. <http://www.unixuser.org/~euske/python/pdfminer/index.html>. Accessed June 18, 2018.

Snijkers, G., Haraldsen, G., Jones, J., and Willimack, D.K. (2013). *Designing and Conducting Business Surveys*. Hoboken, NJ: John Wiley & Sons, Inc.

Statistics Canada. (2018). About us. <https://www.statcan.gc.ca/eng/about/about>. Accessed June 18, 2018.

Tan, P.N., Steinbach, M., and Kumar, V. (2006). *Introduction to Data Mining*. New York, NY: Pearson.

Tennekes, M., de Jonge, E., and Daas, P.J.H. (2013). Visualizing and Inspecting Large Datasets with Tableplots. *Journal of Data Science*, *11*, 43–58.

U.S. Census Bureau. (2018). Construction Spending. <https://www.census.gov/construction/c30/c30index.html>. Accessed June 28, 2018.

Young, L.J. (2018). Evaluating the Use of Web-Scraped List Frames to Assess Undercoverage in Surveys: Lessons from Local Foods Marketing. *2018 Federal Committee on Statistical Methodology Research Conference*.