Harvesting unstructured data in heterogenous business environments; exploring modern web scraping technologies

Robbin Vording
University of Twente
PO Box 217, 7500 AE Enschede
the Netherlands
r.m.vording@student.utwente.nl

ABSTRACT
Web scraping technology can be used to retrieve data from multiple sources efficiently and effectively, but it could be difficult for companies to adopt this technology. This research includes a literature review on web scraping technology in general as well as its techniques and tools. Web scraping has been identified to have a syntactic and a semantic level. There seems to be no valid alternative for web scraping, where APIs have been discussed. A specific adoption model for web scraping technology is nonexistent, making it difficult to manage the adoption of said technology. Current adoption models are discussed and used as foundation for a web scraping adoption model. To determine the applicability of the technology, a case study will be conducted at a Dutch Logistics Service Provider to identify use cases for the use of web scraping technology, resulting in a recommendation for the use of the technology.

KEYWORDS
Web scraping, web harvesting, adoption model

1. INTRODUCTION
Websites contain stored data and are publicly available on the internet. The data is therefore subject to be found by anyone that can access the website. This data can be gathered in many ways as described by Sirisuriya [27], from manually searching and copying the data, to automated programs that analyze the page, identify and retrieve the desired data.

For collecting data efficiently, web scraping technology can be used. Web scraping, also known as web harvesting or web extraction and considered to be a part of data mining, is a technique that consists of retrieving data from the internet and stores it into a file or database [25]. With the use of web scraping technologies, data from webpages, whether that is structured or unstructured, can be harvested and turned into a structured format [25].

Web scraping tools or software are used to automate the data harvesting process [27]. The tools are able to analyze the webpage or website and acquire the desired data. After the data collection from the website(s), the data is parsed into a file or database with the purpose of structuring the data after which the data can be processed.

34th Twente Student Conference on IT, January. 29th, 2021, Enschede, The Netherlands.

Adoption models, if existent, will be researched for applicability and use. It can be difficult for companies to adopt web scraping technology, therefore research will be conducted on the use of adoption models for web scraping, as well as generic information technology adoption models to determine whether they fit web scraping technology.

To determine value and applicability of web scraping technology for businesses, a case study has been conducted at a Dutch business in the logistics sector. This case study was conducted at a company from the Netherlands, that operates as a Logistics Service Provider, and will be referred to as LSP. This logistics company focusses on transport and operates on multiple niche markets within this sector. LSP is focused on innovation and sustainable transport.

LSP is a third party, solely transporting goods, therefore they rely on information that is provided by their customers or the receivers of the transport. This information is acquired by orders or (tracking) websites/platforms. The case study will determine if web scraping technology would fit in their data acquisition.

1.1 PROBLEM STATEMENT
Web scraping technologies are available for businesses. They can adopt a web scraping technology for their business processes, but there are currently no adoption models specifically for web scraping technology.

LSP operates in a heterogenous business environment. Their customers consist of a wide variety of businesses and therefore receives and acquires data in multiple formats.

For LSP, a solution should be presented or recommended that fits the needs of their business.

1.2 RESEARCH GOAL & QUESTIONS
For this research, the research goal is to find or create an adoption model for web scraping technology.

To be able to achieve this goal and structure the research, the following questions have been formulated, divided into knowledge questions and research questions.

Knowledge questions:

i. What techniques are available for web scraping purposes?

ii. What tools are available for web scraping purposes?

iii. What models exist for web scraping technology adoption?

Research questions:

iv. How can web scraping technology be implemented in a logistics service providing company (LSP)?

   a. What are the use cases for web scraping technology?

   b. What is a suitable web scraping solution?

1.3 PAPER STRUCTURE

The paper is structured starting with the literature review on web scraping technology, techniques and tools. The paper will go in-depth on the topics of web scraping, background information about web pages, web scraping techniques, levels of web scraping and lastly web scraping tools.

The paper will then focus on applying the knowledge to current businesses. Therefore, adoption of web scraping technology will be researched and the applicability of the technology is tested with a case study.

2. RELATED WORK

The research has started with a literature review, that has started with the structure of Webster et al. [30] after which literature has been collected by going forwards and backwards in the literature.

Initially, the Scopus database was selected to find relevant literature. For research question i, the search started with the following search query:

- TITLE-ABS-KEY (("web scraping" OR "web scrapping" OR "web harvesting" OR "web extraction") AND techniques) AND (LIMIT-TO (SUBJAREA , "COMP"))

This resulted in 138 when limiting to the subject area “Computer Science”. While reviewing this list of documents, it became apparent that these documents were not focused on web scraping techniques, but they contained a use of a technique, rather than exploring or comparing different techniques. Because of a focus on the various techniques, these papers did not fit the inclusion criteria.

A new approach was taken, by moving to the Google Scholar database. The same search query resulted in approximately 11.000 documents. The first pages with results showed literature including overviews, comparative studies or explorative studies on the web scraping techniques. This resulted in the selection of the following 8 papers: Glez-Peña et al. [9], Gunawan et al. [10], Karthikeyan et al. [12], Malik et al. [14], Munzert et al. [19], Saurkar et al. [24], De S Sirisuriya [27] & Zhao [31]

A second query was used with the Scopus library to search for other comparative studies on web scraping. The following query was used:

- TITLE (("web scraping" OR "web scrapping" OR "web harvesting" OR "web extraction") AND compar*)

This resulted in 3 papers, where 1 was selected: Mehta et al. [17]. This query only searched in paper titles, to ensure finding papers within the inclusion criteria.

The search for research question ii was conducted in the Scopus library with the following search query:

- TITLE (("web scraping" OR "web scrapping" OR "web harvesting" OR "web extraction") AND tools)

This resulted in 4 documents, of which 1 was selected: Matta et al. [15].

After this set of papers has been established for review, other papers were found through backward searching.

A second search for literature has been conducted for web scraping adoption models. Scopus, Google Scholar and Web of Science have been used with the following search query:

- TITLE-ABS-KEY (("web scraping" OR "web scrapping" OR "web harvesting" OR "web extraction") AND (adoption OR "success model" OR acceptance))

This search resulted in 1 paper by Demoulin et al. [5] that applied the Technology Acceptance Model to web scraping technology and found relevant variables for the model. To gain more insights, a search was conducted for the use of Information System (or information technology) adoption models, such as the Technology Acceptance Model (TAM) by Davis et al. [3], the Information System Success Model (ISSM) by DeLone and McLean [4] and the Unified Theory of Acceptance and Use of Technology (UTAUT) by Venkatesh et al [29].

De S Sirisuriya [27], Matta et al. [15] and Mehta et al. [17] have conducted comparative studies on web scraping. These studies have compared web scraping techniques, ranging from manual copy and paste to computer vision analyzers. Research about several web scraping tools is also included.

Octoparse has compared their tool with 4 others [28]. Their extensive comparison includes the tools Octoparse, Parsehub, Mozenda, Dexi.io and Import.io.

Bol Raap et al. have done research on the architecture and common data model for open data-based cargo-tracking in synchromodal logistics [1]. Their research shows a typical architecture for businesses in the logistics sector and proposes a model for this sector. The paper also describes how cargo is currently being tracked.

2.1 WEB SCRAPING

Web scraping, which is also knows as web extraction or web harvesting, is a technique to extract data from the World Wide Web and save it to a file system or database for later retrieval or analysis [31]. The web can be scraped utilizing multiple techniques, which are described in detail later in this paper. It can be done manually or automatically by a bot or a web crawler. Web scraping is acknowledged as an efficient and powerful technique to collect (big) data.

Web scraping is also a technique that is capable of extracting unstructured data and transforming that into structured data that can be stored and analyzed in, for example a database [27]. This can be seen in Figure 1 below.

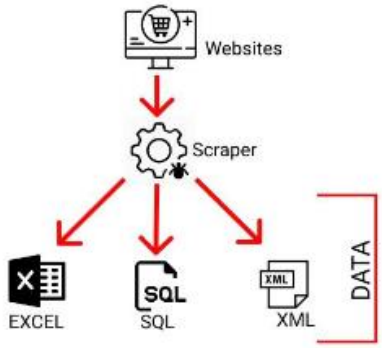

Figure 1. An overview of Web Scraping [15].

Web scraping can be put into a couple of levels: the semantic scraping level and the syntactic scraping level. Fernández-Villamor et al. [7] have created a framework for web scraping, which will be discussed later in this paper. The semantic level defines the mapping between web data and semantic web resources. The syntactic level defines the required technologies to extract actual data from web pages.

2.1.1 Applications

Applications of web scraping are expanding. Matta et al. [15] describes three major applications of web scraping: price monitoring, market research and sentiment analysis. Cost plays a fundamental role in e-commerce. From a business perspective, the competitors need to be monitored Manually tracking the prices of all competitors is not a viable option if prices fluctuate regularly. That is where web scraping has purpose: it automates extracting pricing information from competitors and can provide up to date information from all of the competitors in one document or database that is easy accessible.

Market research requires highly accurate data for a better decision making process. High quality and insightful data fulfils the requirements of market analysis as well as business intelligence worldwide. This makes web scraping a viable technique for business processes, such as market trend analysis, market pricing, optimizing point of entry, research & development and competitor monitoring.

Sentiment Analysis is a popular application of web scraping data from social media. An application here is to predict elections. A computer could predict the name of the winning candidate by analysing tweets and posts where the name of the candidate is not even required. Sentiment recognition algorithms sense hints and detect patterns that go beyond the post. By extracting data using web scraping, more accurate analysis can be performed than before when grouped posts (based on hashtags on Twitter, for example) were used.

2.1.2 Web Crawling

Web crawling is a term that is often used alongside web scraping. Web crawling is a technique for crawling on the internet, which means to traverse web pages by following hyperlinks [16]. The term indicates a program its ability to navigate web pages on its own [2]. Web crawlers can be used for exploring what a website has to offer, but it can also be used without a strict defined goal or purpose. An application of a web crawler would be to crawl through a web shop for products. It has clear boundaries within the domain of the web shop and will collect the data of the product pages. Often, a web crawler does not extract actual data from a page, that is done later with a web scraper. The web crawler its purpose is to explore the web as a whole or within a specific domain. Another application of web crawlers are search engines. Google, for example, makes use of web crawlers to identify websites on the web. Without a web crawler, a search engine could not exist the way it operates nowadays. Mattosinho [16] describes two important issues to address with the use of web crawlers: first to have a good crawling strategy (including the algorithm strategy for crawling through new pages) and intelligent mechanisms to optimize the re-crawling process. Secondly, it must be optimized to the available hardware. Crawling is a computational intensive task, so the system must be able to cope with different situations, such as hardware failure, server problems and parsing errors, while still maximizing the work to ensure that the maximum advantage is taken out of the available resources (such as memory, cpu and (limited) network brandwidth). Mattosinho describes a web crawler system as seen in Figure 2.

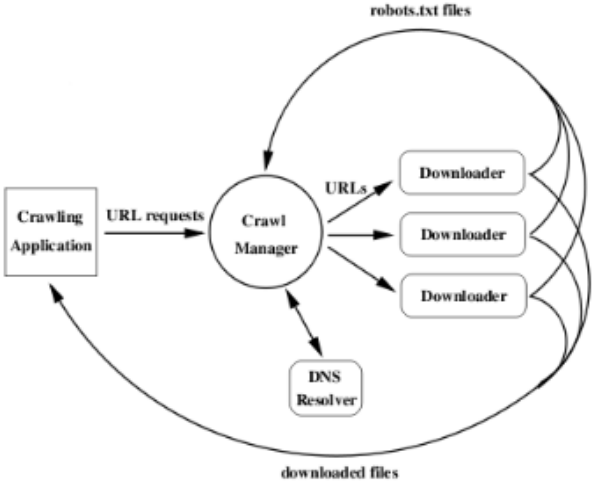

Figure 2. Model of a web crawler system [16].

A web crawler system consists of a crawl manager, which is responsible for forwarding URL requests to the downloaders. The downloaders are responsible for establishing connections with different web servers and polling these connections for arriving data. They are also tasked with downloading the robots.txt document from the website, which is a file that specifies rules for robots to follow when exploring the website. The crawl manager is therefore also responsible for following the set rules in the robots.txt file.

2.2 WEB PAGES

A web page displayed on your screen in a web browser, makes use of a few technologies before it can be displayed. These technologies will be discussed here.

When a website URL is entered in a web browser and enter is hit, the browser starts with finding the corresponding IP address for the website, which is done through the Domain Name System (DNS) protocol. Now, a connection has been established with the web server of the website you are looking for. The browser will send a HyperText Transfer Protocol (HTTP) request to the web server to obtain the data for the web page. The web server responds to this request and normally sends the source code of the page, after which the browser will parse this code and transform it into a web page with its intended layout to be displayed on the screen.

The source code of a web page consists of HyperText Markup Language (HTML) and Cascading Style Sheets (CSS). HTML specifies how a document is structured and formatted as shown in Figure 3, where CSS specifies how to style the document.

A structured HTML page can be parsed into a Document Object Model (DOM), which is a logical hierarchy tree structure of the page as displayed in Figure 3. The DOM makes a tree out of the page its HTML structure, making it easier to find the specific location of data.

Extensible Markup Language (XML) is a markup language that is used to encode a document in an easy understandable format for both humans and bots. It was initially focused on documents, but is currently also used for data structures. XML has been extended to XHTML, which is a stricter form of the HTML language, making documents more formatted.

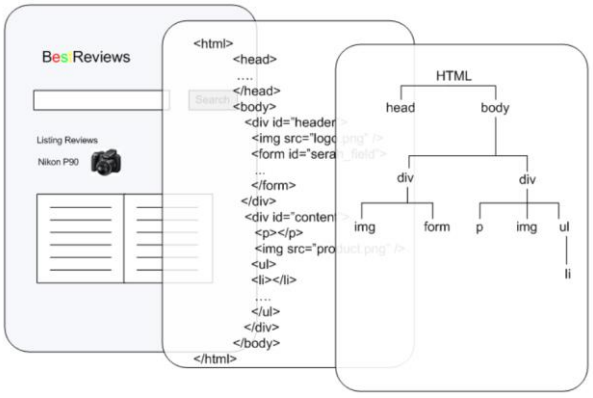

Figure 3. A web page from three different perspectives (Page presentation, HTML code and the DOM) [16].

2.3 TECHNIQUES

2.3.1 Manual Copy and Paste

This first technique is a technique completely manually performed by a human. You can control what you information you extract, but this technique only works for smaller information sources. It becomes difficult to only collect the relevant data you are after when you are increasing the amount of information you collect. This technique is time consuming, especially when working with a large dataset or multiple data sources [27], and repetitive [23]. However, this method is effective when a web site has taken anti web scraping measures [12].

2.3.2 Grepping Text and Regular Expression

With this technique, information can be extracted from a webpage or any other text source. It is based on the UNIX 'grep' command [13] and stands for "globally search for a regular expression and print matching lines". This technique searches for a regular expression within your source and returns matching lines. This technique is to be run from a command line and can be used by a bot. This function is often integrated with standard programming languages, such as Python, R and Perl.

An alternative to this technique would be the use of the search function that browsers and operating systems offer. This would be similar and can be done manually or automatically.

2.3.3 Hypertext Transfer Protocol (HTTP) Programming

This is the first technique that is only applicable on web servers. By sending a GET request to a web server, you are able to extract the information on a web page. For this technique, you request a web page, after which you can extract its information. It can be used for both static and dynamic web pages [24]. HTTP Requests can be sent through standard programming languages, such as Java and Python.

The alternative to this method would be to visit the web page you sent a HTTP request to. This way you can look at the page you requested and manually search the information needed.

2.3.4 Hyper Text Markup Language (HTML) Parsing

This technique is only applicable on web pages, which are built up using HTML. With this technique, information can be extracted by making use of the web page its layout and structure. It mainly targets nested HTML pages [17] and can be used for various applications, such as extracting text, links and contact information. HTML Parsing can be done with programming languages, such as JavaScript, as well as Hyper Text Query Language (HTQL) and XQuery.

Alternatively, a web browser can show the web page it's source code, which will show the HTML structure behind the page.

2.3.5 Document Object Model (DOM) Parsing

Through this representation of the web page, its structure is easily identifiable, after which you can pinpoint which node contains information.

XML Path Language (XPath) is a query language that can be used to extract information from a DOM. XPath can be used to process the tree and pick the nodes containing information that is requested within the query [20] as shown in Figure 4. The red line shows how a web scraping agent goes through a DOM tree using XPath. In the left document the line ends at a hyperlink, where it continues on the hyperlink target page on the right document making its way to the information of the desired paragraph.

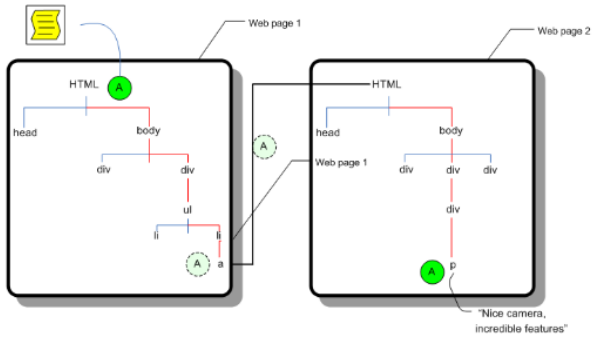

Figure 4. Example of a XPath query search [16].

Gunawan et al. [10] has conducted a study which showed DOM Parsing needs a large amount of memory in comparison to Regular Expression and using Xpath, but does work faster and use less data.

Internet browsers, such as Microsoft Edge, Google Chrome and Mozilla Firefox, already parse a page into a DOM, which can be used for manual search as an alternative.

2.3.6 Vertical Aggregation Platforms

Vertical Aggregation Platforms are platforms created by businesses with access to huge computing power [17]. They target a definite vertical. Without almost any human involvement, the platform creates and monitors bots for specific verticals. The bots are created automatically based on knowledge of their targeted vertical and their efficiency can be measured by the quality of the data extracted.

2.3.7 Semantic Annotation Recognition

As described by Malik et al. [14], semantic annotations can be used to locate specific data snippets. The annotations may be organized into a semantic layer where they are stored and manager separately from web pages. This means that with this technique, a data schema and instructions can be retrieved from the semantic layer before scraping the web page. Semantic Annotation can be considered enriching a document by creating a connection between the text and their semantic descriptions.

2.3.8 Visual Web Page Analysers

This is a technique that is focussed on extracting information from web pages by using machine learning and computer vision to attempt to interpret pages visually as a human being

could. This technique could enable easier extraction from unstructured pages.

2.3.9 Alternative Techniques

With the previous described web scraping techniques and whether these specific techniques have an alternative, a general alternative to web scraping also exists, however it is not always available. An alternative to web scraping is the use of an Application Programming Interface (API) offered by the website/data owner. An API provides a means for the outside world to access their data repository in a structured way – meant to be consumed and accessed by computer programs, not humans [2]. An API allows the owner of the data control over who can access their data, how often it can be accessed and what data is accessible. However, there are limitations and situations where web scraping may be preferred as described by Vanden Broucke et al. [2]:

- The website where you want to retrieve data from does not provide an API.
- The API provided is not free, whereas the website is.
- The API provided is rate limited (only a number of times per minute, per day, etc.).
- The API provided does not expose all the data you wish to obtain, whereas the website does.

Glez-Peña et al. [9] also concluded in their research that an API is no priority for web site creators, as they first focus on providing high quality content.

2.4 SEMANTIC SCRAPING FRAMEWORK

Fernández-Villamor et al. [7] have created a semantic scraping model, which can be seen in Figure 5.

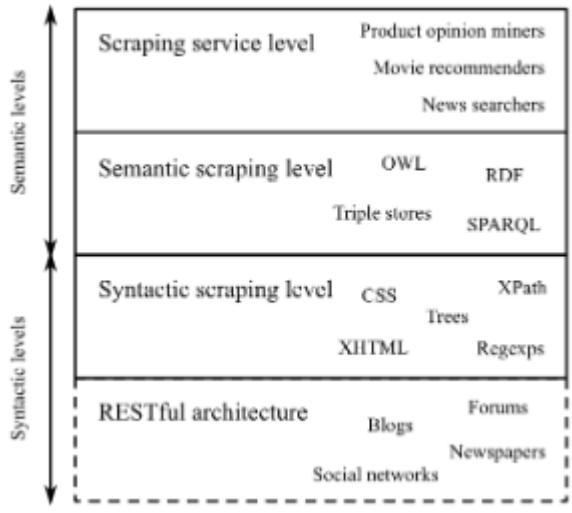

Figure 5. Semantic Scraping Framework [7].

The framework consists of three levels of abstraction. These levels are discussed below. The three levels have been stacked on top of the REST architectural style, which is what the World Wide Web is based on [7].

2.4.1 Scraping Service Level

Services and applications, providing value to their users, that make use of the semantic scraping level are to be found in this level. Examples given in the framework are Product opinion miners, Movie recommenders and News searchers. These services make use of the Segment Analysis application of web scraping as mentioned earlier in section 2.1.1. They tend to benefit from an increased level of knowledge.

2.4.2 Semantic Scraping Level

Fernández-Villamor et al. [7] have created a semantic scraping Resource Description Framework (RDF) model, which can be seen in Figure 6 below. This level defines the mapping between web data and semantic web resources.

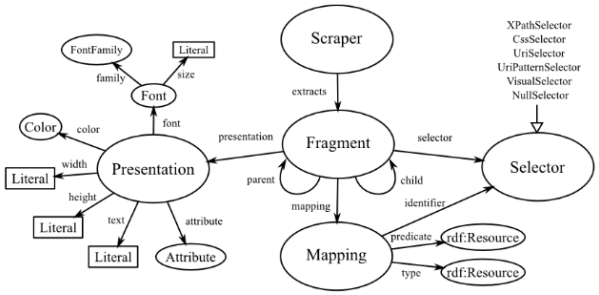

Figure 6. Semantic scraping RDF Model [7].

The model consists of a few classes. The first being the Scraper, which is an agent that is able to extract particular fragments from the web (page). A Fragment can be any element of an HTML document and serves to represent and traverse a whole subtree of a document. Selectors are means to identify a web document fragment. Multiple selectors are included in the model, such as the XPathSelector or CssSelector. Mapping is done between a fragment and an RDF resource or blank node. RDF triples are produced here. The Presentation of a fragment is also included in the framework, which includes HTML attributes as well as visual parameters, like colors or fonts. The semantic scraping level is mostly focussed on the document its structure, where the syntactic level focusses on extraction of actual data. Earlier discussed techniques in 2.3.5, 2.3.6 and 2.3.7 are classified to be a part of this level.

2.4.3 Syntactic Scraping Level

This level defines the required technologies to extract data from web resources. Fernández-Villamor et al. [7] describes a few techniques, such as Cascading Style Sheet (CSS) selectors, XPath selectors, URI patterns and Visual selectors. Earlier discussed techniques in 2.3.1, 2.3.2, 2.3.3 and 2.3.4 would be classified in this level.

2.5 TOOLS

There exist a large amount of software and tools that provide web scraping technology to be used. Their main purpose is to extract data from websites and make use of one or multiple techniques that are described in this paper. Using a tool makes the process easy. You can insert where or select the data from web page that you intent to extract. Then, you organize how the data will be extracted to the output document, such as a database. The tool will then be able to perform your scrape assignment, that you are also able to schedule at specific days and times. Most of the tools are only accessible with paid plans, however there are also plenty of tools that provide free plans. A selection of prominent tools has been examined and discussed below.

2.5.1 Scrapy

Scrapy [26] is an open source web crawling framework written in Python that runs on Windows, Linux and Mac. Originally it was designed for web scraping, and can therefore still be used for data extraction. Scrapy is run from the

command line and can easily be used on Windows, Linux and Mac.

2.5.2 Import.io

The Import.io [11] platform allows data extraction behind a login. It also provides several options for data visualization and dashboarding: it offers an export to Tableau and has an API that offers full access to their platform. Behind the functionalities of Import.io is also a Machine Learning aspect, making their tool better in finding new data for you specifically. They also allow for custom code to be run through their tool, as well as API usage.

2.5.3 Dexi.io

Dexi.io [6] offers a large amount of possibilities, where they allow API connections and their platform offers integrations with third party applications, such as social media, Google Drive and Sheets. They allow data extraction behind a login and have a third party integration with a CAPTCHA solving platform. They also offer so called Triggers, which allow information that is scraped to be used in a new web scraping event.

2.5.4 ParseHub

ParseHub [22] is considered a complete tool, with extra functions. It includes data extraction behind a login, solving for text input CAPTCHAs, allowing scraped data to be used in a next page or website to scrape, has a Tableau integration for data export and is available for Windows, Linux and Mac. Parsehub also offers a free version of their tool.

2.5.5 Octoparse

Octoparse [21] is point and click and is easy to use. The tool allows for signing into websites for data extraction as well as solving CAPTCHAs when run locally. Their tool can deal with many features a website can offer, such as use of AJAX, JavaScript and other visual alterations. It also provides API access. Octoparse is a tool that offers a free plan.

2.5.6 Mozenda

Mozenda [18] can extract data from websites as well as (PDF) files and images. It does not need technical knowledge to set up, since there is an interface that is rather simple to use. It allows data extraction of data behind a login. They also offer data integration with a lot of common platforms, such as Facebook, Google, Github and Trello.

2.5.7 FMiner

FMiner [8] is a visual web scraping tool that is built in Python which also allows custom Python code to be run. It offers a variety of techniques for optimal scraping and is capable of dealing with dynamic websites (Ajax, JavaScript). CAPTCHA solving is to be done manually or can be set up with a third party integration. FMiner also allows a keyword input list that can be used in the target website for searching purposes. FMiner offers a version for Windows as well as Mac.

3. ADOPTION

For businesses willing to adopt web scraping technology, there currently is no specific adoption or acceptance model for web scraping technology available. The research from Demoulin et al. [5] introduces the key variables and determinants and applies them to the Technology Acceptance Model.

For Information Systems, or information technology, there exist a number of adoption models, acceptance models or success models. Based on the fact that web scraping technology classifies as a information technology, these general acceptance models can possibly be used for web scraping technology. Amongst the most common models are the Technology Acceptance Model (TAM) by Davis et al. [3], the Information System Success Model (ISSM) by DeLone and McLean [4] and the Unified Theory of Acceptance and Use of Technology (UTAUT) by Venkatesh et al [29]. These models are designed to determine the acceptance of information technology. This means they are used to predict and evaluate success of new technology acceptance. The UTAUT by Venkatesh et al. [29] is a unified model that evolved out of eight major models on technology acceptance. These eight models are:

- Theory of reasoned action (TRA)
- Technology acceptance model (TAM)
- Theory of planned behaviour (TOPB)
- The motivational model (MM)
- The innovation diffusion theory (IDT)
- The model of PC utilization (MPU)
- The social cognitive theory (SCT)
- Combined model of TAM and TOPB (C-TAM-TOPB)

For that reason this model has been selected to use for the adoption of web scraping technology and will be discussed further.

The model, as seen in Figure 7, consists of four independent variables, formed by the main constructs from all models, which make the two dependent variables predictable. The independent variables of the model are:

- Performance Expectancy (PE)
- Effort Expectancy (EE)
- Social Influence (SI)
- Facilitating Conditions (FC)

And the two dependent variables are defined as:

- Technology behavioral intention
- Technology usage behaviour

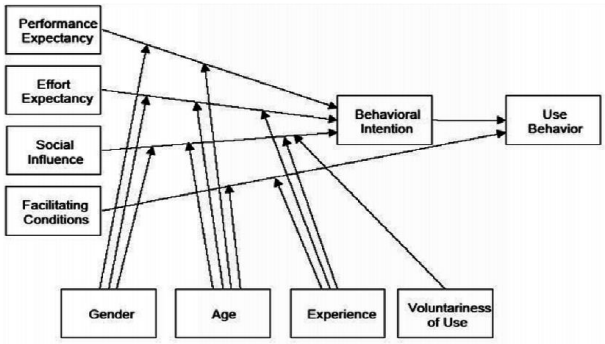

Figure 7. UTAUT Model.

The variables Performance Expectancy, Effort Expectancy and Social Influence have a direct influence on the Behavioral intention variable, meaning the intention to adopt the technology. The variables Facilitating Conditions and Behavioral Intention have a direct influence on the Usage Behavior variable, meaning the initial use of the technology. For the independent variables, a few moderators have been identified that are likely to influence the variables. These moderators are:

- Gender

- Age
- Experience
- Voluntariness of Use

3.1.1 Behavioral Intention

Looking at Behavioral Intention and its independent variables, some things can be predicted about web scraping for it to be successfully accepted by a user. First of all, when considering Performance Expectancy, the technology should perform as expected, so the outcome of using the technology should be the same, or better, than the current situation without use of the technology. With the techniques and tools discussed in this paper, it is likely that web scraping technology would work more efficient than when the tasks are done manually, resulting in a good score for this variable. For Effort Expectancy, it is important for the technology to be easy to work with, so it should not take more effort than in the current situation. As discussed with the techniques and tools, it is highly likely that web scraping technology would require less effort than the manual alternative, so a good score for this variable is expected as well. Lastly, Social Influence is focussed on social influence from the user their primary social circle, but also influence from other employees or users of the technology as well as their business management. This variable seems to be less predictable in regards to the technology and is more focussed on the individual using it. The moderators have different impact on the variables, which can be found in Venkatesh et al. [29].

3.1.2 Usage Behavior

There are two variables that have influence on Use Behavior: Behavioral Intention and Facilitating Conditions. Behavioral Intention, which comes down to the intention to use the technology as broken down in the previous paragraph. For the Facilitating Conditions variable, the organizational and technical infrastructure must be considered adequate to support use of the technology. This variable is considered to be important for web scraping, as it does require computational power to be performed. Everything considered will result into the actual usage behaviour of the technology, which is what the model is used for to determine whether the technology is a success.

The research by Demoulin et al. [5] has shown that the model should be extended with Information Quality determinants, that have influence on the Performance Expectancy and Effort Expectancy. Furthermore, they concluded that top management support is key in customer-oriented companies. For the UTAUT model, this fits as a determinant into the Social Influence variable.

4. CASE STUDY

To determine whether web scraping technology fits into a current business, a case study was conducted to determine what the technology can be used for and how that can be applied. A business in the logistics sector, earlier introduced as LSP, was selected for the case study. This business aims for sustainability and innovation, and is therefore open to explore what web scraping technology can offer.

Interviews have been conducted with stakeholders from the LSP to gain insights in its business processes and markets. Several use cases have been identified that are interesting within the field of web scraping. The use cases were identified as follows:

- Retrieval of information, at specific times, from a predefined page with predefined fields and attributes.
  - ❖ Retrieving the price of vehicle gasoline (Diesel)
- Retrieval of information, at specific times, from a predefined page with predefined fields, but with a flexible amount of pages.
  - ❖ Retrieving orders. The layout of an order is predefined, however the amount of orders is not.
- Retrieval of information, at specific times, from a predefined page with multiple elements.
  - ❖ Retrieving currency rates, with currency as an attribute of the data.
- Retrieval of information, on request, from a flexible amount of pages and multiple elements.
  - ❖ Retrieving requests and orders from different platforms, based on specific elements, such as location, time, freight type, etc.).

These four use cases have been identified to fit web scraping technology. They represent business processes of the LSP and are used on a regular basis. Currently, they search, find and organize this data manually. Web scraping technology could make these business processes more efficient.

5. DISCUSSION

Not much research has been done in the field of web scraping technology. Literature found are limited to either the same source or fail to bring anything new to the table. The comparative studies used in this paper were of help, but needed adjustments and additions to their information. Comparing tools is found to be difficult, since tools are being innovated regularly and information from papers is quickly out of date.

The UTAUT model appears to be a good foundation for a web scraping technology adoption model, however the model needs to be tested with the technology to determine whether more or less variables are needed in the model. The research by Demoulin et al. [5] introduces Information Quality as influencing factor for the acceptance of web scraping technology and should therefore be considered with the UTAUT model. Considering Demoulin et al. [5] their research, it became apparent that the UTAUT model is a good foundation and includes most variables to adoption. Demoulin et al. [5] only introduced Information Quality to the model, making the model overall more complete and valid for this technology. Future work should include a research on testing the model and modifying it to web scraping if necessary.

6. CONCLUSION

The UTAUT model is determined to be a viable adoption model for web scraping technology. The model is a unified model of information technology adoption models which fits well for web scraping. The prediction for some of the variables of the model is that web scraping technology would be successfully adopted. However, there are still other factors, such as facilitating conditions and social influence that are beyond the control of the technology itself and makes it difficult to predict web scraping technology adoption in general. Although the model is not tested, it can be considered as a model at the foundation of web scraping adoption.

Several techniques have been presented and discussed for alternatives. Most techniques have a manual alternative by use of the common web browsers. From manual copy and paste, regular expression search, HTTP programming, HTML and DOM parsing to vertical aggregation platforms, semantic annotation recognition and visual page analyzers. Use of bots starts to come in play when data needs to be extracted at specific times or multiple times per specific timeframe and if there is a large network of sources that contain data to be scraped and when the web needs to be crawled. This paper can be used by businesses that are exploring web scraping technology and are interested in adoption the technology.

There are not many alternatives to the technology as a whole. The use of APIs has been discussed as an alternative to web scraping, however web scraping seems to be more of an alternative to use of APIs. The use of APIs is limited for various reasons, while web scraping remains a viable option under most conditions.

The case study of the LSP has resulted in a few use cases, which can be transformed into requirements for a web scraping solution. Important requirements are retrieving data behind logins and the ability to give parameters or keywords when using the scraper. Using prior scraped data would definitely benefit the business and would therefore be a pro for the solution. After research on the selected tools in section 2.5, only two tools remain as a best solution for LSP. These two tools are Dexi.io and ParseHub. Both tools fit the constructed requirements and can be considered by LSP. A first exploration has been done in this paper, however the LSP should identify their needs closely to determine what tool fits the best to their needs.

7. REFERENCES

[1] Bol Raap, W., Iacob, M.-E., van Sinderen, M., & Piest, S. 2016. *An Architecture and Common Data Model for Open Data-Based Cargo-Tracking in Synchromodal Logistics., 327–343.* doi:10.1007/978-3-319-48472-3_19

[2] Broucke, S. Vanden, Baesens, B. 2018. *Practical Web Scraping for Data Science*. doi:10.1007/978-1-4842-3582-9

[3] Davis, F.D., Bagozzi, R.P., Warshaw, P.R. 1989. *User acceptance of computer technology: a comparison of two theoretical models*. Management Science, Vol. 35, No. 8, August 1989, pg. 982.

[4] DeLone, W. H., & McLean, E. R. 1992. *Information Systems Success: The Quest for the Dependent Variable*. Information Systems Research, 3(1), 60–95. doi:10.1287/isre.3.1.60

[5] Demoulin, N. T. M., & Coussement, K. 2018. *Acceptance of Text-Mining Systems: The Signaling Role of Information Quality*. Information & Management, Volume 57, Issue 1, January 2020. doi:10.1016/j.im.2018.10.006

[6] Dexi.io. Available: https://www.dexi.io/, accessed on: Jan. 15, 2021.

[7] Fernández-Villamor, J.I., Blasco-García, J., Iglesias, C.A., Garijo, M. 2011. *A Semantic Scraping Model for Web Resources – Applying Linked Data to Web Page Screen Scraping.* Proceedings of the 3rd International Conference on Agents and Artificial Intelligence, Volume 2 - Agents, January 28-30.

[8] FMiner. Available: http://www.fminer.com/, accessed on: Jan. 15, 2021.

[9] Glez-Peña, D., Lourenço, A., López-Fernández, H., Reboiro-Jato, M., & Fdez-Riverola, F. 2013. *Web scraping technologies in an API world*. Briefings in Bioinformatics, 15(5), 788–797. doi:10.1093/bib/bbt026

[10] Gunawan, R., Rahmatulloh, A., Darmawan, I., Firdaus, F. 2018. *Comparison of Web Scraping Techniques: Regular Expression, HTML DOM and Xpath*. IcoIESE 2018, Atlantis Highlights in Engineering, vol. 2.

[11] Import.io. Available: https://www.import.io/, accessed on: Jan. 15, 2021.

[12] Karthikeyan T., Sekaran, K., Ranjith D., Vinoth kumar V, & Balajee J M. 2019. *Personalized Content Extraction and Text Classification Using Effective Web Scraping Techniques*. International Journal of Web Portals, 11(2), 41–52. doi:10.4018/ijwp.2019070103

[13] Kernighan, Brian 1984. *The Unix Programming Environment*. Prentice Hall. pp. 102. ISBN 0-13-937681-X.

[14] Malik, S. K., & Rizvi, S. 2011. *Information Extraction Using Web Usage Mining, Web Scrapping and Semantic Annotation.* 2011 International Conference on Computational Intelligence and Communication Networks. doi:10.1109/cicn.2011.97

[15] Matta, P., Sharma, N., Sharma D., Pant, B., Sharma, S. 2020. *Web Scraping: Applications and Scraping Tools.* International Journal of Advanced Trends in Computer Science and Engineering, Volume 9, No.5, September-October 2020. ISSN 2278-3091. doi:https://doi.org/10.30534/ijatcse/2020/185952020

[16] Mattosinho, F.J.A.P., 2010. *Mining Product Opinions and Reviews on the Web.* Master's Thesis. Technische Universität Dresden. Dresden, Germany.

[17] Mehta, K., Salvi, M., Dand, R., Makharia, V., Natu, P. *A Comparative Study of Various Approaches to Adaptive Web Scraping*, ICDSMLA 2019, Lecture Notes in Electrical Engineering 601, doi:https://doi.org/10.1007/978-981-15-1420-3_136

[18] Mozenda. Available: https://www.mozenda.com/, accessed on: Jan. 15, 2021.

[19] Munzert, S. 2014. *Automated Data Collection with R: A Practical Guide to Web Scraping and Text Mining.* ISBN: 9781118834787

[20] Myllymaki, J. 2002. *Effective Web data extraction with standard XML technologies*. Computer Networks, 39(5), 635–644. doi:10.1016/s1389-1286(02)00214-1

[21] Octoparse. Available: https://www.octoparse.com/, accessed on: Jan.15, 2021.

[22] ParseHub. Available: https://www.parsehub.com/, accessed on: Jan. 15, 2021.

[23] Petta, D.L., Mohs, B.K. 2013. U.S. Patent No. 8,595,847. Washington, DC: U.S. Patent and Trademark Office.

[24] Saurkar, A.V., Pathare, K.G., Gode, S.A. 2018. *International Journal on Future Revolution in Computer Science & Communication Engineering*, Volume: 4 Issue: 4, ISSN: 2454-4248, pp. 363-367.

[25] Schintler, L.A., McNeely, C.L. 2017. Encyclopedia of Big Data, *Web scraping*, ch483-1. ISBN: 978-3-319-32001-4

[26] Scrapy. Available: https://scrapy.org/, accessed on: Jan. 15, 2021.

[27] Sirisuriya, S.C.M. De S. 2015. *A Comparative Study on Web Scraping*. URI: http://ir.kdu.ac.lk/handle/345/1051

[28] *Top 5 Web Scraping Tools Comparison*, Octoparse. Available: https://www.octoparse.com/blog/top-5-web-scraping-tools-comparison-2, accessed on: Nov. 22, 2020.

[29] Venkatesh, V., Morris, M.G., Davis, G.B., Davis, F.D. 2003. *User acceptance of information technology: toward a unified view*. MIS Quarterly, Vol. 27, No. 3, September 2003. Pp. 425-478.

[30] Webster, J., & Watson, R. (2002). *Analyzing the Past to Prepare for the Future: Writing a Literature Review*. MIS Quarterly, Vol. 26, No. 2 (Jun., 2002), pp. xiii-xxiii.

[31] Zhao, B. 2017. *Web Scraping*. Encyclopedia of Big Data. DOI: 10.1007/978-3-319-32001-4_483-1