A deep search method to survey data portals in the whole web: toward a machine learning classification model

Andreiwid Sheffer Correa[a,*], Alencar Melo Jr.[a], Flavio Soares Correa da Silva[b]
[a] *Federal Institute of Education, Science and Technology of Sao Paulo - IFSP, Campinas, SP, Brazil*
[b] *Institute of Mathematics and Statistics – University of Sao Paulo, Sao Paulo, SP, Brazil*

This is the author's version of Accepted Manuscript* of an article published by Government Information Quarterly – ISSN: 0740-624X (copyright Elsevier). Please refer to the formal publication at https://doi.org/10.1016/j.giq.2020.101510.

Please cite this article as:
Correa, A. S., Melo Jr., A., Silva, F. S. C. (2020). A deep search method to survey data portals in the whole web: Toward a machine learning classification model. Government Information Quarterly, 37(4), 101510. https://doi.org/10.1016/j.giq.2020.101510

Abstract

The emergence of standardized open data software platforms has provided a similar set of features to sustain the lifecycle of open data practices, which includes storing, managing, publishing, and visualizing data, in addition to providing an out-of-the-box solution for data portals. Accordingly, the dissemination of data portals that implement such platforms has paved the way for automation, wherein (meta)data extraction supplies the demand for quantity-oriented metrics, mainly for benchmark purposes. This has given rise to an issue regarding how to survey data portals globally, especially reducing the manual efforts, while covering a wide variety of sources that may not implement standardized solutions. Thus, this study raises two main problems: searching for standardized open data software platforms and identifying specific developed web-based software operated as data portals. This study aims to develop a method that deeply searches each web page on the internet and formalizes a machine learning classification model to improve the identification of data portals, irrespective of how these data portals implement a standardized open data software platform and comply with the open data technical guidelines. The contributions of this work have been demonstrated through a list of 1,650 open data portals generalized in a training model that makes it feasible to distinguish between a data portal (that may or may not implement a standardized platform) and an ordinary web page. The results provide new insights on how machine-readable, publicly available data are affected by artificial intelligence, with special focus on how it can be used to understand data openness worldwide.

Keywords: Artificial intelligence (AI), machine learning, data portals, open data, survey.

1. Introduction

The rising number of data portals worldwide has provided a rich source of metrics with high potential for automation, which may benefit the society by ensuring quality assessment of the data and metadata. Benchmark exercises emerged from this context as a quantity-oriented and efficient approach to match the rising demands for evidence (Ulrich et al., 2015). A current problem involving the widespread usage of data portals includes how to survey them and gather basic information regarding, for example, the underlying software platforms in use, published datasets, and geographic locations, primarily avoiding any manual processes. An automated method to survey the data portals operating on the internet will reduce the efforts to determine and complement the data that

are somewhat necessary to consolidate a viable whitelist of web addresses or Uniform Resource Locators (URLs) that point to working data portals in the fast-changing context of open data initiatives.

Previous studies (A. S. Correa et al., 2018; A. S. Correa & da Silva, 2019; van der Waal et al., 2014) introduced the concept of managing the repositories of data portals as an open issue; this approach invited a few current initiatives that are sparse and manual, thereby raising the concerns regarding the redundancy of entries, discoverability of new data portals, and traceability of the software platform in use. Initially, the study by (A. S. Correa et al., 2018) identified at least seven different sources of data portals and introduced a method to identify open data software platforms, envisaging a way to automatically survey and check the potential open data from the web. Later, the same authors (A. S. Correa & da Silva, 2019) took advantage of the URL index of the Common Crawl project (an open repository of web crawl data) to survey potential data portals by searching the URL text strings for keywords related to the term "data" and its variants in other languages.

This study advances this topic by surveying data portals automatically using a deep search method directly into the Common Crawl web archive (WARC), which comprises the most detailed content related to the crawled web pages. The deep search approach is based on previous works with novel improvements to analyze each line of the Hypertext Markup Language (HTML) sources to determine a manageable list of keywords that are commonly related to data portals. The main concept behind this approach is providing an in-depth search method that has not yet been explored instead of superseding the previous proposals.

The motivation and usefulness of this research are outlined by the need of a single, up-to-date, and reliable source of data portals available worldwide. These data portals are supposed to be published by institutions and organizations with respect to the data-oriented initiatives, irrespective of whether they implement standardized open data software platforms or rely on in-house software development for their own data infrastructure. Such a repository will support benchmarking exercises with lesser manual effort and more representative samples because the data initiatives are intended to be previously catalogued. Consequently, the government and policy makers can take advantage of the more accurate details based on the best practices to guide their data-focused initiatives.

The research objective of this study is twofold: first, to develop a method that deeply searches every single web page on the internet to gather information regarding the characteristics of a data portal that implements a standardized open data software platform; second, to formalize a machine learning classification model to improve the discovery of data portals irrespective of whether these data portals implement a standardized open data software platform and comply with the open data technical guidelines. Both objectives are related to each other because the first servers as a pillar for the second.

This contributes to an ongoing project related to an easy-to-reproduce method for searching and identifying data portals from the entire web. The results and findings obtained from the experiment in this study produced relevant insights about understanding the adoption and use of open data software platforms. In addition, they demonstrated the extent of automation at the core of machine-readable data availability and how it can be used to manage the vast amount of data publicly available. Machine learning techniques help with the identification process of a wide variety of data portals, especially those implementing specific developed web-based software, which would be difficult to accomplish with the traditional techniques.

The application of artificial intelligence (AI) in the context of this work aims to face the challenges of identifying data portals that do not implement standardized open data software platforms, but publish open data to generate public value. Such data portals implement their own methods to access the underlying data; the process of identifying them as *data portals* (also differentiating from ordinary web pages) must be based on the adaption to new circumstances as well as the detection and extrapolation of patterns and techniques that are typically embraced by machine learning (Russell & Norvig, 2009). Thus, AI algorithms presuppose learning from the examples of the implementation of open data software platforms to identify specific developed web-based software.

Under this study, we agreed on a list of the commonly used and cited open data software platforms that served as the proof-of-concept for the development of this work and showcase the research approach. This list also considered those platforms that frequently appeared in community-driven projects related to open data. The basics of the selected open data software platforms are discussed in the background section.

Every artifact produced throughout this study—such as computer algorithms, running data, and obtained results—has been openly provided with this paper. Thus, open data researchers and practitioners can reproduce the method, help in increasing its validity, and even improve it. Readers are requested to refer to the dataset of this publication to access the repository[1].

[1] http://dx.doi.org/10.17632/8fr6v9xf6h.1

This work is an expanded version of the paper titled *Laying the foundations for benchmarking open data automatically: a method for surveying data portals from the whole web*, presented in the 20th Annual International Conference on Digital Government Research—dg.o'2019. This expanded version presents a more accurate approach to identify data portals with the design and implementation of a machine learning classification model.

The remainder of this paper is organized as follows. Section 2 presents the most basic concepts to understand this study and its proposal. Section 3 presents and discusses the related work that also served as the basis for this research. Section 4 describes the design and implementation of the method, and Section 5 focuses on the results and discussion regarding the data gathered, in addition to a comparison with previous research. Finally, Section 6 concludes this work.

2. Background

In the following subsection, we provide a background for the essential concepts about open data evaluation, open data software platforms, the Common Crawl project, and machine learning techniques for data portal classification. An understanding of these elements and how they relate to each other is essential to comprehend the methods developed throughout this paper and the basis of the results and findings produced herein.

2.1. Open data portals as software platforms

Not all data portals publish open data. Case studies (A. S. Correa et al., 2019; Corrêa et al., 2017) conducted at the subnational level of the government in Brazil pointed out that a considerable number of data portals seem to be built upon specific developed web-based software lacking essential features necessary to publish open data, which are provided by default by the leading standardized open data software platforms available in the market.

A data portal can implement a standardized open data software platform. A software platform is a shortcut to formalize any data infrastructure because it facilitates the development phase required to build such a software from scratch. An open data software platform is mainly determined in compliance with the open data requirements (Open Data Charter, 2015; Open Government Working Group, 2007; Tauberer, 2014), demanding features such as metadata management, basic visualization, user management, data publishing, data storage, and application programming interface (API) support (European Union, 2017; Lisowska, 2016; Milic et al., 2018). Braunschweig et al. (2012) identified several requirements for open data software platforms that are grouped into categories regarding standardization, API, materialization, integration, and policies. A total of 11 standardized platforms available in the market were compared by Osagie et al. (2015).

An open data software platform is essentially a data container. Therefore, adopters can store any type of data, including those that may reveal private information about individuals, thereby raising concerns about data privacy issues. Green at al. (2017) discussed the benefits of open data despite the risks. As annotated by the authors, it is expected from open data practitioners to publish data at the highest level of granularity; however, this often conflicts with privacy as data may contain personal and sensitive information. Traditional practices include anonymization, which involves identifying and removing personally identifiable information before publishing in open data platforms.

A consolidation of a structured repository of data portals with information regarding the use of software platforms is a part of the objective of this research. Thus, we determined a list of the commonly used software platforms that served as the proof-of-concept for the development of this study. This list was created based on the platforms most cited by the literature, including the community-driven project Dataportals.org that triggered the need for such a repository, and the availability of documentation regarding the use of the APIs of these platforms. Our list included four open data software platforms: 1) Comprehensive Knowledge Archive Network (CKAN), 2) Socrata, 3) OpenDataSoft, and 4) ArcGIS Open Data. Other standardized platforms can be subjected to further investigation if they include APIs to access the data and metadata with enough information on how to use them.

CKAN is probably the most well-known global platform that is trusted by high-load data portals, such as the US Federal Government Open Data Portal (Bolychevsky, 2013) and the European Data Portal. CKAN has been positioned at the top of the list of leading open data software platforms owing to its free and open-source characteristics that are being actively developed since 2007. It is currently supported by Open Knowledge International, a worldwide nonprofit network focused on openness with the use of technology as well as training to unlock information and knowledge sharing. CKAN has also led to the development of a variant called Drupal-based CKAN (DKAN), which basically integrates a content management system.

Socrata is a commercial product focused on the data infrastructure for government institutions whose expertise date from its foundation in 2007. Socrata was previously owned by a company with the same name before its acquisition by Tyler Technologies in April 2018. Its platform provides data-as-a-service and cloud-based solutions specialized in the public sector, and it was considered as the main open data solution before the growth of CKAN installations around the world. The Colombian data portal https://datos.gov.co is an example of the use of Socrata.

OpenDataSoft was founded in 2011 by a start-up company, and it currently has solutions directed at the government and private sectors, with specialization in smart-city support. Open data are only one front of its foci, which also include private data sharing. A considerable number of OpenDataSoft installations can be found in France, while only some can be found in the USA. An example of OpenDataSoft can be found in the City of Brussels (Belgium) https://opendata.brussels.be.

ArcGIS, a product of the Environmental Systems Research Institute (ESRI), has traditionally been recognized for a software suite that deals with maps and geographic information. The open data features are accessible with subscription to its online base platform, offered in the form of software-as-a-service in cloud under the standardized domain *.opendata.arcgis.com; however, dedicated installations can be made on the client site. There is not much information about the introduction of ArcGIS in the open data field; however, the company is a newcomer, as revealed by the lack of official documentation to deal with the product APIs. An example of ArcGIS Open Data includes the City of Rio de Janeiro (Brazil) http://www.data.rio.

2.2. Common Crawl Project

Common Crawl is a nonprofit organization that features a project with the same name, aiming to build and maintain an open repository of web-crawl data freely available for everyone. Companies, researchers, and even individuals can access the crawl data that were previously only available to large search engine corporations.

Common Crawl works similarly to Google and Bing search engines; it crawls the entire internet by following web page links. Its corpus makes this project unique; it contains petabytes of data collected in recent years. The Common Crawl corpus contains a copy of the HTML raw data at a given point in time, including human-readable content and metadata, but without the images, CSS stylesheets, JavaScript files, and other non-HTML contents (Leetaru, 2017). It is useful for large-scale data-science exercises to explore the textual web and reveal information that cannot be easily obtained owing to the enormous size of the internet and its continuous and rapidly changing characteristics.

Currently, crawls are built once a month, and the content is stored using the WARC format, thereby allowing multibillions of web page archives that are hundreds of terabytes in size. The Common Crawl structure includes three types of main files: WARC, metadata (WAT), and text data (WET). The most detailed file (WARC) includes the tracking of HTTPs requests and full respective response (HTML source code) from web pages, along with all metadata, including headers. The WAT files carry on the computed metadata for the main elements stored in the WARC files in the JavaScript Object Notation (JSON) format to make a file as small as possible. The WET files only include plain text information from web pages that are usually shown to users through a browser. Additionally, there exists a more concise file with only URLs to allow a quicker analysis in which web page addresses are sufficient.

In this study, we used WARC files as the baseline to test our deep search method to determine specific keywords in the full corpus of the crawled data. The most challenging part of dealing with the WARC archives is managing the size and quantity of files. To process a single WARC file, a download of approximately 800 MB of compressed content is required, which expands to approximately 5 GB after decompression. Each WARC file contains approximately 55 million lines; an archive from April 2019 contains 56,000 WARC files.

All files from Common Crawl are provided through the Amazon Public Datasets program. Throughout this study, we observed no significant difference in the downloading speed outside the Amazon network, as demonstrated in the next sections. However, it is impractical for most practitioners to store data locally; thus, an algorithm that can process and discard files accordingly is required.

After handling downloads and storage, the rest of the work can be performed using a modern programming language such as Python. This enables the analysis of the whole web content, even for the newcomers in the open data field.

2.3. Machine learning for data portal classification

The popularity of data portals available on the internet introduces new sources of information through the availability of both textual and machine-readable data. Useful insights can be extracted from these sources by implementing the machine learning techniques that are widely used with text analytics (Aggarwal, 2018; Russell & Norvig, 2009).

This research builds a solution for the classification of data portals, above all those implementing specific developed web-based software because of the complexity to identify them. Thus, a machine learning training model considers the representation of prior knowledge to enable the machine to learn from an input, herein defined as a known data portal modeled to feed the machine learning algorithm. Accordingly, known data portals that implement open data software platforms are considered as examples to define and distinguish the set of words present in them to perform the process of learning from examples.

Behind the scenes, a selected machine learning process uses a family of algorithms based on the Bayes theorem. In this study, we rely on a variant called the Naïve Bayes classifier that allows us to classify a web page according to the given set of features extracted from a training data model (essentially, presence of words) and their probability to occur in a document. This classifier assumes independence among predictors, implying that a feature is unrelated to the presence of any other feature; thus, it is called naïve. Practically, a web page can contain buzzwords, such as "open" and "data," while simultaneously containing other ordinary words with no relation to those that clearly indicate open data-related contents. This no dependence between the words makes the Naïve Bayes classifier a simple and powerful choice for text classification, including web pages with text content, as annotated by the literature (McCallum et al., 1998; Ting et al., 2011).

Before using the words present in web pages, a preprocessing phase is required to remove tags and other unimportant content from the HTML source code; words with no or little lexical content are also removed. At the end of the preprocessing phase, a list of highly relevant words is extracted to engineer the features to be used in the machine learning algorithm. These features, in turn, encompass information regarding the presence of a given word within instances (web pages) alongside the label classifying each instance as a data portal.

3. Related work

The existence of an independent, reliable, and up-to-date repository as a single source of data portals operated around the world has already been introduced in a seminal work (A. S. Correa et al., 2018), which also introduced a method to automatically identify open data software platforms and estimate the number of published datasets and geographic location. Previously, working data portals spread over the internet could only be found via a manual process or by relying on the few sparse repositories created by an interested community. Figure 1 illustrates the timeline of related work, from the need of a data portal source to the machine learning classification model based on a deep-search of the entire web proposed by the current state of this research.

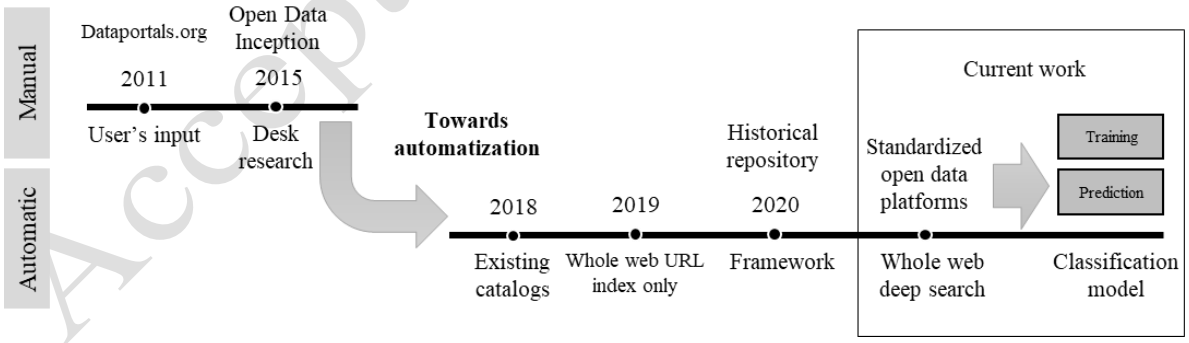

Figure 1. Related work timeline regarding surveying data portals automatically.

In 2011, a group of experts launched a list, which was probably the first initiative to catalog data portals globally, named "Dataportals.org," supported by Open Knowledge International. There seems to be no history about the evolution of the number of cataloged data portals in this source; however, in 2012, the use of 50 entries was reported (Braunschweig et al., 2012). As of September 2019, the project listed 587 data portals and there seem to be no regular updates, which differs from the expectations of the curator. The major problem is probably related to the

process of adding new data portals that is done directly in a repository on Github; this is not a straightforward process. The entry update process follows the same rule as that of adding new ones. Only recently, DataPortals.org allowed us to flag the software platform in use, which used to be a major limitation of this initiative.

The Open Data Inception project[2] is a joint effort by employees from OpenDataSoft, who envisioned improvements on the existing repositories; this project was started in 2015. The authors of this initiative conducted a wide desk-based research to check, clean, and supplement information about the data portals found on the internet. They relied on many different repositories, including DataPortals.org. They reported their experiences in a web page (Mercier, 2015), thereby enabling us to understand how they managed this project, especially the part involving establishing the exact geographical location of each data portal. Data Inception maintains its reputation of being the largest single and manual repository in existence. As of September 2019, a total of 3,314 data portals were listed and available for users to download and work on the underlying data. In previous studies, conducted in 2018 and 2019, we reported 2,814 and 2,844 data portals, respectively. This indicates that the project is under regular development and supervision; however, drawbacks of adding/maintaining data portals and a lack of software platform remained.

The initiatives of DataPortals.org and Open Data Inception served as an inspiration for the proposition of our first paper (A. S. Correa et al., 2018), which attempted to consolidate the existing lists of data portals from seven different sources considered in the paper. It resulted in a list of 3,152 nonduplicated data portal candidates to be checked against the software platforms in use. From this set, the software platforms of 1,104 data portals could be identified, such as CKAN, Socrata, OpenDataSoft, and ArcGIS Open Data. After identification, information regarding the number of published datasets and how institutions managed their update processes as an indicative of the use of open data portals was gathered; additionally, the geographical distributions according to the software platform adopted and number of datasets per portal were demonstrated.

At that point in previous investigation, we suggested an automated and more replicable method, in contrast to manual exercises that demanded rigorous efforts to find, check, and complement the lists of data portals found on the internet. Thus, we envisaged a general web crawler that could find more data portals. However, building a crawler from scratch required computing resources that are not available to most of the applied e-government researchers and even academia networks. This led us to recall the Common Crawl project, the open repository of web crawl data (see a brief introduction in Section 2.2), as a rich source of data from the web that is publicly available.

Later, a new version of the method was proposed using the Common Crawl URL index database as the main source to find potential data portals (A. S. Correa & da Silva, 2019). This improved version worked by searching keywords that occurred frequently in the URLs of data portals (e.g., the word "data"). This insight into the most commonly occurring emerged from the findings of previous studies, where the majority of identified open data portals contained "data" in the URL (e.g., http://europeandataportal.eu/data, https://www.data.gov), in addition to the same word in other languages, to allow wide coverage and then find maximum data portals worldwide. Thus, the translated keywords *datos* (Spanish), *dati* (Italian), *dados* (Portuguese), and *daten* (German) were a part of the search. Consequently, 1,339 data portals were automatically found, demonstrating an increase of 18% in comparison to the previous result. Nevertheless, only 272 (20.3%) of the data portals found were common in both studies, showing the rapid context of changes around open data initiatives that affect the implementation and maintenance of software platforms and how institutions manage their data infrastructure.

Moreover, a work took advantage of previous contributions and developed a framework that envisioned the use of the Common Crawl project on a time basis, repeating the process to build a baseline that the authors introduced as the historical repository. This type of repository supports comparisons between the data collection events, in addition to keeping track of the evolution of data portals available worldwide (A. Correa & Fernandes, 2020).

Such findings reinforced the need for automated methods repeated on an ongoing basis to survey open data portals. In addition, the method that relies on the entire web content was shown to be highly effective because it takes a snapshot of the data portals working on the internet at a given point in time as a workaround of having manual and multiple sources of data portals that cause the following issues.

- Redundancy—distinct sources can have multiple entries for the same data portal. Ensuring no redundancy implies no extra effort to consolidate and contain data.
- Discoverability and updatability—a new data portal can be found once someone inputs its URL in the repository. In addition, updating it depends on personal efforts that may be incompatible with the fast-changing context of open data initiatives. Ensuring that data portals are automatically

[2] https://opendatainception.io

discovered/updated (mainly from the whole web) can help in reducing human interaction and making the method scalable.

- Platform traceability and independency—current repositories do not allow an efficient way to identify the software platforms implemented in data portals. Although standardized platforms share a similar set of features, they work very differently from each other. Identifying them in an independent manner helps practitioners in developing cross-platform methods that will extract the best platforms to support benchmarking.

After determining a road map for surveying open data portals from the entire web, the next concern involves the generalization of the methods introduced in the series to identify ordinary web pages published as data portals that implement engines other than those provided by standardized open data platforms. Briefly, an ordinary web page is published as a data portal when it contains data primarily related to the context of a government; however, it lacks standardized ways to access the underlying data and maybe does not fully comply with the open data principles.

Consequently, a new branch of this work inaugurated the use of AI techniques to predict web pages with specific developed web-based software operated as data portals, based on the text classification extracted from typical examples through open data software platforms. The use of AI in public and government-related domains has increased in recent years owing to the dissemination and popularization of computing techniques among practitioners; however, a study by Weslei Gomes de Sousa et al. (2019) pointed out that it is still insufficient. The same study observed that the most recurrent AI technique involves artificial neural networks, followed by fuzzy logic and machine learning; it also indicated that India, the US, and China exhibit a growing trend of interest in AI in public sector-related domains, mostly applied to general public services.

The AI technique involved in this research is machine learning because it can adapt to new circumstances to detect and extrapolate patterns (Russell & Norvig, 2009). Accordingly, we noticed a lack of applications designed specifically for discovering data portals (van der Waal et al., 2014), which is the aim of the current state of this research. Similarly, we found recent applications relying on machine learning to analyze and predict crimes (Alves et al., 2018; Ku & Leroy, 2014). In accordance with this research, we report a work by Androutsopoulou et al. (2019) that developed an approach based on natural language processing and machine learning to introduce chatbots to improve communication between the government and citizens.

3.1. Benchmarking open data automatically: from theory to practice

It is not sufficient to only publish data to fulfill the open data requirements. Stakeholders usually demand evidence on how effectively the government and institutions implement open data. Although some works (Conradie & Choenni, 2014; Sadiq & Indulska, 2017; Vetrò et al., 2016) discussed and raised concerns about the quantity and its implication on data quality, arguing that it is perhaps difficult to measure open data owing to the lack of theoretical models (Susha et al., 2014); such demands can be satisfied by conducting benchmark exercises based on quantitative evidence (Ulrich et al., 2015).

The literature reports benchmark exercises worldwide that intend to evaluate and rank countries in terms of their practice on disclosing open data. Major examples are Open Data Barometer (Brandusescu et al., 2016), a peer-reviewed expert survey covering 155 countries in the 2016 edition; the Global Open Data Index, covering 94 countries in the 2016/2017 edition of the study; and the United Nations E-Government Survey (United Nations Publications, 2016), an assessment containing specific questions about open data since 2014, covering 194 countries in the eighth edition of the survey. In addition, it reported works (Thorsby et al., 2017; Veljković et al., 2014) that are more focused on benchmark proposals using the data available on the US open data portal, which is one of the biggest data portals in the world.

A white paper about benchmarking open data automatically (Ulrich et al., 2015) introduced the feasibility of conducting automated assessments based on a methodology as a framework called Common Assessment Methods for Open Data (CAF), wherein the first draft was developed by the World Wide Web Foundation in a workshop held in June 2014 (Caplan et al., 2014). The CAF framework only provides a standardized and conceptual overview of four high-level dimensions (Context/Environment, Data, Use, and Impact) that can vary widely in their potential for automation. Given the quantitative nature of data portals, the data dimension focuses on technical openness (e.g., API availability, format, and licensing) and has the highest potential for automation.

Some exercises benchmark the data portals based on a semi-automated approach to obtain the list of data portals to be studied. Traditionally, such exercises impose the need to register data catalogues in advance to enable data collection. Examples include OpenDataMonitor[3] that provides information about dataset consistency automatically obtained from open data portals across Europe; Datashades[4] focuses on the data portals that implement CKAN, promising the index of metadata and statistics on this specific platform across the globe.

In academia, we reported works (Kubler et al., 2018, 2016; Neumaier et al., 2016, 2017; Yang et al., 2015) that explored the automated quality assessment of metadata from open data portals. A series of works (Kubler et al., 2018, 2016; Neumaier et al., 2016, 2017) first reported an assessment of 82 data portals that later increased to 260. Currently, the authors have provided an online tool called Open Data Portal Watch[5] to monitor a set of data portals to perform the analysis. None of these works perform automated surveys of data portals because the underlying URLs are apparently input manually into the system prior to any assessment process.

Such examples call the attention of academia and the interested community to the subject of benchmarking open data automatically, especially on the evaluation of data portals on a larger scale, at a higher frequency, and with less cost to match the rising demands for evidence (Ulrich et al., 2015). Subsequently, improving the evaluation process of open data became an emerging topic that researchers strived to manage. The challenges in this topic include how to handle the vast amount of publicly available data and the rapid context of change around the open data initiatives.

As reported herein, the method developed in this work contributes in reaching some level of automation that tends to make the evaluation of data portals less dependent on manual efforts and more subject to machine processing.

4. Methodology and data collection

The method developed in this research is organized such that it addresses two main problems regarding building and maintaining a viable list of data portals operated globally. First, it searches for standardized software platforms to gather training data extracted from the typical examples of data portals. Second, based on the training data, a machine learning model is generated to distinguish and predict specific developed web-based software operated as data portals. The following subsections discuss the main processes in detail.

4.1. Deep searching the web

If a data portal works on the internet and implements an open data platform among CKAN, Socrata, OpenDataSoft, and ArcGIS Open Data, there is a high chance that it will be discovered throughout the proposed method. As we will describe in this section, the deep-search method is organized into four main processes, and its lifecycle is illustrated in Figure 2.

[3] https://www.opendatamonitor.eu
[4] https://datashades.info
[5] https://data.wu.ac.at/portalwatch

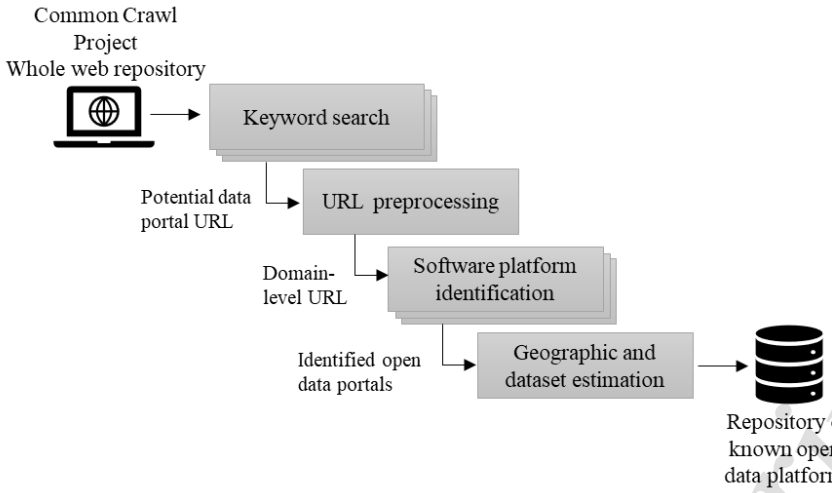

Figure 2. Deep-search method lifecycle.

The method presupposes the access to Common Crawl WARC files that contain the full content of crawled web pages, including the HTML source code. An experiment running the method reported herein was conducted on the top of April 2019 archives, crawled by Common Crawl servers between April 18 and 26. As reported on the project web page, these crawl archives contain 2.5 billion web pages in 198 TB of uncompressed content, split over 56,000 WARC files.

The first process, keyword search, expects an input text file named "warc.paths" that can be found from the Common Crawl project; it contains the paths to all WARC files located in the Amazon Public Datasets repository[6]. This process is conducted in the following four steps: 1) download files, 2) decompress files, 3) iterate through its content while searching the keywords, and 4) write a result file that stores the URLs whose web page matches the keywords. In detail, Step 3) considers a predefined list of keywords such as "open data," "opendata," "ckan," "socrata," "opendatasoft," and "arcgis," which are normally found in data portal web pages and were a part of this experiment. These keywords were selected after a careful analysis of the HTML of each software platform, where we noticed at least one occurrence of respective product reference in the source code. Both "opendata" and "open data" strings were selected based on an internationally agreed list of terms that usually describe an open data portal.

All steps within the keyword search process occur sequentially in the order from one to four for each WARC file, where the processed files are deleted from the machine after execution. This is necessary owing to the size of the entire dataset, which is equal to approximately 44 TB of compressed content. Such a volume was not available for storage for the team responsible for this research.

Keyword search is computationally the most expensive process of the entire method. Thus, we compared the performance of this process to let the community estimate the necessary resources. Table 1 presents the comparison using different tiers of cloud services, considering Google Compute and Amazon EC2, where we conducted a trial of 200 WARC files processed for estimation and comparison purposes. Both companies offer a free starter tier that limits users based on the computer-instance type and processing hours of use. In other words, a Linux-based virtual machine can be run for an entire month for free. Amazon states that its free tier is available for 12 months, and Google states that it is always free within certain configuration.

Table 1. Performance comparison of keyword search for a single virtual machine running at a time. Estimates for comparison based on a trial of 200 WARC files.

	Average time to process				Estimates to complete 56,000 WARC files	
	Download	Decompression	Keyword search	Total	Time	Cost US$
Google free tier	0:00:57.147	0:02:09.083	0:09:00.700	0:12:06.931	11,308 h	-

[6] https://registry.opendata.aws/commoncrawl/

Compute f1-micro 1 vCPU, 0.6 GB RAM					(471 days)	
Amazon free tier EC2 t2.micro 1 vCPU, 1 GB RAM	0:00:58.815	0:05:18.694	0:16:16.836	0:22:34.347	21,068 h (878 days)	-
Google large instance Compute 1 vCPU, 9 GB memory	0:00:00.181	0:00:19.397	0:02:21.570	0:02:41.148	2,507 h (104 days)	0.061/h = 152,92
Amazon large instance EC2 r5.large 2 vCPU, 16 GB memory	0:00:00.180	0:00:11.898	0:01:36.273	0:01:48.351	1,685 h (70 days)	0.126/h = 212,31

According to Table 1, running the process on a free tier with a single virtual machine at a time would significantly lengthen the process (at least 471 days in the case of Google). It seems unreasonable to wait for more than a year to process a single month of crawl archives. Thus, the full experiment of this study was conducted on many large instances from both Google and Amazon; the time and cost estimates are also included in Table 1 for comparison. Using a larger instance allows the process to run in a memory-optimized mode; additionally, it enables segmentation to process more than one WARC file at a time. Different versions of the algorithm are provided alongside this paper to take advantage of a larger amount of available memory (more than 8 GB) and process many segments simultaneously. Algorithms are also adapted to take advantage of direct access to Amazon S3 buckets (boto3 library) if the Common Crawl archives are processed on the Amazon virtual machines.

After the execution of the keyword search process, all full URLs gathered with the potential data portals are supposed to be shortened until they reach the domain level to avoid numerous versions that direct to the same web address. A previous study (A. S. Correa & da Silva, 2019) demonstrated that domain-level URLs are more effective in obtaining the endpoints of open data portal in the subsequent process. Thus, the URL preprocessing method yields a list of domain-level URLs.

The software platform identification process consists of querying all domain-level URLs by making HTTP GET requests specifically to the four main considered software platforms, i.e., CKAN, Socrata, OpenDataSoft, and ArcGIS Open Data. Each platform is designed to respond to a requester with a predefined message through the open standard JSON; they work in the context of the Representational State Transfer API (RESTful API) architecture. The messages in the JSON format returned by platforms are interpreted as mapped signatures to uniquely identify a platform, as outlined in Table 2 (A. S. Correa & da Silva, 2019). This process finishes with a list of data portals with identified platforms.

Table 2. Mapped signatures for software platform identification.

Platform signature (API request)	JSON expected structure response	Point of verification
CKAN: endpoint + /api/3	{ "version": 3 }	Existence of a pair with a key named “version” and a value “3”
Socrata: endpoint + /api/catalog/v1	{ "results": [], "resultSetSize":, "timings": {} }	Existence of an array called “results”
OpenDataSoft: endpoint + /api/v2	{ "links": [] }	Existence of an array called “links”
ArcGIS Open Data: endpoint + /api/v2	{ "datasets": {}, "items": {},	Existence of a member called “datasets”

"groups": {},
"sites": {},
"organizations": {},
"pages": {},
"params": {}
}

The last process, geographic and dataset estimation, aims to gather information about the number of published datasets in each data portal and attempts to determine the country that it is related to. For dataset estimation, each software platform provides APIs to reveal the number of published datasets. Here, we refer to our previous paper (A. S. Correa & da Silva, 2019) wherein we designed four different methods, summarized in Table 3.

Table 3. API requests and narrowing parameters for dataset estimation.

Platform and API request(s)	Narrowing parameters
CKAN	rows=1
(1) /api/action/package_search	
Socrata	only=dataset
(1) /api/catalog/v1	domains=
	search_context=
OpenDataSoft	rows=1
(1) /api/v2/catalog/datasets	
ArcGIS Open Data	
(1) data.json	-
(2) /api/v2/datasets/{:id}	-
(3) /api/v2/datasets	filter[owner]=
	page[size]=1

We designed two methods to help the process of geographic estimation. The first and more precise method navigates through the Country-Code Top-Level Domain (ccTLD) extracted from a URL whenever possible. For example, www.data.gov does not provide any clue about its related country (United States), but www.dados.gov.br explicitly indicates "br," which stands for "Brazil." The second method is through the Internet Protocol (IP) location, which attempts to obtain the country where a data portal is hosted.

At the end of the process, a checked list of data portals is produced as a repository with standardized open data software platforms, which will be used to make the training sets feed into the machine learning algorithm.

4.2. Generating a Machine Learning Classification Model

Section 4.1 provided a direction for surveying and obtaining a list of working data portals on the internet. In addition to being a highly effective method to work with information from the entire web, this approach demonstrates a limitation of finding data portals only from the four considered software platforms (CKAN, Socrata, OpenDataSoft, and ArcGIS Open Data). Occasionally, if a data portal implements a software platform other than those considered, it would not be found. This can be explained by what and how the APIs of those unknown platforms or web-based software provide access to the underlying data. Therefore, in the next step, we explore the problem of identifying data portals, irrespective of whether they implement standardized open data platforms or specific developed web-based software.

However, there are no out-of-the-box solutions just because we do not know how the specific developed web-based data portals work. This can be overcome by accessing these data portals—published as web pages—and visually checking whether they look like a data portal; for example, providing datasets and search mechanisms, and offering multiple data formats. The decision to consider a web page as a data portal lies in what content it discloses. Thus, a possible solution is to treat those web pages as documents (corpus) and rely on classification, which is the task of selecting the correct label (*Data portal* or *Non-data portal*) for a given input of a typical web page corpus. Fortunately, the classification of text—better described herein as supervised classification—is a typical problem solved by natural language processing by employing machine learning techniques, which is the focus of this section for automating the process.

Figure 3 outlines the machine learning classification process employed to decide whether a given web page refers to a data portal. The entire process encompasses three basic steps defined as 1) Extract, 2) Train, and 3) Predict. The Python codes used to perform the process are supplied with this paper repository.

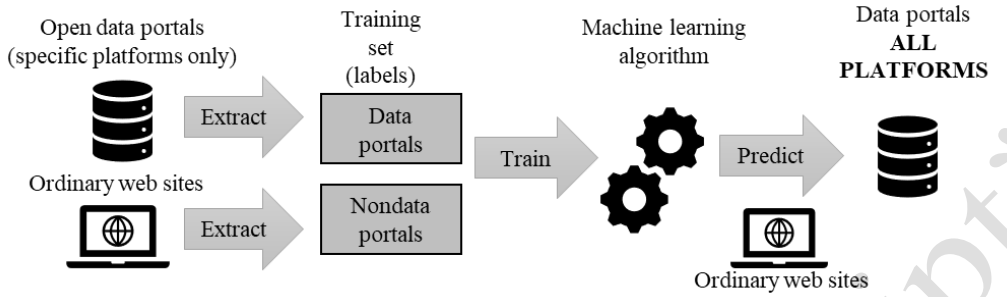

Figure 3. Machine learning classification process.

The first step, extract, involves the extraction of pieces of information (known as features), taken from web pages as examples of both data and non-data portals. Features include words and information regarding their occurrence, ignoring the word order and sentence construction. A machine learning algorithm will work with these features to predict whether a given non-classified web page is a data portal based on the occurrence of words that were previously classified in the training model.

As detailed in Section 4.1, the web pages classified as data portals were obtained from a list of 1,832 known URLs (1,650 were not duplicated) that were served to input into the training set as examples of typical data portals, which certainty is attained in the API verification process previously conducted. In contrast, a source with examples of non-data portals (ordinary web pages) to input into the training set was obtained through random requests made in the Common Crawl URL index archives, whose web addresses were selected based on the absence of words that commonly identify a data portal.

Both sets with data and non-data portals had their content extracted through a process that scraped the respective web pages to extract meaningful text from HTML and remove the tags and other unimportant content from the source code using the Beautiful Soap library (Nair, 2014). After extracting the plain text, the next process removes the high-frequency words with no or little lexical content for features. These words are known as stopwords, and the Natural Language Toolkit (NLTK) library (Bird & Klein, 2009) offers a method to filter them. The results of this preprocessing phase with the corpus that served as the training set are available with this paper repository at this link[7].

The languages within the training set are worth noticing because they impose extra complexity in the algorithm. Web pages taken from the entire web in multiple languages and the resulting machine learning process should be able to handle such a multilingual complexity. In this experiment, we conducted tests with a monolingual training set (e.g., English, the most frequent language), but the results were not satisfactory. Thus, the training set considered in this work includes entries in many languages in a way that they are found on the internet with added metadata to describe which language each one refers to.

The method used to identify languages was implemented in three different ways to handle the inaccuracies when retrieving text from web pages. The first method attempts to identify a language from the HTML source code by searching for the *lang* tag. Problems involving this method include web pages that do not expose the tag or provide it with unexpected content, and those that offer on-the-fly translation (e.g., ArcGIS-based data portals); thus, the language is undesirably changed. The second method attempts to predict the language by comparing the extracted words with the stopwords contained in the NLTK corpora. Problems include the impossibility to identify languages other than the 21 currently available. The third method implements the Langdetect[8] Python library to detect 55 languages using Google Translation Services.

The obtained plain text is divided into tokens of words that enable us to check the extent of repetition in each token. This information is necessary for preparing the frequency table that will be used to calculate the occurrence of

[7] http://dx.doi.org/10.17632/8fr6v9xf6h.1#folder-c090d676-fcb4-4462-863e-733c442e15d5
[8] https://github.com/Mimino666/langdetect

words for a given label (*Data portal* or *Non-data portal*) composing the features set. Consequently, the training set contains pairs of features set and labels to feed into the machine learning algorithm and generate a training model.

The second step, train, consists of using a probabilistic classifier—in this case, the Naïve Bayes classifier—to find the probability based on the contribution of each feature to estimate the likelihood of a label defined in the training model. To demonstrate how the process works with the related words used in the context of this research, only for illustrative purposes, Figure 4 shows a corpus built with only six words in the features set.

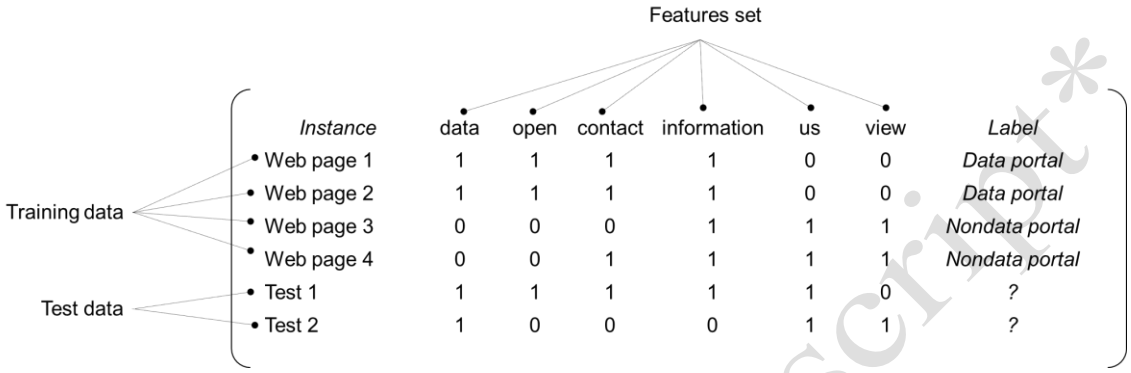

Figure 4. Corpus of features set—only for illustrative purposes.

In Figure 4, the first row contains the words "data," "open," "contact," "information," "us," and "view," which are supposed to be a part of the list of the most frequent words extracted from data portals with standardized open data software platforms. The first column specifies four training web pages and two test web pages, called instances. The label defining each instance as a data portal is shown in the final column for only the first four rows. The labels of the last two rows are missing, and they are expected to be classified by the algorithm. In the following, we reproduce a step-by-step method to predict the label of *Test 1* and *Test* 2 instances by considering the algorithm presented by Aggarwal (2018) for Naïve Bayes Multinomial.

First, we must examine the training data to calculate the prior probabilities *P(Data portal)* and *P(Non-data portal)*. A total of four web pages can be observed within the training data; two of them are labeled as *Data portal* and the other two are labeled as *Nond-ata portal*. Thus, we have

$$P(Data\ portal) = \frac{Number\ of\ web\ pages\ with\ Data\ Portal\ label}{Total\ number\ of\ web\ pages\ with\ Data\ Portal\ label} = \frac{2}{4} = \frac{1}{2},$$

$$P(Nondata\ portal) = \frac{Number\ of\ web\ pages\ with\ Nondata\ Portal\ label}{Total\ number\ of\ web\ pages\ with\ Nondata\ Portal\ label} = \frac{2}{4} = \frac{1}{2}.$$

The multinomial model assumes the word frequency. Thus, a matrix containing the occurrences of each word as shown in Figure 5-a can be produced. Next, we must compute the multinomial parameters according to the number of occurrences of each word. As shown in Figure 5-b, there are two words "data" out of a total of eight words in the training data for the label *Data portal*, and this formulates 2/8 plus Laplacian smoothing (1/6) because there are six words in the corpus, resulting in 3/14. Without smoothing, the parameters resulting in zero would cause inconvenience (nullify the equation) in calculating the probabilities in the next step.

(a)

	data	open	contact	information	us	view	Total
Data portal	2	2	2	2	0	0	8
Nondata portal	0	0	1	2	2	2	7

(b)

	data	open	contact	information	us	view
Data portal	$\frac{2+1}{8+6}=\frac{3}{14}$	$\frac{2+1}{8+6}=\frac{3}{14}$	$\frac{2+1}{8+6}=\frac{3}{14}$	$\frac{2+1}{8+6}=\frac{3}{14}$	$\frac{0+1}{8+6}=\frac{1}{14}$	$\frac{0+1}{8+6}=\frac{1}{14}$
Nondata portal	$\frac{0+1}{7+6}=\frac{1}{13}$	$\frac{0+1}{7+6}=\frac{1}{13}$	$\frac{1+1}{7+6}=\frac{2}{13}$	$\frac{2+1}{7+6}=\frac{3}{13}$	$\frac{2+1}{7+6}=\frac{3}{13}$	$\frac{2+1}{7+6}=\frac{3}{13}$

Figure 5. (a) Computing the occurrences of each word. (b) Computing the probabilities of each multinomial parameter.

With the estimated parameters, we performed predictions by multiplying the prior probability already known (1/2) with the calculated parameters, considering the instances *Test 1* and *Test 2* frequency vectors, [1,1,1,1,1,0] and [1,0,0,0,1,1], respectively, as the exponents of the probabilistic parameters for each word, as shown in Figure 4 with word frequency. Consequently, we have the following:

$$P(\textit{Data portal}|\textit{Test 1}) = P(\textit{Data portal}).P(\textit{data}|\textit{Data portal}).P(\textit{open}|\textit{Data portal}).P(\textit{contact}|\textit{Data portal}).\\ P(\textit{information}|\textit{Data portal}).\ P(\textit{us}|\textit{Data portal}).P(\textit{view}|\textit{Data portal})$$

$$P(Data\ portal|Test\ 1) = \frac{1}{2}\cdot\left(\frac{3}{14}\right)^1\cdot\left(\frac{3}{14}\right)^1\cdot\left(\frac{3}{14}\right)^1\cdot\left(\frac{3}{14}\right)^1\cdot\left(\frac{1}{14}\right)^1\cdot\left(\frac{1}{14}\right)^0 = 0{,}000075,$$

$$P(\textit{Nondata portal}|\textit{Test 1}) = P(\textit{Nondata portal}).P(\textit{data}|\textit{Nondata portal}).P(\textit{open}|\textit{Nondata portal}).\\ P(\textit{contact}|\textit{Nondata portal}).P(\textit{information}|\textit{Nondata portal}).\\ P(\textit{us}|\textit{Nondata portal}).P(\textit{view}|\textit{Nondata portal})$$

$$P(Nondata\ portal|Test\ 1) = \frac{1}{2}\cdot\left(\frac{1}{13}\right)^1\cdot\left(\frac{1}{13}\right)^1\cdot\left(\frac{2}{13}\right)^1\cdot\left(\frac{3}{13}\right)^1\cdot\left(\frac{3}{13}\right)^1\cdot\left(\frac{3}{13}\right)^0 = 0{,}000024,$$

$$P(\textit{Data portal}|\textit{Test 2}) = P(\textit{Data portal}).P(\textit{data}|\textit{Data portal}).P(\textit{open}|\textit{Data portal}).P(\textit{contact}|\textit{Data portal}).\\ P(\textit{information}|\textit{Data portal}).\ P(\textit{us}|\textit{Data portal}).P(\textit{view}|\textit{Data portal})$$

$$P(Data\ portal|Test\ 2) = \frac{1}{2}\cdot\left(\frac{3}{14}\right)^1\cdot\left(\frac{3}{14}\right)^0\cdot\left(\frac{3}{14}\right)^0\cdot\left(\frac{3}{14}\right)^0\cdot\left(\frac{1}{14}\right)^1\cdot\left(\frac{1}{14}\right)^1 = 0{,}000547,$$

$$P(\textit{Nondata portal}|\textit{Test 2}) = P(\textit{Nondata portal}).P(\textit{data}|\textit{Nondata portal}).P(\textit{open}|\textit{Nondata portal}).\\ P(\textit{contact}|\textit{Nondata portal}).P(\textit{information}|\textit{Nondata portal}).\\ P(\textit{us}|\textit{Nondata portal}).P(\textit{view}|\textit{Nondata portal})$$

$$P(Nondata\ portal|Test\ 2) = \frac{1}{2}\cdot\left(\frac{1}{13}\right)^1\cdot\left(\frac{1}{13}\right)^0\cdot\left(\frac{2}{13}\right)^0\cdot\left(\frac{3}{13}\right)^0\cdot\left(\frac{3}{13}\right)^1\cdot\left(\frac{3}{13}\right)^1 = 0{,}002048.$$

By simplification and normalization, it can be demonstrated that the probabilities of the *Data portal* and *Non-data portal* labels for *Test 1* are approximately 0.76 and 0.24, respectively, and those for *Test 2* are approximately 0.21 and 0.79, respectively, as detailed in the following.

$$P(Data\ portal|Test\ 1) + P(Nondata\ portal|Test\ 1) = 0{,}000075 + 0{,}000024 = 0{,}000100$$

$$\frac{P(Data\ portal|Test\ 1)}{0{,}000100} = 0.76, \qquad \frac{P(Nondata\ portal|Test\ 1)}{0{,}000100} = 0.24$$

$$P(Data\ portal|Test\ 2) + P(Nondata\ portal|Test\ 2) = 0{,}000547 + 0{,}002048 = 0{,}002595$$

$$\frac{P(Data\ portal|Test\ 2)}{0{,}002595} = 0.21, \qquad \frac{P(Nondata\ portal|Test\ 2)}{0{,}002595} = 0.79$$

The presence of words "data," "open," and "contact" determines the classification of *Test 1* as *Data portal*, to which frequency was found in both existing web pages in the training data classified as data portals. In contrast, the words "us" and "view," commonly found in ordinary web pages, determine the classification of *Test 2* as *Non-data portal*, apart from the existence of the word "data," usually found in data portals. As evident, the algorithm considers independent features that seemed to present reasonable results throughout the calculation of the probabilities, given an entire corpus.

Following an example extracted from the findings of this work, the word "data" was the most common word found in open data portals; it occurs in nearly 80% of the web pages labeled as *Data portal* and in approximately 9% of the web pages labeled as *Non-data portal*. Thus, the likelihood score will be multiplied by 0.80 for the *Data portal* label and by 0.09 for the *Non-data portal* label. The overall effect comprises a dramatic reduction in the score of the *Non-data portal* label with this feature (word "data") owing to the low frequency of the word in this type of web page defined in the training model. In this oversimplified example, there exists a single word "data" in the features set; however, the current method establishes a combination of 2,000 different features to train the model. In Section 5, we discuss how we produced a list of 2,000 most common words to better fit the features set.

The third and last step, predict, aims to determine the label that should be applied for a given input that has not yet been classified—called test set. In this step, the algorithm decides whether a web page is a data portal and learns how to generalize the model to new examples. The machine learning algorithm takes decisions based on a probabilistic model, where there is room for mistakes. This implies that a web page can be misclassified by producing false positive and false negative. The best approach is to check whether the algorithm performed well, but it is not as straightforward as it seems. To verify the accuracy of the results, they should be checked manually, which would require unreasonable efforts. This can be performed on a small sample to reduce the efforts; however, this sample cannot guarantee 100% accuracy.

In this experiment, we relied on two different samples to test the accuracy of the classification model. One of them is a list of 3,502 URLs manually gathered in a previous work (A. S. Correa et al., 2018) from seven different sources. Those URLs have a remarkably high chance to reveal data portals, as someone in the past indicated them accordingly. If we input this dataset into the algorithms, the number of data portals classified by the machine should be high as well. The second dataset contains 38,350 web pages randomly selected such that there is no clue related to whether they are data portals. Based on our experience, such a sample should be classified with very few data portals.

5. Results and discussion

This section depicts the results from processing approximately 2.5 billion web pages from the April 2019 Common Crawl archives to identify data portals based on standardized software platforms. In accordance with these results, we extended a machine learning classification model and produced results on identifying specific developed web-based software operated as data portals. The main quantitative results are summarized in Table 4.

Table 4. Main quantitative results.

	Total
Web pages in April 2019 crawl archives	2.5 billion

Potential data portals (web pages with matched keywords)	25.6 million
Domain-level URLs	977,567
Identified open data portals	1,832
Dataset estimation error—false positives (excluded)	42
Duplicate entries (excluded)	140
Nonduplicate data portals (main result)	1,650
ccTLD (domain country) identified	480

From the set containing all 2.5 billion crawled web pages, the number that presented the matched keywords seemed to be significant (25.6 million or 1% of the entire set). We realized that such a number was obtained owing to the high incidence of texts with "open data" in the corpus and words with "ckan" in their composition string, thereby resulting in several false positives. The web pages with potential data portals were reduced to a set of 977,567 web pages after shortening to domain-level URLs; therefore, different web pages that pointed to the same root address were removed. To illustrate the significance of this process, the US Federal Government Open Data Portal was caught 10,873 times in the crawl archives. After shortening to domain-level URLs, only one remained, which is the desired endpoint of the respective CKAN instance (https://catalog.data.gov). This implies that all paths underneath https://catalog.data.gov/... (after single slash) were discarded because they did not take the desired software endpoint.

Each potential data portal URL was visited at least four times to identify the open data software platform being employed. Subsequently, 1,832 URLs were identified. It was still possible to obtain false positives from this set owing to the presence of undesired URLs that respond to the APIs.

Accordingly, the last 42 false positives were removed from the results after estimating the number of datasets because false positives do not respond to specific APIs exclusively designed for this function.

A total of 140 duplicated entries were also removed from the list. These duplicates can be attributed to the URL versions with "http://" and "https://" prefixes. Additionally, there were portals that also exposed their addresses with and without the "www" prefix, resulting in duplicate entries that pointed to the same data portal endpoint.

We identified different domain-level URLs that seemed to be a part of the same open data initiative. Some of these URLs could be easily recognized by observing the domain-level URL likeness (e.g., "https://data.alberta.ca" as an alias of "https://open.alberta.ca"). Others were more difficult to recognize, and we had to check whether the number of published datasets was the same in addition to the visual patterns of the web pages (e.g., "http://www.hetor.it" and "http://open.databenc.it"). The process to identify these duplicates can also be potentially automated; however, it is beyond the scope of this study. A complementary study is required to propose specific methods to determine the likenesses based on some characteristics, such as the number of published datasets and HTML source code similarity. At the current state of this research, where automation is at the core, we considered all found versions as different data portals to avoid human intervention for checking them.

The repository produced contains a list of 1,650 working open data portals, whose software platforms were identified and checked against the core APIs to obtain the number of published datasets when available.

Tests for estimating the geographic location demonstrated that only 29% (480) of the data portals successfully passed through ccTLD (domain country); additionally, their locations were obtained with the most accurate method. From this set, wherein ccTLD was not available, the IP location was determined and the location of the majority of data portals (1,166, 71%) was returned.

A crucial metric to ensure the efficiency of the method, which has been improved, is its ability of finding more data portals. Table 5 makes a comparison with previous studies, focusing on the total number of data portals found. In the current state of this research, a total of 1,650 nonduplicate data portals were found, exhibiting an increase of 23.2% in comparison with the last study. This means that the current method found more data portals by searching the detailed crawl archives.

Table 5. Total number of data portals found in comparison with previous study.

	2018	2019	Current work
Data portals found	1,104	1,339	1,650
Growth in comparison with last study	-	21.3%	23.2%

In common with current work	255	1,089	-
New data portals not in previous studies	-	1,087	510

Table 6 presents another dimension of the comparison, focusing on the software platform and published datasets.

Table 6. Total number of data portals and datasets by platform: a comparison with previous studies.

		Work	CKAN	Socrata	OpenDataSoft	ArcGIS Open Data	Total
Data portals		2018	185	132	39	748	1,104
		2019	351	201	167	620	1,339
		Current	439	255	143	813	1,650
Average of datasets		2018	9,952	226	205	56	1,740
		2019	7,865	225	356	97	2,185
		Current	5,967	228	425	104	1,711
Data portals per total of datasets range	=0 or NA	2018	10	16	0	**242**	268
		2019	1	16	0	21	38
		Current	1	12	1	49	63
	1–10	2018	8	9	3	186	206
		2019	29	18	30	68	145
		Current	36	28	14	112	190
	11–100	2018	44	**60**	**19**	225	348
		2019	113	**101**	**84**	**375**	673
		Current	161	**135**	**83**	**462**	841
	101–1,000	2018	**71**	42	15	90	218
		2019	**134**	59	48	152	393
		Current	**165**	73	39	176	453
	1,001–10,000	2018	33	5	2	5	45
		2019	51	7	3	4	65
		Current	47	5	4	14	70
	>=10,001	2018*	19	0	0	0	19
		2019*	23	0	2	0	25
		Current	29	2	2	0	33

Values printed in bold and underlined indicate the highest concentration of data portals in the respective platform and dataset range.

*Originally, in the works performed in 2018 and 2019, the total number of data portals with more than 10,000 datasets was not listed. Thus, the values were adjusted to be compared with the current work.

In terms of software platforms, CKAN and Socrata showed an increase in the number of data portals found in the three editions of the series, while ArcGIS Open Data only increased in comparison with last year. Data portals with CKAN, Socrata, and ArcGIS Open Data grew by 25.1%, 26.9%, and 31.1%, respectively. However, even though OpenDataSoft showed a significant increase in 2018/2019, in this study, it decreased by 14.4% in terms of the number of data portals.

The average number of published datasets per data portal was 21.7% smaller than that in the last study. This global average was affected by the increment in the number of data portals found.

Except for ArcGIS Open Data, the remaining software platforms were in the same range of total number of datasets as in the previous two studies. However, if we only consider the study conducted in 2019, all software platforms were in the same range of total number of datasets. Thus, in comparison to 2019, only CKAN had a significant number of data portals in the range of 101 and 1,000 datasets per data portal; Socrata, OpenDataSoft, and ArcGIS Open Data were in the range of 11 and 100 datasets each.

By comparing to 2019, we noticed an increase in the number of ArcGIS Open Data-based data portals with unavailable datasets (Table 6 column "=0 or NA"); this can be attributed to a set of data portals that inhibited the algorithms to perform the sequence of three API calls necessary to determine the number of published datasets. The first call for "/data.json" returned an empty set of datasets in JSON, which impeded to call subsequent APIs; however, the number of datasets could be checked through the web page interface. We did not make any manual adjustments in the values, and we are investigating why this occurs with a few installations of the platform.

After obtaining the list of the identified open data portals, the training set was built and fed into the machine learning algorithm. Because this technique involves natural language processing, the language of words contained in the features set became an important element that directly affected the results. As briefly mentioned before, we decided to take a multilingual approach to increase the chances of the learning process to generalize the model to new examples, irrespective of the language of the input. Accordingly, we obtained the training set detailed in Tables 7 and 8, which emphasize the languages obtained across the three different methods for each label.

Table 7. Top five most frequent languages in the corpus of the training set labeled as open data portals.

Language	Method 1	Method 2	Method 3
English	1,236 (70.4%)	514 (29.3%)	1,192 (67.9%)
French	138 (7.9%)	105 (6.0%)	155 (8.8%)
Spanish	115 (6.5%)	62 (3.5%)	104 (5.9%)
Portuguese	70 (4.0%)	886 (50.5%)	68 (3.9%)
Italian	44 (2.5%)	40 (2.3%)	44 (2.5%)
Others	134 (7.6%)	127 (7.2%)	189 (10.8%)
Not detected	19 (1.1%)	22 (1.3%)	4 (0.2%)
Total number of data portals		1,756 (100%)	

Table 8. Top five most frequent languages in the corpus of the training set labeled as non-data portals.

Language	Method 1	Language	Method 2	Language	Method 3
English	992 (56.5%)	English	652 (37.1%)	English	849 (48.3%)
Russian	173 (9.8%)	Russian	88 (5.0%)	Russian	155 (8.8%)
French	85 (4.8%)	French	55 (3.1%)	Chinese	78 (4.4%)
Dutch	63 (3.6%)	Dutch	50 (2.8%)	French	74 (4.2%)
Romanian	58 (3.3%)	Czech	46 (2.6%)	Korean	68 (3.9%)
Others	264 (15.0%)	Other	311 (17.7%)	Other	506 (28.8%)
Not detected	122 (6.9%)	Not detected	555 (31.6%)	Not detected	27 (1.5%)
Total number of web pages			1,757 (100%)		

As demonstrated in Table 7, the training set labeled as open data portal contains 1,756 examples of data portals. These data portals were obtained after executing the previous deep-search method, which resulted in 1,832 identified platforms. We could not use the exact number of platforms found (1,832) because some of them were no longer functional, probably due to the interruption of the respective data portals. Additionally, in this experiment, we decided to not discard the duplicates to keep the resulting dataset in its original condition as obtained from the previous deep search method. In this regard, the existing few duplicate entries should not impair the desired results.

However, in Table 7, the top five languages found by the three different methods remained the same (English, French, Spanish, Portuguese, and Italian). We noticed a significant difference in the number of data portals in Portuguese and English found by Method 2, which extracts languages from the HTML source code. This can be explained by the on-the-fly translation feature present in the ArcGIS Open Data-based data portals. We noticed that this platform attempts to render text based on the requester's browser language. In our experiment, Portuguese was prominent because we collected data from servers configured in this language; thus, the HTML returned code pages in Portuguese instead of the original language, which is primarily English. This result can be understood in conjunction with the significant number of ArcGIS Open Data installations found, as presented in Table 6.

The training set labeled as *Non-data portal* contains 1,757 examples of ordinary web pages, as shown in Table 8. The URLs of these web pages were randomly retrieved from the Common Crawl URL index archives considering that the respective web address strings did not contain words that commonly identify an open data portal. Because the web pages were randomly retrieved with no control on the languages they referred to, the top five most frequent

found languages diverged. English, Russian, and French shared a ranking across three different methods, with a highlight of English that appeared most frequently.

The features set was defined with the 2,000 most common words extracted from the open data portals. The features were encoded using a simple Boolean value representing *True* or *False* according to the presence of a word in the training set. Therefore, we observed a pattern of frequent words that guided the learning model to differentiate between the labels accordingly. Figure 6 illustrates this mechanism by demonstrating the distribution for the most relevant words (first 20) in the features set.

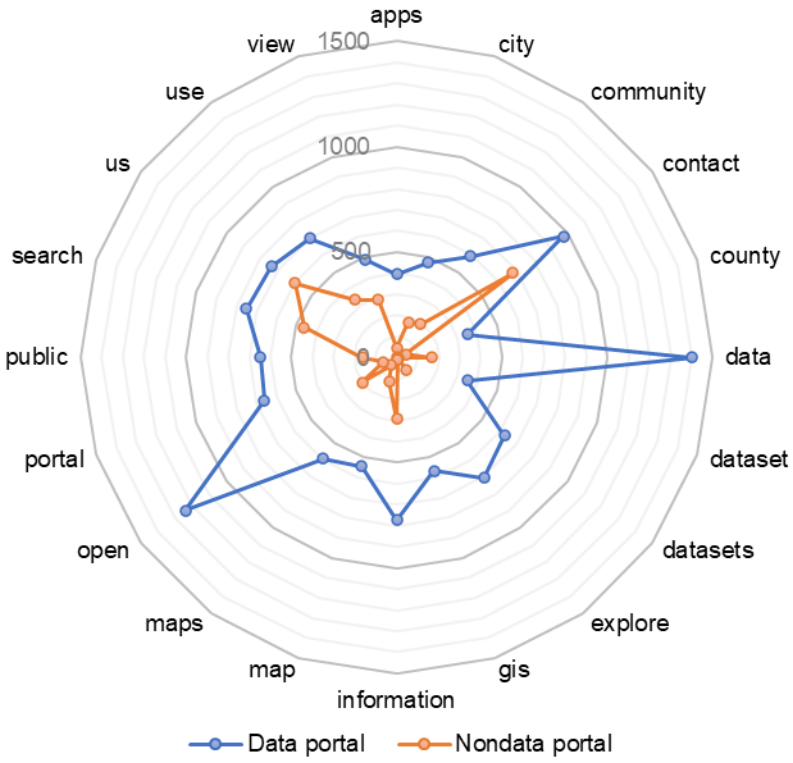

Figure 6. Distribution for the most relevant words in the features set.

As shown in Figure 6, words such as “data” and “open” appeared 1,402 and 1,239 times, respectively, in the corpus of data portals, and the same words appeared 164 and 203 times, respectively, in the corpus of web pages with the opposite label. This implies that there are at least six to eight times more chances of finding the words “data” and “open” associated with an open data portal. It seems obvious, but when we analyze a myriad of words extracted from both data and non-data portals (at first sight with no sense), every pattern is useful and relevant in composing the features set. Moreover, the generated training model considers a combination of the probabilities of all 2,000 words defined in the features set. In Figure 6, a reduced number of features can be observed, which are in English, but the learning model can extend up to the entire list of features given their frequency. Consequently, the more is the training set the more accurate is to generalize the model to new examples in other languages.

The selection of relevant features has an enormous impact on the learning model. In terms of the features (words) that should be contained within a model, we were sufficiently sure about the words extracted from the data portals because this work designed an accurate method to determine the checked data software using standardized platforms with relevance to the problem. However, the size of the features, in terms of how many words it should cover, was selected by trial and error, guided by intuition, followed by a series of tests performed. Accordingly, we consolidated Table 9 by describing the trials performed and data portals found by the designed machine learning classification model.

Table 9. Trials performed and data portals found by the designed machine learning classification model.

Trial #	Count of words (features)	Data portals found	
		Sample 1 (data portals)	Sample 2 (ordinary websites)
1	500	1,337 (38.2%)	592 (1.5%)
2	1,000	1,430 (40.8%)	212 (0.6%)
3	2,000	1,876 (53.6%)	279 (0.7%)
4	3,000	1,735 (49.5%)	404 (1.1%)
5	5,000	2,290 (65.4%)	652 (1.7%)
6	10,000	2,784 (79.5%)	11,365 (29.6%)
Total number of tested URLs		3,502 (100%)	38,350 (100%)

As shown in Table 9, we report the results from six trials with inputs ranging from 500 to 10,000 words on two selected samples. *Sample 1* comprised a total of 3,502 URLs manually gathered in a previous work from seven different sources with a very high chance to take to data portals. *Sample 2* contained 38,350 web pages that were randomly selected such that they demonstrated no identification of being a data portal. We did not manually check the entries contained in the samples; thus, there is no information about the accuracy, but the tests and error analysis were guided by the perception of the nature of samples. Consequently, *Sample 1* should contain many data portals and *Sample 2* should contain very few web pages labeled as data portals.

According to the trials reported in Table 9, we noticed harmonic results with 1,000 and 2,000 words in the features set. Tests with 1,000 words could classify 1,430 data portals representing 40.8% of *Sample 1* and 212 data portals or 0.6% of *Sample 2*. Similarly, tests with 2,000 words were able to classify 1,876 data portals or 53.6% of *Sample 1* and 279 data portals or 0.7% of *Sample 2*. Thus, the best results indicated a proportion of one data portal found in *Sample 2* for 6.7 data portals found in *Sample 1* for both trials, which enabled us decide to work with a features set with 2,000 words. In contrast, we can verify that if the features set contains more than 10,000 words, it can lead to overfitting, which causes an algorithm to have a higher chance of relying on the idiosyncrasies of the training data, instead of appropriately generalizing the new examples.

The results discussed herein contribute and suggest some implications for open data, government organizations, AI, and intelligent automation. For open data, the feasibility to create a repository with a worldwide list of data portals opens the way for an entire new source for automation. Mainly, manual benchmarks can improve this by relying on a broader variety of evidence automatically collected from such a repository. As the results of benchmarking exercises tend to emphasize the strengths and sometimes reveal the weaknesses, government institutions and organizations can base their open data action plans on the best practice models run by others. For AI, the use of machine learning for website classification in the field of data portals inaugurates a new domain of applications where the underlying techniques showcase their effectiveness and potential for novel research. Finally, for intelligent automation, the use of massive sources of web data, as demonstrated using the Common Crawn project, shows fair practical applications even for newcomers to the domain of web data retrieval.

6. Conclusion

This study is part of a series of works intended to build an independent, reliable, and up-to-date repository as a single source of data portals operated globally. The intended repository is expected to be built and maintained based on the data availability from the entire web whose automation is at the core. To ensure this, we have been working toward improving the method to survey data portals automatically and then addressing the two aspects involved in the objective of this research: searching for data portals that implement standardized open data software platforms and identifying specific developed web-based software operated as data portals, whose data openness depends on their design.

To overcome the problem of searching for standardized open data software platforms, the current state of this research demonstrated major improvements in comparison with previous works (A. S. Correa et al., 2018; A. S. Correa & da Silva, 2019), wherein an increase in the number of data portals was automatically observed. This approach is built on a previous method where we introduced a way to identify the main open data software platforms through mapped signatures; however, it is excessively reliant on a consolidated list of seven different manual sources. In a later proposal, we superseded the manual sources by introducing the method of surveying open data portals from the entire web, but limiting the search to the Common Crawl URL index archives that only contain the addresses of web pages. The current deep search method works directly in the Common Crawl WARC archives,

which contain the most detailed corpus of the entire textual web that is openly and regularly available to the research community.

The deep search method is a potential direction for achieving full automation. After determining the potential data portals through a manageable list of keywords, they were queried to identify the software platform being used from the four platforms considered in the study (CKAN, Socrata, OpenDataSoft, and ArcGIS Open Data), and how the datasets are quantified in them. These four platforms were selected based on the most mentioned platforms by the literature and the availability of documentation regarding the use of APIs.

We highlight that a total of 1,650 working data portals were found from the April 2019 crawl, exhibiting an increase of 21.3% in comparison to the last study. This method supersedes both previous studies in the series by improving the method to identify open data software platforms from the whole web as the single source of information.

The software platforms of open data portals found are distributed as follows: 49% to ArcGIS Open Data, 27% to CKAN, 15% to Socrata, and 9% to OpenDataSoft. The most identified data portal was ArcGIS; this platform had the lowest global average of published datasets. In contrast, CKAN has a global average of 5,967 datasets per data portal; it is also the only platform to demonstrate a significant participation in the range between 101 and 1,000 datasets per data portal. The remaining identified platforms were in the range of 11 and 100 datasets per data portal. These figures indicate an ongoing competitive scenario among the main players in the market of software platforms.

The automated method of obtaining geographic location was introduced in the last study and was reproduced in this experiment with a new set of data portals. The ccTLD method used to determine the location was successful in only 29% (480) of the cases, a number close to that obtained in the last study (30%). This reaffirms our recommendation for open data practitioners to always expose their ccTLD, apart from any other friendly versions of URL. This will increase the chances to automatically determine the precise location of a data portal, at least at the national level.

Achieving the level of automation intended by the deep search method involves some losses. For instance, the well-known European Open Data Portal (www.europeandataportal.eu/data) was not included in the current results. After investigating, we realized that the process of shortening URLs to the domain level discarded the URLs that included the path "/data" as a part of their CKAN endpoint. We adopted domain-level URLs to reduce the number of false positives supposed to be checked according to a finding from a previous study. However, the process omitted an important data portal. Meanwhile, we also realized that the European Open Data Portal did not respond to the CKAN API requests anymore, which indicated that they moved to another platform (other than CKAN). Consequently, we affirm our recommendation to include manual checking in the process of improving the results—depending on the desired degree of accuracy—in addition to the automated output.

The deep search approach addresses the problem of searching standardized open data software platforms stated in this research. The main limitation of the deep-search approach lies in the impossibility to find data portals that implement a software platform other than those considered in this study (CKAN, Socrata, OpenDataSoft, and ArcGIS Open Data). Thus, we designed a subsequent method that considered the data portals found to feed into a machine learning-based model and identify other data portals, irrespective of whether they implement standardized platforms or specific developed web-based software.

Thus, the second problem addressed by this research encompassed a machine learning classification model trained with data from the examples of typical data portals and ordinary web pages. The training data revealed good examples used to extract and encode features with information relevant to the analyzed problem; this information was used to determine whether a given input (in the form of a web page) can be labeled as a data portal or ordinary web page (non-data portal). This was performed by a probabilistic Naïve Bayes algorithm that calculates the likelihood of determining a data portal based on an independent word present in a web page. The combination of the 2,000 most common words defined as features set enabled the method to efficiently generalize a model to new examples and improve the accuracy of the classification task.

After inputting the training model into the machine learning algorithm, the entire process required no manual intervention, which is one of the major contributions of this research. Tests were conducted in two samples. *Sample 1* contained 3,502 manually collected URLs with a high probability of being classified as data portals, and *Sample 2* contained 38,350 randomly selected from ordinary web pages with no indication of their classification. In the experiment, approximately 0.7% or 279 data portals from a set of 38,350 web pages were found. This guided the best results indicating a proportion of approximately 6.7 (or 1,876 by 279) data portals classified in *Sample 1* for each instance in *Sample 2*. Because no manual checking was involved, the results were interpreted according to the nature of the sample, i.e., *Sample 1* should contain many data portals and *Sample 2* should contain very few labeled web pages.

As a part of the limitations of this work, it is worth emphasizing that not all data portals classified accordingly actually operated as data portals or even contained open data. The availability of standardized open data software platforms only opens a pathway for the identification of specific developed web-based software used for data publishing, and somehow serves as an indicative of open data taking part in the institution's agenda. The certainty of data openness in data portals can be improved, which will be investigated in the next stages of this research.

Moreover, the deep search approach demonstrated the need for accomplishing several computational resources. A workaround for those with limited resources is using a mixed new approach combining the efficacy of deep search and efficiency of the URL index method introduced in the last study. The subsequent machine learning classification experiment also demonstrated such a heavy process conducted on a global scale, containing billions of web pages; thus, classification should be performed in a reduced sample. These findings reinforce the adoption of a combined approach with the use of the URL index and classification process to identify data portals that rely on specific developed web-based software.

The availability and use of the intended worldwide repository of data portals will reduce efforts to find and complement the data that are somewhat necessary for conducting benchmark exercises. Thus, a single, up-to-date, and reliable source of open data portals surveyed automatically on a regular basis should be determined. Such a source will supersede the sparse initiatives and current challenges of having multiple and manual repositories that raise concerns about the redundancy, maintainability, and traceability of entries.

The accuracy of the classification process of the machine learning model can be improved via two methods. First, by expanding the training model by mapping additional open data software platforms other than the four considered in this study. Second, by potentially including other features in addition to words; for example, HTML elements such as headings, font formatting, and metatags. A better solution for determining the geographic locations includes the development of a method to identify the footprints of geocodes and location names directly from the HTML source code and datasets. Accordingly, the possibility of identifying locations at various levels, such as local (e.g., city/county), regional (state, province), national (country), and international (cross countries) will enrich the results and feature benchmark exercises. Finally, the validation of automatically collected data should be considered. As discussed, we prioritized maximum automation without any human intervention. Therefore, manual tests must be a part of the validation of the model, whose planning and execution will consume resources typical of a long-term research project.

We expect that the results section has showcased the designed method. The open data community can benefit from a handy list of data portals to be used (e.g., in benchmark exercises), where the discoverability of working data portals is a crucial component of the work. This community also has a starting model, whose contribution inaugurates a new approach for working with data portal availability.

7. References

Aggarwal, C. C. (2018). *Machine Learning for Text* (1st ed. 2018 edition). Springer.

Alves, L. G. A., Ribeiro, H. V., & Rodrigues, F. A. (2018). Crime prediction through urban metrics and statistical learning. *Physica A: Statistical Mechanics and Its Applications*, *505*, 435–443. https://doi.org/10.1016/j.physa.2018.03.084

Androutsopoulou, A., Karacapilidis, N., Loukis, E., & Charalabidis, Y. (2019). Transforming the communication between citizens and government through AI-guided chatbots. *Government Information Quarterly*, *36*(2), 358–367. https://doi.org/10.1016/j.giq.2018.10.001

Bird, S., & Klein, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media.

Bolychevsky, I. (2013, May 23). *U.S. government's data portal Data.gov relaunched on CKAN | ckan—The open source data portal software*. http://ckan.org/2013/05/23/data-gov-relaunch-on-ckan/

Brandusescu, A., Iglesias, C., & Robinson, K. (2016). *Open Data Barometer. Global Report. Fourth Edition*. The World Wide Web Foundation. http://opendatabarometer.org/doc/4thEdition/ODB-4thEdition-GlobalReport.pdf

Braunschweig, K., Eberius, J., Thiele, M., & Lehner, W. (2012). The State of Open Data—Limits of Current Open Data Platforms. *Proceedings of the 21st World Wide Web Conference 2012, Web Science Track at WWW'12, Lyon, France, April 16-20, 2012*. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.309.8903

Caplan, R., Davies, T., Wadud, A., Verhulst, S., Alonso, J., & Farhan, H. (2014). *Towards common methods for assessing open data: Workshop report & draft framework*. The

World Wide Web Foundation. http://opendataresearch.org/sites/default/files/posts/Common%20Assessment%20Workshop%20Report.pdf

Conradie, P., & Choenni, S. (2014). On the barriers for local government releasing open data. *Government Information Quarterly*, *31, Supplement 1*, S10–S17. https://doi.org/10.1016/j.giq.2014.01.003

Correa, A., & Fernandes, I. (2020, April). Open science-based framework to reveal open data publishing: An experience from using Common Crawl. *ELPUB 24rd Edition of the International Conference on Electronic Publishing*. https://hal.archives-ouvertes.fr/hal-02544245

Correa, A. S., & da Silva, F. S. C. (2019). Laying the foundations for benchmarking open data automatically: A method for surveying data portals from the whole web. *Proceedings of the 20th Annual International Conference on Digital Government Research: Governance in the Age of Artificial Intelligence*. https://doi.org/10.1145/3325112.3325257

Correa, A. S., Souza, R. M. de, & Silva, F. S. C. da. (2019). Towards an automated method to assess data portals in the deep web. *Government Information Quarterly*, *36*(3), 412–426. https://doi.org/10.1016/j.giq.2019.03.004

Correa, A. S., Zander, P.-O., & da Silva, F. S. C. (2018). Investigating Open Data Portals Automatically: A Methodology and Some Illustrations. *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age*, 82:1–82:10. https://doi.org/10.1145/3209281.3209292

Corrêa, A. Sh., Paula, E. C. de, Corrêa, P. L. P., & Silva, F. S. C. da. (2017). Transparency and open government data: A wide national assessment of data openness in Brazilian local

governments. *Transforming Government: People, Process and Policy*, *11*(1). https://doi.org/10.1108/TG-12-2015-0052

European Union. (2017). *Recomendations for Open Data Portals: From setup to sustainability* (p. 76). https://www.europeandataportal.eu/sites/default/files/edp_s3wp4_sustainability_recommendations.pdf

Green, B. Z., Cunningham, G., Ekblaw, A., Kominers, P. M., Linzer, A., & Crawford, S. P. (2017). Open Data Privacy. *Berkman Klein Center for Internet & Society Research Publication*. https://dash.harvard.edu/handle/1/30340010

Ku, C.-H., & Leroy, G. (2014). A decision support system: Automated crime report analysis and classification for e-government. *Government Information Quarterly*, *31*(4), 534–544. https://doi.org/10.1016/j.giq.2014.08.003

Kubler, S., Robert, J., Le Traon, Y., Umbrich, J., & Neumaier, S. (2016). Open Data Portal Quality Comparison Using AHP. *Proceedings of the 17th International Digital Government Research Conference on Digital Government Research*, 397–407. https://doi.org/10.1145/2912160.2912167

Kubler, S., Robert, J., Neumaier, S., Umbrich, J., & Le Traon, Y. (2018). Comparison of metadata quality in open data portals using the Analytic Hierarchy Process. *Government Information Quarterly*, *35*(1), 13–29. https://doi.org/10.1016/j.giq.2017.11.003

Leetaru, K. (2017, September 28). *Common Crawl And Unlocking Web Archives For Research*. Forbes. https://www.forbes.com/sites/kalevleetaru/2017/09/28/common-crawl-and-unlocking-web-archives-for-research/

Lisowska, B. (2016). *Metadata for the open data portals*. Technical Report. Joined-up Data Standards Project. http://juds.joinedupdata.org/wp-content/uploads/2016/12/JUDS-DP6-Metadata-for-the-open-data-portals.pdf

McCallum, A., Nigam, K., & others. (1998). A comparison of event models for naive bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization*, *752*(1), 41–48.

Mercier, R. (2015, November 2). *We listed 2600+ Open Data portals around the world! See how!* OpenDataSoft. https://www.opendatasoft.com/2015/11/02/how-we-put-together-a-list-of-1600-open-data-portals-around-the-world-to-help-open-data-community/

Milic, P., Veljkovic, N., & Stoimenov, L. (2018). Comparative analysis of metadata models on e-government open data platforms. *IEEE Transactions on Emerging Topics in Computing*, 1–1. https://doi.org/10.1109/TETC.2018.2815591

Nair, V. G. (2014). *Getting Started with Beautiful Soup*. Packt Publishing.

Neumaier, S., Jürgen, U., & Axel, P. (2017, April 3). *Lifting data portals to the web of data*. WWW2017 Workshop on Linked Data on the Web (LDOW2017), Perth, Austrialia. http://events.linkeddata.org/ldow2017/papers/LDOW_2017_paper_9.pdf

Neumaier, S., Umbrich, J., & Polleres, A. (2016). Automated Quality Assessment of Metadata Across Open Data Portals. *J. Data and Information Quality*, *8*(1), 2:1–2:29. https://doi.org/10.1145/2964909

Open Data Charter. (2015). *International Open Data Charter Principles*. International Open Data Charter. https://opendatacharter.net/principles/

Open Government Working Group. (2007). *The 8 Principles of Open Government Data (OpenGovData.org)*. http://opengovdata.org/

Osagie, E., Mohammad, W., Stasiewicz, A., Hassan, I. A., Porwol, L., & Ojo, A. (2015). *State-of-the-art Report and Evaluation of Existing Open Data Platforms* (645860 H2020-INSO-2014). https://project.routetopa.eu/deliverable-2-1-state-of-the-art-report-and-evaluation-of-existing-open-data-platforms-now-available/

Russell, S., & Norvig, P. (2009). *Artificial Intelligence: A Modern Approach* (3 edition). Pearson.

Sadiq, S., & Indulska, M. (2017). Open data: Quality over quantity. *International Journal of Information Management*, *37*(3), 150–154. https://doi.org/10.1016/j.ijinfomgt.2017.01.003

Susha, I., Zuiderwijk, A., Janssen, M., & Grönlund, Å. (2014). Benchmarks for Evaluating the Progress of Open Data Adoption: Usage, Limitations, and Lessons Learned. *Social Science Computer Review*. https://doi.org/10.1177/0894439314560852

Tauberer, J. (2014). *Open Government Data: The Book - Second Edition*. https://opengovdata.io/

Thorsby, J., Stowers, G. N. L., Wolslegel, K., & Tumbuan, E. (2017). Understanding the content and features of open data portals in American cities. *Government Information Quarterly*, *34*(1), 53–61. https://doi.org/10.1016/j.giq.2016.07.001

Ting, S., Ip, W., & Tsang, A. H. (2011). Is Naive Bayes a good classifier for document classification. *International Journal of Software Engineering and Its Applications*, *5*(3), 37–46.

Ulrich, A., Tom, H., & Jamie, F. (2015). *Benchmarking open data automatically* (ADI-TR-2015-000). Open Data Institute. https://theodi.org/guides/benchmarking-data-automatically

United Nations Publications. (2016). *United Nations E-Government Survey 2016: E-Government in Support of Sustainable Development*. United Nations. http://workspace.unpan.org/sites/Internet/Documents/UNPAN97453.pdf

van der Waal, S., Węcel, K., Ermilov, I., Janev, V., Milošević, U., & Wainwright, M. (2014). Lifting Open Data Portals to the Data Web. In S. Auer, V. Bryl, & S. Tramp (Eds.), *Linked Open Data—Creating Knowledge Out of Interlinked Data: Results of the LOD2 Project* (pp. 175–195). Springer International Publishing. https://doi.org/10.1007/978-3-319-09846-3_9

Veljković, N., Bogdanović-Dinić, S., & Stoimenov, L. (2014). Benchmarking open government: An open data perspective. *Government Information Quarterly*, *31*(2), 278–290. https://doi.org/10.1016/j.giq.2013.10.011

Vetrò, A., Canova, L., Torchiano, M., Minotas, C. O., Iemma, R., & Morando, F. (2016). Open data quality measurement framework: Definition and application to Open Government Data. *Government Information Quarterly*, *33*(2), 325–337. https://doi.org/10.1016/j.giq.2016.02.001

Weslei Gomes de Sousa, Elis Regina Pereira de Melo, Paulo Henrique De Souza Bermejo, Rafael Araújo Sousa Farias, & Adalmir Oliveira Gomes. (2019). How and where is artificial intelligence in the public sector going? A literature review and research agenda. *Government Information Quarterly*. https://doi.org/10.1016/j.giq.2019.07.004

Yang, H.-C., Lin, C. S., & Yu, P.-H. (2015). Toward Automatic Assessment of the Categorization Structure of Open Data Portals. In L. Wang, S. Uesugi, I.-H. Ting, K. Okuhara, & K. Wang (Eds.), *Multidisciplinary Social Networks Research* (pp. 372–380). Springer. https://doi.org/10.1007/978-3-662-48319-0_30