Design and Evaluation of a Data Collector and Analyzer to Monitor the COVID-19 and Other Epidemic Outbreaks

Lucas C. de Almeida, Francisco L. de Caldas Filho, Natália A. Marques, Daniel S. do Prado, Fábio L. L. de Mendonça, and Rafael T. de Sousa Jr.

Faculty of Technology, Electrical Engineering Department, University of Brasília, Brasília, DF 70910-900, Brazil
lucas.almeida@redes.unb.br

Abstract. Pandemic situations require analysis, rapid decision making by managers and constant monitoring of the effectiveness of collective health-related approaches. These works can be more efficient with the help of clearer and more representative views of the data, as well as with the application of other measures and projections of epidemiological nature to the information. However, performing such aggregations of data can become a major challenge in contexts with little or no integration between databases, or even when there is no technological core mature enough to feed and integrate technological advances in the workflow of health professionals. This paper aims to present the results of the meeting of project approaches such as the OSEMN framework, a software architecture based on Microservices and Data Science technologies, all tools aligned to make the environment of descriptive and predictive analysis of epidemic data (still dominated by manual processes) evolve towards a context of automation, reliability and application of machine learning, aiming at the organization and addition of value to the results of the data structuring. The project’s validation objects were the documents of the situation of the Covid-19 disease pandemic in the region of the city of Brasília, Federal District, Capital of Brazil.

Keywords: Data scraping · Data visualization · Epidemic data analysis · Data automation

1 Introduction

Several areas have gone through revolutions with the application of Data Science methodologies. Among these areas, one has gained space and accelerated the concern with the availability and visualization of data: public health. Recently, a disease has gained the status of a pandemic and, in a short period, has reached all continents and countries around the world, including Brazil, which will be...
the test and implementation environment for the system of this work. The disease called Covid-19 [16], because of its infectiousness, requires greater care and short response time in decision making, which can only be made possible with the availability of aggregated data, historical series, projections and other epidemiological measures [18].

Opportunely, Data Science has very widespread and well-known work processes, and one of them, considered almost universal, is the OSEMN framework. good example of implementing this methodology can be seen in [14]). A work in this area, according to the aforementioned framework, must follow five steps, in order, to achieve the kind of results desired by a data scientist: obtain, scrub, explore, model and interpret. In other words and briefly: acquire the data of interest for the study; clean and format them so that machines can understand them; if necessary, find patterns and other measures and build models that exploit those patterns, such as projections; and organize visualizations that allow a complete understanding of the results. Therefore, it is clear that, for health professionals to enjoy the benefits of Data Science applied to collective health, it would be extremely important that they have ways to comply with each of the steps mentioned, which can be challenging in various contexts.

In the context of Emerging Countries, such as Brazil, it is common to verify regions without access to health service management technologies, or, even when access exists, to have jobs marked by the low use and integration of these solutions (as a software to manage patients data that is not fully integrated with central systems and that the professionals don’t know how to use properly). In these places, several activities are still done manually, such as control of medical records, recipes and, in the case of epidemics, notifications of active cases and monitoring of the general situation [19]. Therefore, the conclusion is direct that works that demand greater availability and sophisticated data analysis become quite complicated and, in some cases, even unfeasible in these scenarios.

To solve the problem, first of all, it is necessary to understand that even if the data is spread over the Internet, or in documents, it is possible to accumulate this information in a structured way and, later, process it and add value to the analysis. The first step in the case of this work, is to understand the concept of web scraping (a good study on the topic can be seen in [15]): technique in which a program interacts with web pages and seeks information similarly to the way a human user would do. In a complementary way, the so-called “regular expressions” provide research methodologies and filtering capabilities that are extremely powerful, representing almost a standard tool for data scraping over web pages. This is due their flexibility and ability to search for string patterns with high precision and speed. Although there is no standard for the operation of the mentioned “regular expressions”, there are efforts to centralize references, and one of them can be seen in [9]. Combining the two concepts, the creation of programs that interact with resources available online and find scattered and unformatted data becomes fast and uncomplicated. This approach is especially interesting when one considers the fact that most information on diseases and epidemics is available in the form of links for downloading files from websites of government agencies and departments, as in the case of Brazil.
This work describes the design and implementation of a system that aims to automate the various phases of a Data Science work in order to aggregate, structure, make available and model public data on the epidemic of Covid-19 disease provided in PDF file format. The application context is related to public health data from the Federal District, capital of Brazil. As a result, it was demonstrated that with a simple project and a lean team, it was possible to convert a scenario of manual readings and analyzes to high availability access to historical series and epidemiological analyzes, in addition to speed in the implementation of tests of new calculations and studies.

It is important to highlight that publishing data in PDF files is as a big challenge to make it accessible by machines. The PDF extension stands for Portable Document Format (PDF) and it defines an ISO standardized document that can contain text, fonts, vector graphics, and figures.

This paper consists of six sections, including this introduction. Section 2 deals with related work, especially on the necessary technical concepts. Section 3 specifies the proposed solution, uniting methodologies widely used in the industry and that integrate seamlessly into the context of users. Section 4 serves to discuss and demonstrate the results obtained in implementing the project with real data. Finally, findings and future work can be found in Sect. 5 and, in the last section, the acknowledgments of the authors.

2 Related Works

The recent arrival of the Covid-19 disease motivated the development of several software in order to inform about the impact of the disease. This work aims to disseminate contagion and death data in a structured and aggregated way, allowing for in-depth analysis, and for that, Big Data, data analysis and Web scraping concepts were used. However, the use of these technologies and methods, together, to integrate and facilitate the work of health professionals is still new, with few proposals with the same purpose. Some works could be related, as described as it follows.

[20] shows a definition of Web scraping, which consists of a set of techniques for obtaining data from an Internet source just like a person would personally do. In the article cited, this tool was implemented with the purpose of collecting images of the company or service to be advertised in order to do a collaborative filtering that aims to suggest suitable ads for a generic web page. The present work uses similar techniques to obtain and filter reports in PDF files from the website of the Secretary of Public Health of the Federal District. The collection of this data is important for the realization of multidimensional modeling of the data extracted about Covid-19 disease.

In the context of Big Data, the article [13] defines several concepts about open data. In addition, their proposed work supports the analysis of a large volume of open government data with the need for rapid processing. However, the proposed solution is barely feasible for technologically immature environments, and its application for cases with different layers of data without proper structuring,
for example, documents in PDF format listed as download links in websites, is complex even for advanced users. In contrast, in our work the results and the workflow show that the objective is to integrate completely into the users’ routine without requiring new knowledge and concepts.

Due to the large number of cases of malaria contagion in the mountains of East Africa, the work [22] developed an application that processes satellite data for modeling and forecasting epidemics of this disease, which is currently a major public health problem in the region. The cited article uses advanced data science concepts to add value to the predictive analysis of the disease. However, by using past data that is not accessible in real time, it is limited to seeking historical patterns. Similarly, our proposal also deals with an application in which one of the objectives is to make predictions and modeling of the data collected about a disease (Covid-19). However, Data Science is used in the present work to constantly and quickly bring aggregability and integration to public data, even allowing one to trace the start of possible patterns, going beyond the need of a complete organized historic (and often outdated) set of data.

In terms of data analysis, work in [17] makes a significant contribution by addressing relevant methods in the process of linear and non-linear analysis of data by applying efficient techniques of analysis and process diagnosis. These methods can be applied in the present work to perform the test of the data collected from the website of the Public Health Department of the Federal District. However, in a different manner, the modelling is automatically made, gaining time and reducing complexity.

When it comes to the use of Big Data applied to public health, the work [21] brings an explanatory analysis of how Taiwan used a national health insurance database and integrated it with immigration and customs information to analyze possible patients with probability of contagion by Covid-19. It also used new technologies, such as reading online travel history reports and health symptoms to classify travelers’ infection risks based on flight origin and travel history in the last 14 days. The use and implementation of these technologies facilitated the country to map the cases of Covid-19. Based on this, it is observed that the work to be proposed in this article represents a major contribution in the area of public health in the sense of mapping the cases of Covid-19 in each region. The major difference is that the present work was implemented aiming to be applied in smaller contexts, generally trying to bring sophistication to less empowered regions.

3 Design of an Epidemic Data Science Project

This work is included in the field of Data Science, which follows, in a generalized way, a sequence of steps very well defined and based on the framework OSEMN, as stated in Sect. 1. Directing the application of this framework to the context of information about diseases implies the choice of certain technologies and preferable procedures. This is due to the fact that the source of the information (health professionals) does not always have knowledge in relation to technological procedures for collecting, aggregating and structuring data storage, therefore, the
operation of the project should favor methods that are easy to adhere for these professionals and for existing workflows. In addition, from a technical point of view, the system should use a separate microservices approach, aiming at the low coupling between the different processing steps and the possibility of creating new modules that perform the same steps, but with a different purpose, as it will be addressed.

After describing the application context, it becomes possible to go through the specification of the implementation of the proposed project. Therefore, based on the general work sequence of the Data Science field, it was possible to derive a basic flowchart expected for the project, as shown in Fig. 1. This flowchart describes the basic steps needed for any Data Science project applied to a context of public health. They can be used as a guide to designing other systems and softwares with similar purposes.

Next, the choices for the implementation to be demonstrated will be described for each stage of the proposed flow. Also, from Fig. 1, a diagram of the modules and the general functioning of the project as a validation result of the application of the framework was derived according to Fig. 2.

**Obtain:** A Python script [8] is executed periodically using the cron scheduler service native to the Linux operating system distributions [5]. In each iteration, the program:

1. Accesses the online repository of the Federal District Health Department;
2. Obtains, by processing the code of the web page, the list of links to the reports in PDF format;
3. Searches and separates, in an ordered list format, the numbers of the available documents using regular expressions to filter variations in the code of the page layout and buttons, as explained in Sect. 1;
4. Checks in a local folder which files (present in the generated list) have not yet been saved and downloads the reports without correspondence;
5. Asks the Operating System to execute the next module.

**Scrub:** It is also a script developed in Python language. In each iteration, the program:

1. Compares the local folder with files in PDF format with another folder containing files in comma-separated values (CSV) format that is data arranged as tables. The goal is to verify that the two folders are synchronized, so that for each PDF file there is a corresponding CSV one. When reports are found without corresponding table files, it means that the report has not yet been processed, in which case the program proceeds with data extraction;
2. For extraction, using the Python language library Tika-Python [12] (which communicates with the public API Apache Tika [1]), it is sought, using regular expressions that mark the beginning and end of the table of interest, the values separated by administrative regions

1 Administrative regions are government divisions of Brazil’s Federal District. For the sake of simplicity, an administrative region can be defined as a district.
using regular expressions, each value of each row and column expected for the table structure is inserted into the new CSV file to be created. It is important to note that, as [2], extracting tables from files with the extension PDF can be quite computationally expensive. For this reason, the method used in the aforementioned library does not return tables, only the text available in the document, but without great formatting guarantees. Therefore, it is quite common to check for misalignments, multiple spaces, characters that have been added or incorrectly mapped, among others. However, with the use of regular expressions, it is possible to search the table of interest using the beginning and end markers. It is also possible, by modeling the return with regular expressions for each line, to search the data and to filter the errors of the extraction process quite efficiently;

Fig. 1. Flowchart expected from the proposed system.
3. At the end of the extraction, the program saves the new CSV files in a backup folder (for future references and/or uses). It also deletes all files available in another workbook, used by the next module, and adds the new files extracted there as an output;

4. At the end, it requests the execution of the next module to the operating system.

**Explore and Model:** For the case of the project, which required agility in the development and approval of features, two versions of this module were made. As explained earlier, the division of tasks was done aiming at the maximum reduction of the coupling, so that there is also no problem in having other versions of the same module creating new branches in the flow of processing of the collected data.

The difference between the versions is related to functionality: one is used in production, for large volumes of requisitions, while the other serves the purpose of accelerating tests and developments. The robust version is based on a Bash language script [3] that automates the inclusion of data in an instance of the MongoDB database [7]. The other version is based on a Python script that includes data in a SQLite database [11] (simple relational database in file format). In this version, data is served using the Flask framework [6], which operates based on Python and has a very low implementation time curve.
A valuable aspect demonstrated at this stage is the flexibility of work once the information is properly organized and categorized after extraction and aggregation. The same workflow can benefit from different types of data storage and access methods (table files, relational databases in file format and high performance non-relational databases served with high availability).

Referring to the term “model”, for further study of the data and tests, a method was implemented in the API supported by the Flask framework which receives an administrative region name and a number of days, and based on the projection model described in [4] and using the library sklearn [10], returns the simple projection of the number of cases from the last available days in the database to the future days, according to the amount passed. This projection is based on the prediction of natural logarithms of the number of cases accumulated per day with an unsupervised linear regression algorithm.

Other models could also be added to the test API to, for example, study the most adherent for each region and, subsequently, implement such calculation in the high performance and availability application interface.

Still on the topic of data modeling, it is important to note that the value of application interfaces is also related to the easy interoperability between different systems, the possible sophistication in data filtering and the speed. It is possible, for example, to request very quickly average values, the time series of a single region, of a set of them, the data of all regions on a given date, the maximum value already collected, among others.

Interpret: This module, as it is only a consumer of processed and structured data, does not work in tandem with the others. In fact, it doesn’t even have to be a single program with a single output. Thus, for the commented scenario, the following structured data consuming services were created, which add value to the analysis:

1. A web page that renders time series graphs for the number of reported cases and deaths by region and by region aggregate (for example, all north regions), as well as georeferenced data (based on the name of the administrative region) and with color grading according to the number of cases in relation to the greatest value of the research;
2. Also consuming the data, another sophisticated application was created which, using georeferenced formats and the complete compilation and analysis of the last available historical data from the API, creates short animated videos showing the disease progression in each region for each day of the pandemic.

Several other consumer services could have been created for this stage of the process, such as mobile applications, the use of third-party data analysis software and even the creation of customized report generation software. Again, given the independence between each module and the successful cleaning and organization of data, impractical or extremely repetitive and time-consuming activities for human operators can become fast and even automated.
4 Evaluation of Tests and Results of the Proposed System

In this section, the results obtained with the application of the system will be presented in the context of the data from the Health Department of the Federal District of Brazil made available until August 15, 2020, with 165 reports in the repository.

First, on the efficiency of extraction, it was found that 100% of the reports were downloaded and renamed correctly in PDF format. Of 165 reports correctly obtained from March 23 to August 15, 2020, 138 were processed and resulted in successful data extraction. Also, from reports numbered 1 to 23, the table of interest did not exist yet. Therefore, only 142 reports were real objects of the extraction, and of these, 138 resulted in success. This corresponds to an error rate of about 2.81% and a success rate of 97.19%. This margin of error is due to the various formatting and tabulation problems both in the source code of the repository’s web page and in the documents themselves. As stated earlier, regular expressions are used to filter out these flaws, however not all cases can be predicted.

With the data already stored in the database, it was possible to make projections, as explained previously. The most common ways of presenting this information are line and bar graphs, using both ways to create the first visualizations, as shown in Fig. 3.

![Fig. 3. Line and bar time series graphs with moving averages.](image.png)
In Fig. 3 it is possible to analyze the evolution of the cases in three different perspectives: accumulated data (figures on the left side); cases per day (bar graphs on the right side); and moving average of the cases (line graphs on the right side). For Covid-19 disease, the period of seven (7) days was used to calculate the averages as shown in the figure. It is also possible to select other periods for the average, such as 15 or 21 days.

Also, in health outbreaks, it is of great importance to geographically identify the spread of the disease. Understanding the evolution in each region, overlapping this data with other factors, such as economic, social, and even population density are of great relevance to understand where possible governmental decisions are most needed. This platform is able to project data by overlaying information on a map based on the names of administrative regions. Figure 4 exemplifies in a georeferenced way the data overlaid in the Federal District region. With this it is possible to analyze, by color, the number of infections and deaths in each region.

Still on the same way of visualization, the platform is also capable of generating short videos showing the spread of the disease over time, generating a historical series of the evolution of the number of georeferenced active cases, according to sample frames from one of the videos in Fig. 5.

Finally, considering the possibility of comparing different regions, it is also possible to create visualization scenarios in which the user chooses the regions of interest. In Fig. 6, three regions are selected: Ceilândia, Taguatinga and Gama. It is possible to observe, for example, that Ceilândia currently holds around 50% in the number of infected and deaths, considering only these 3 options as a total data set.
There are still several other filters and methods available, especially through APIs, allowing different integrations in third party systems. The malleability of the data resulting from a complete iteration of the system adds new layers of study and decision possible for health professionals without requiring technical knowledge in systems, all because with these approaches, they can continue to use the same human-friendly documents, and still data can be aggregated and modeled with the implementation of different software services in each step of the presented framework.

5 Conclusion and Future Work

In this paper, a system designed to approximate the information and workflows already existing in the field of public health with the most current methodologies of Data Science was presented. It was possible to show that even in teams with precarious conditions for the application and support of data analysis and aggregation technologies, it is possible to produce automated solutions
that are integrated into the routine of these teams, being only required a reliable and efficient framework (as the slightly modified OSEMN presented) and associated methodologies (such as microservices architecture principles) for the project design. The system was tested with real data from the government entity responsible for public health in the capital of Brazil and the results of the application reached levels of visualization and analytical modeling as advanced as those performed in technologically mature environments, allowing, ultimately, georeferenced visualization of the spread of the Covid-19 disease epidemic (in the form of a video) from simple, manually edited PDF reports. Still, it should be noted that even in an urban center, situations of lack of integration, problems in the availability and handling of data, among others, are verified, and for this reason, the construction of applications with the objective of automating and abstracting technical processes adds a lot of value to the work scenario and management of health systems.

However, the system, although complex, is still only an inspiration for other initiatives of the same nature. For new advances in the implementation of this proposal, it would be important to create an interface in which the end user would be able, through simple commands that do not require long training, to choose the form and location of the data to be extracted in a model report. In addition, adding the use of OCR (Optical Character Recognition) technology could allow, for example, to register in a structured way past medical records of patients and even medical prescriptions, after the simple scanning of these documents. Still, with the proper training of an artificial intelligence model, it would be possible to add the functionality of checking the level of trust of signatures and stamps, extending the use of the system to pharmacies, audits and transparency portals, for example.

Acknowledgment. The authors would like to thank the support of the Brazilian research, development and innovation agencies CNPq (Projects INCT SegCiber 465741/2014-2, PQ-2 312180/2019-5 and LargEWiN BRICS2017-591), CAPES (Projects FORTE 23038.007604/2014-69 and PROBRAL 88887.144009/2017-00) and FAPDF (UOiT Projects 0193.001366/2016 and SSDCC 0193. 001365/2016), as well as the support of the LATITUDE/UnB Laboratory (SDN Project 23106. 09941/2016-43), cooperation with the Ministry of the Economy (TEDs DIPLA 005/2016 and ENAP 083/2016), the Office of Institutional Security of the Presidency of the Republic (TED 002/2017), the Attorney General’s Office (TED 697,935/2019) and the Administrative Council for Economic Defense (TED 08700.000047/2019-14).

References

1. Apache Tika - a content analysis toolkit. https://tika.apache.org/ 
2. Apache Tika - Class PDFParser. http://tika.apache.org/1.24.1/api/org/apache/tika/parser/pdf/PDFParser
3. Bash Reference Manual. https://www.gnu.org/savannah-checkouts/gnu/bash/manual/bash.html
4. Coronavirus 10-day forecast. https://covid19forecast.science.unimelb.edu.au/
5. Cron(8)—linux manual page. https://man7.org/linux/man-pages/man8/cron.8.html
6. Flask - Web Development. https://flask.palletsprojects.com/en/1.1.x/
7. MongoDB Database. https://www.mongodb.com/
8. Python Programming Language. https://www.python.org/
9. Regular Expressions Reference. https://www.regexlib.info/ref/reference.html
10. Scikit-Learn. https://scikit-learn.org/stable/
11. Sqlite database. https://www.sqlite.org/index.html
12. Tika-Python. https://github.com/chris-mattmann/tika-python
13. de Carvalho Victorino, M., Shiessl, M., Oliveira, E.C., Ishikawa, E., de Holanda, M.T., de Lima Hokama, M.: Uma proposta de ecossistema de big data para a análise de dados abertos governamentais concetados. Informação & sociedade 27(1) (2017)
14. Dineva, K., Atanasova, T.: OSEMN process for working over data acquired by iot devices mounted in beehives. Current Trends Nat. Sci. 7(13), 47–53 (2018)
15. Diouf, R., Sarr, E.N., Sall, O., Birregah, B., Bousso, M., Mbaye, S.N.: Web scraping: state-of-the-art and areas of application. In: 2019 IEEE International Conference on Big Data (Big Data), pp. 6040–6042 (2019)
16. Lima, C.M.A.D.O.: Information about the new coronavirus disease (COVID-19). Radiologia Brasileira 53, V–VI, April 2020. https://doi.org/10.1590/0100-3984.2020.53.2e1
17. Mogk, M.: Automatic data processing in analyses of epidemics. In: Epidemics of Plant Diseases: Mathematical Analysis and Modeling, pp. 55–77. Springer (1974). https://doi.org/10.1007/978-3-642-96220-2_3
18. Santos, A.D.F.D., Fonseca Sobrinho, D., Araujo, L.L., Procópio, C.d.S.D., Lopes, E.A.S., Lima, A.M.d.L.D.d., Reis, C.M.R.D., Abreu, D.M.X.D., Jorge, A.O., Matta-Machado, A.T.: Incorporação de Tecnologias de Informação e Comunicação e qualidade na atenção básica em saúde no Brasil 33 (2017). https://doi.org/10.1590/0102-311x00172815
19. Santos, S.R., Paula, A.F.A., Lima, J.P.: O enfermeiro e sua percepção sobre o sistema manual de registro no prontuário. Revista Latino-Americana de Enfermagem 11, 80–87 (2003). https://doi.org/10.1590/S0104-11692003000100012
20. Vargiu, E., Urru, M.: Exploiting web scraping in a collaborative filtering-based approach to web advertising. Artif. Intell. Research 2(1), 44–54 (2013)
21. Wang, C.J., Ng, C.Y., Brook, R.H.: Response to COVID-19 in Taiwan: big data analytics, new technology, and proactive testing. Jama 323(14), 1341–1342 (2020)
22. Wimberly, M.C., Chuang, T.W., Henebry, G.M., Liu, Y., Midekisa, A., Semunguise, P., Senay, G.: A computer system for forecasting malaria epidemic risk using remotely-sensed environmental data (2012)