

OPEN SOURCE INTELLIGENCE (OSINT) GATHERING AND ANALYSIS TOOLKIT

Project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

In

Computer Science and Engineering/Information Technology

By

SAKSHAM TANDON 171071

Under the supervision

of

MR. PRAVEEN MODI

to



Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

CERTIFICATE

CANDIDATE'S DECLARATION

I hereby declare that the work presented in this report entitled “**OPEN SOURCE INTELLIGENCE (OSINT) GATHERING AND ANALYSIS TOOLKIT**” in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering/Information Technology submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from January 2021 to May 2021 under the supervision of (Parveen Modi) (Assistant Professor and CSE&IT).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Saksham Tandon 171071

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Mr. Praveen Modi

Assistant Professor (Grade-I)

Department of computer science & engineering and information technology,

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh.

05-12-2020

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our teacher and mentor Mr. Praveen Modi who gave us the golden opportunity to do this project on the topic open source intelligence (OSINT) gathering and analysis toolkit, who also helped us in doing a lot of Research and we came to know about so many new things. We are thankful to him. Secondly, we would also like to thank Lab assistants who helped us a lot in finalizing this project within the limited time frame.

I am grateful to, Head of Dept. of Computer Science & Engineering and Information Technology for his excellent support during our work. We would also like to thank all the professors & other supporting members of the department of Computer Science & Engineering and Information Technology for their generous help in various ways for the completion of this work. I am also obliged to all my faculty members for their valuable support in their respective fields which helped me in reaching at this stage of our project.

Saksham Tandon (171071)

TABLE OF CONTENT

Abstract	1
Chapter-1 Introduction	2
1.1 Introduction	2
1.2 Problem Statement	2
1.3 Objectives	3
1.4 Methodology	3
1.5 Organization	4
Chapter-2 Literature Survey	5
2.1 Intelligence in the internet age: The emergence and evolution of Open Source Intelligence (OSINT)	5
2.2 A Framework for the Systematic Collection of Open Source Intelligence	6
2.3 Digital forensic intelligence: Data subsets and Open Source Intelligence (DFINTOSINT): A timely and cohesive mix	7
2.4 Open-source intelligence and privacy by design	7
2.5 OSINT as a part of cyber defence system	8
Chapter-3 System Development	9
3.1 General facts about web scraping	9
3.1.1 Purpose of web scraping	9
3.2.2 Analysis and research	9
3.3.3 Social mining and sentiment analysis	10
3.2 Legal aspects of data extraction	10
3.2.1 Law areas for web scraper defence action	10
3.2.2 Website terms and conditions	11
3.2.3 Copyright, intellectual property rights	11
3.2.4 Database rights	11
3.2.5 Trademarks	11
3.2.6 Data protection	12
3.2.7 Criminal damage	12
3.3 General legal aspects	12
3.4 Methods of web scraping	13
3.4.1 Manual scraping	13
3.4.2 Html parsing	13
3.4.3 Dom parsing	13
3.4.4 Xpath	14
3.4.5 API	14
3.5 Available software tools	15
3.5.1 Cloud software	16
3.5.2 Desktop software	17
3.6 Algorithm design	17
3.6 System design	18

Chapter-4 performance analysis	19
4.1 Requirement analysis	19
4.2 Technical requirements	19
4.2.1 Jupyter notebook	19
4.2.2 Numpy	19
4.2.3 Matplotlib	20
4.2.4 Seaborn	20
4.2.5 Reportlab	20
4.2.6 Pyqt	20
4.3 Requirements for user interface	20
4.3.1 Application frontend editor	21
Chapter-5 Conclusion & Future Directions	24
5.1 Conclusion	24
5.2 Future Scope	24
References	25
Appendices A	26

LIST OF FIGURES

S.No	Name of Figures	Page No.
1.	FIGURE 1: PROJECT WORKFLOW	3
2.	FIGURE 2: HTML PARSING	14
3.	FIGURE 3: DOM PARSING	14
4.	FIGURE 4: XPATH	15
5.	FIGURE 5: ALGORITHM	17
6.	FIGURE 6: SYSTEM DESIGN	18
7.	FIGURE 7: APPLICATION PRE-LOADER	21
8.	FIGURE 8: APPLICATION MAIN SCREEN	21
9.	FIGURE 9: OBJECT INSPECTOR	22
10.	FIGURE 10: PROPERTY EDITOR	22
11.	FIGURE 11: STYLE SHEET	23
12.	FIGURE 12: SETTINGS PAGE	23

LIST OF ABBREVIATIONS

S.No	Name of Abbreviation	Full Form
1.	OSINT	Open source intelligence
2.	ALGOL	Algorithmic Language
3.	API	Application Program Interface
4.	ASCII	American Standard Code for Information Interchange
5.	AUI	Attachment Unit Interface
6.	BIOS	Basic Input Output System
7.	CMD	Command
8.	CPI	Clock / Cycle Per Instruction
9.	DAC	Data Acquisition and Control
10.	DBMS	Data Base Management System

ABSTRACT

There are a few pages of findings that are seen when we search the internet. We just look at the first page and research. But if you've ever wondered what the rest of the hundred pages of the result is? "Information!" open source intelligence (OSINT) gathering and analysis toolkit gets this information using a range of modules.

Many people think using their favorite Internet search engine is enough to find the data they need and do not realize that most of the Internet is not indexed by search engines.

With this toolkit we will implement effective methods of finding these data. By integration of automation OSINT collection and deep data analysis scripts this toolkit will provide processed report.

To create automated OSINT data gathering and analysis scripts and integrating all into one toolkit this project is divided into four phases Research and analysis, Structuring and optimizing the process of data collection, Designing and deploying of scripts, Integration of scripts into one toolkit.

This project uses technique like search engine dorking queries and other OSINT information gathering techniques to acquire data from the internet.

Modules including social media post scrapers and public personal information scrapers.

CHAPTER 1

INTRODUCTION

1. Introduction

There are a few pages of findings that are seen when we search the internet. We just look at the first page and research. But if you've ever wondered what the rest of the hundred pages of the result is? "Information!" open source intelligence (OSINT) gathering and analysis toolkit gets this information using a range of modules.

Many people think using their favourite Internet search engine is enough to find the data they need and do not realize that most of the Internet is not indexed by search engines.

OSINT stands for the wisdom of open source. The Internet is an ocean of data that is both an asset and a drawback. Pros are that the Cyberspace is free and accessible to all, unless it is limited by law or organization. The Internet does have all the information that anyone can easily access. The disadvantages are that resource is known that can be misused by anyone with a malicious intent. Collection and correlation of information using such methods is referred to as open source knowledge.

This project uses technique like search engine dorking queries and other OSINT information gathering techniques to acquire data from the internet. Modules including social media post scrapers and public personal information scrapers, structure all the information into a dataset which is available to be saved as a CSV file or PDF report.

2. Problem Statement

It's one thing to know that the knowledge is open. Collection of information is second and the third is to make an interpretation or intelligence out of it. The details can also be obtained manually, but it will take the time that can be used in later stages instead. This tool will help us collect data from the sites in a matter of minutes and thus ease the collection process.

For an analyst, having such a vast amount of information available is both a blessing and a curse. On the one hand, you have access to almost anything you might need—but on the other hand, you actually have to be able to find it in a never-ending stream of data.

3. Objectives

The aim of this project is to create automated OSINT data gathering and analysis scripts and integrating all into one toolkit which can create datasets in form of CSV files and PDF reports. This project uses technique like search engine dorking queries and other OSINT information gathering techniques to acquire data from the internet. Modules including social media post scrapers and public personal information scrapers, structure all the information into a dataset which is available to be saved as a CSV file or PDF report.

Although there are many free and useful resources accessible to security experts and threat actors alike, some of the most widely used and abused) open source intelligence tools are search engines like Google this toolkit will ease the process for collecting and storing on daily bases.

4. Methodology

First a clear strategy and framework for the acquisition and use of open source intelligence must be in place. It is not recommended to approach open source intelligence from the point of view of finding anything and everything that might be interesting or useful—as we have already discussed, the sheer amount of information available through open sources will simply overwhelm you.

To identify and remediate weaknesses in our network — and focus our energies specifically on accomplishing those goals. Second, describe a range of methods and techniques for the storage and processing of open source material. Again, the amount of information available is much too high to make manual processes even marginally efficient.

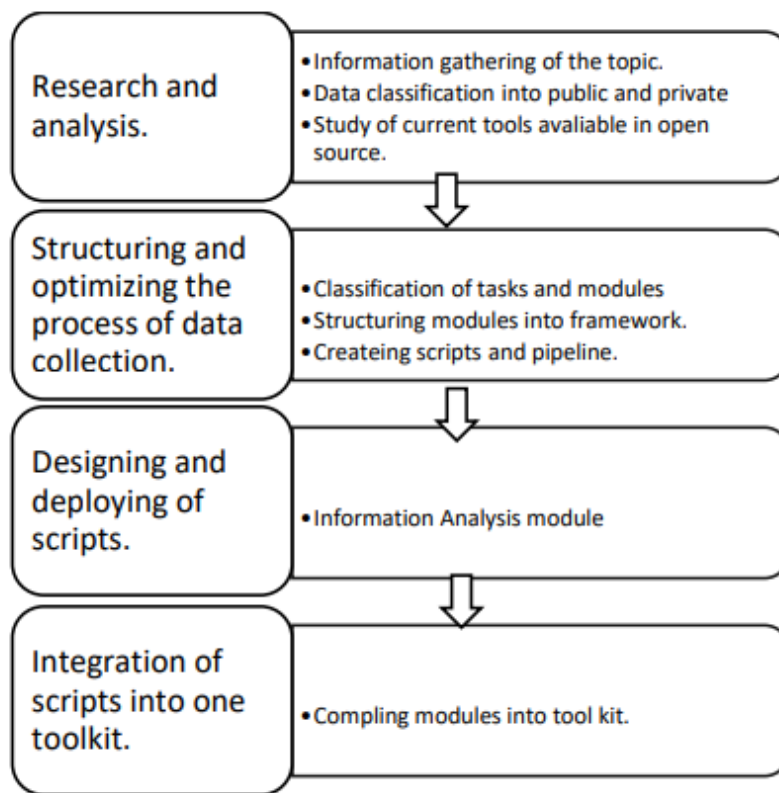


FIGURE 1: PROJECT WORKFLOW

Organization

Here is a brief overview of the report.

PART I: Introductory Material

This provides very basic background for the rest of the report.

Chapter 1: Introduction

This chapter offers some general perspective and includes some motivational examples, objectives and applications of project.

PART II: Planning and Implementation

The primary source of inspiration for the problems and algorithms discussed in this section is datamining. Methods, however, are usually appropriate for use in other applications in other fields, such as data collection, design and UI.

Chapter 2: Literature Survey

The chapter gives important related information available in standard Books, Journals, Transactions, Internet Websites etc. till date

Chapter 3: System Development

This chapter introduces Analysis/Design/Development/Algorithm and Model Development

Chapter 4: Performance Analysis

This chapter introduces Analysis of system developed either by at least two methods depending upon depth of standard. These methods normally used are Analytical /Computational/Statistical/Experimental/ or Mathematical. Results at various stages may be compared with various inputs. Output at various stages. Comparison of above results by at least two methods and justification for the differences or error in with theory or earlier published results.

Chapter 5: Conclusions

Conclusions and Future Scope also Applications Contributions. The innovative work/invention/new ideas generated from the analysis of the work which can be taken from the conclusions.

CHAPTER 2

LITERATURE SURVEY

2.1 INTELLIGENCE IN THE INTERNET AGE: THE EMERGENCE AND EVOLUTION OF OPEN SOURCE INTELLIGENCE (OSINT) [1]

2.1.1 AUTHOR: MICHAEL GLASSMAN, MIN JU KANG

2.1.2 PUBLICATION: ELSEVIER, MARCH 2019

2.1.3 Summary: This paper presents the idea of Open Source Intelligence (OSINT) as a significant segment for understanding human critical thinking in the 21st century. OSINT is from multiple points of view the consequence of evolving human–data connections coming about because of the rise and developing strength of the Internet and the World Wide Web in regular daily existence. This paper proposes that the Internet/Web changes the dynamic connection between what Cattell and Horn have recognized as the two general components of human knowledge: solidified insight and liquid insight. The Internet/Web open additional opportunities for getting to data and rising above over-decided social knowledge in critical thinking. This offers liquid insight, which frequently trails off in adulthood, another essentialness over the life expectancy. However, the diminishment of solidified insight, and particularly social knowledge, likewise presents various significant issues in support of durable, social cooperatives. The advancement of OSINT (utilizing devices and ethos made by the Open Source development of the most recent couple of many years) offers both a structure for coming to past the limits of customary social insight and approaches to make helpful, open, critical thinking networks. The Internet/Web will keep on making disarray and dread as we move further into this new age, yet additionally presents phenomenal opportunities for enlarging human astuteness in the event that we can get it and figure out how to tackle its latent capacity.

2.1.4 ADVANTAGES:

1. The Loughran & McDonald's and Affective Space Dictionaries are sentiment analysis dictionaries which contains words specially related to Stocks thus, the sentiment extraction is much more accurate.
2. The use of both EMH & AMH results in a more positive outcome as compared to using them separately

2.1.5 DISADVANTAGES:

1. The model's performance is predicted on the basis of a trading simulation and not real-life situations; thus, the performance metrics might be inaccurate.

2.2 A FRAMEWORK FOR THE SYSTEMATIC COLLECTION OF OPEN SOURCE INTELLIGENCE [2]

2.2.1 AUTHOR: LINE C. POUCHARD, JONATHAN M. DOBSON, AND JOSEPH P. TRIEN

2.2.2 PUBLICATION: ELSEVIER, 2016

2.2.3 SUMMARY: Following administrative headings, the Intelligence Community has been ordered to utilize Open Source Intelligence (OSINT). Endeavours are in progress to build the utilization of OSINT however there are numerous snags. One of these snags is the absence of apparatuses assisting with dealing with the volume of accessible information and determine its believability. We propose an interesting framework for choosing, gathering and putting away Open Source information from the Web and the Open Source Centre. Some information the board assignments are mechanized, record source is held, and metadata containing topographical directions are added to the archives. Examiners are consequently engaged to look, view, store, and break down Web information inside a solitary instrument. We present ORCAT I and ORCAT II, two usage of the framework.

2.2.4 ADVANTAGES:

1. The use of both the historical data as well as social media sentiments is sought to have predicted trends much more accurately than either used separately.

2.2.5 DISADVANTAGES:

1. The proposed model is only able to predict the trends in the data and thus, unable to predict the amount that can be gained by the users.

2. The sentiment dataset might be sparse, meaning that news for each day was not present thus resulting in inaccurate trend predictions.

2.3 DIGITAL FORENSIC INTELLIGENCE: DATA SUBSETS AND OPEN SOURCE INTELLIGENCE (DFINTOSINT): A TIMELY AND COHESIVE MIX [3]

2.3.1 AUTHOR: DARREN QUICKA, KIM-KWANG RAYMOND CHOOBA

2.3.2 PUBLICATION: ELSEVIER, 2018

2.3.3 SUMMARY: The authors in this paper discussed propels in innovations and changing patterns in buyer conduct have prompted an expansion in the volume, assortment, speed, and veracity of information accessible for computerized measurable investigation. An advantage of investigation of large computerized measurable information is that there might be case-related data contained inside dissimilar information sources. This paper presents a system for element distinguishing proof and open source data attachment to enhance information property from advanced measurable information subsets. Utilization of the system to test information brought about finding extra data identifying with the substances contained inside the computerized criminological information subsets, which prompted adding knowledge esteem identifying with the elements. Examination of certifiable information affirmed the possibility to increase the value of huge advanced criminological information to reveal different data and open source data. The outcomes exhibit the advantages of applying the cycle to accomplish more prominent comprehension of advanced scientific information in an opportune way.

2.3.4 ADVANTAGES:

1. CNN was observed to be a better model as it was able to capture the patterns in the provided dataset.

2.3.5 DISADVANTAGES:

1. The advantages of using a hybrid network were not considered for making predictions.

2.4 OPEN-SOURCE INTELLIGENCE AND PRIVACY BY DESIGN [4]

2.4.1 AUTHOR: BERT-JAAP KOOPSA, JAAP-HENK HOEPMANBC, RONALD LEENESA

2.4.2 PUBLICATION: ELSEVIER, 2015

2.4.3 Summary: In this paper, the creators proposed an exhibited by different papers on this issue, open-source insight (OSINT) by state specialists presents difficulties for security assurance and licensed innovation authorization. A potential system to deliver these difficulties is to adjust the plan of OSINT apparatuses to insert regulating necessities, specifically lawful prerequisites. The experience of the VIRTUOSO stage will be utilized to show this technique. Preferably, the specialized improvement cycle of OSINT instruments is joined with lawful and moral protections so that the subsequent items have a legitimately consistent plan, are adequate inside society (social installing), and simultaneously meet in an adequately adaptable way the fluctuating necessities of various end-client gatherings. This paper utilizes the insightful structure of protection plan procedures (limit, independent, total, stow away, advise, control, implement, and illustrate), contending that two methodologies for inserting lawful consistence appear to be encouraging to investigate specifically. One methodology is the idea of revocable security with spread duty. The other methodology utilizes an approach increase language to characterize Enterprise Privacy Policies, which decide suitable information taking care of.

2.4.4 ADVANTAGES:

1. The expansion in volume, and particularly sorts of information in account region gives opportunities to comprehend the financial exchange all the more effectively and makes the value expectation results better.
2. Since the framework proposed here is model independent, so the framework can be combined with any predictive model.
3. Through the combination of framework with the predictive model, the efficiency to the model is increased.

2.4.5 DISADVANTAGES:

1. The correlation among the time series is neglected as the observations are treated independently.

2.5 OSINT AS A PART OF CYBER DEFENCE SYSTEM [5]

2.5.1 AUTHOR: DMYTRO V. LANDE, ELLINA V. SHNURKO-TABAKOVA

2.5.2 PUBLICATION: ELSEVIER, 2015

2.5.3 Summary: The paper presents the aftereffects of examination on the advancement of crucial and applied standards for investigating data streams in worldwide PC organizations while leading open source insight (OSINT). The significance of this assignment, specifically, concerning the arrangement of digital protection, the boundaries of the cutting edge data space, the current hypothetical and innovative arrangements are validated. The depiction of methodological and instrumental methods for examination and demonstrating of data streams, appropriated content checking of worldwide organizations, the production of multilingual full-text information bases, investigation of the elements of topical data streams with the utilization of nonlinear investigation, programmed development of models of branches of knowledge in the field of network protection are introduced.

2.5.4 Advantages:

1. The increase in volume, and especially types of data in area provides chances to understand the OSINT more efficiently and makes the prediction results better.
2. Since the framework proposed here is model independent, so the framework can be combined with any predictive model.
3. Through the combination of framework with the predictive model, the efficiency to the model is increased.

2.5.5 Disadvantages:

1. The correlation among the time series is neglected as the observations are treated independently.

CHAPTER 3

SYSTEM DEVELOPMENT

3.1 GENERAL FACTS ABOUT WEB SCRAPING

A few meanings of web scratching came up during the writing research. Every one of the three beneath introduced definitions notice information extraction from various sources. They vary as the underlying hotspots for the separated data.

Now and again it is important to assemble data from sites that are planned for human peruses, not programming specialists. This procedure is known as "web scratching". The main definition makes reference to information sources, which are initially planned for human peruses. Such definition has substantiated itself as obsolete. With the development of computerized methods, the chance additionally emerged by extraction from programming meaningful sources. Nonetheless, it must be viewed as that the date of distribution was 2009. Application Programming Interface (API) Sources around then were constrained. Open API index accessible on Programmable Web site (Berlin, 2015) had approx. 750 accessible sources accessible, contrasted with 17175 recorded in 2017.

Web scratching, otherwise called web extraction or reaping, is a procedure to extricate information from the World Wide Web (WWW) and spare it to a document framework or database for later recovery or examination. Regularly, web information is rejected using Hypertext Transfer Protocol (HTTP) or through an internet browser. This is practiced either physically by a client or naturally by a bot or web crawler. Since a tremendous measure of heterogeneous information is continually produced on the WWW, web scratching is generally recognized as an effective and amazing method for gathering large information.

The present circumstance is all the more unequivocally depicted continuously definition, where Web Scraping is referenced as one of hotspots for large information assortment. In the definition is likewise referenced another term

Web Crawler. Web Crawling is acted in an alternate manner with various results. Figure 1 shows the two exercises. The procedure steps noticeable on the left side show that Web Crawling doesn't have a characterized remarkable objective and procedures any accessible information without pointing on explicit data. In correlation on the correct side, where the Web Scraper gets, forms and parses the information from a predefined source. Web Crawling isn't canvassed in this paper.

The definition beneath doesn't refer to numerous subtleties. Nonetheless, it briefly catches the exercises of Web scratching most exactly.

Information assortment from online sources got one of the statistical surveying strategies. It offers a lot quicker reaction, contrasted with an old-style studying. While Know it thinks of it as best to use conventional reviews, Web-scratching is viewed as savvy support for such instruments. To get far reaching

picture and to pick up information on the apparatuses in business sectors numerous sources ought to be utilized

3.1.1 PURPOSE OF WEB SCRAPING

Gigantic measures of source data, accessible on the World Wide Web, are still in the arrangement of a Hypertext Mark-up Language (HTML) page. Computerized extraction is troublesome on the grounds that the expected peruse was a human. This section presents the inspiration and reason for Information extraction through Web Scraping. Fast development of the World Wide Web has essentially changed the manner in which we offer, gather, and distribute information. Immense measure of data is being put away on the web, both in organized and unstructured structures. As to questions or research themes, this has brought about another issue never again is the worry of information shortage and detachment at the same time, somewhat, one of beating the tangled masses of online information. These uses are regularly just conceivable in light of the fact that the presence of mechanized Web Scraping. Without these procedures, it is difficult to gather the measure of information over and over and in sensible time.

3.1.2 ANALYSIS AND RESEARCH

Information assortment from online sources got one of the statistical surveying strategies. It offers a lot quicker reaction, contrasted with an old-style studying. While Know it thinks of it as best to use conventional reviews, Web-scratching is viewed as savvy support for such instruments. To get far reaching picture and to pick up information on the apparatuses in business sectors numerous sources ought to be utilized.

Shoppers are dynamic in the online world and offer their experience, disappointment, or inspiration. Organizations that desire to gain more from shoppers can include online wellsprings of data. Web scratching is one of the strategies to gather such information. Directed information assortment from e-shop and promoting servers assists with refreshing Indexes. Which depend on every now and again changed costs. Files worked through computerized Web scratching can offer increasingly visit update interims.

With the expanding significance and accessibility of on-line costs that we see today, it is normal to solicit whether the forecast from the customer value list (CPI), or related insights, may conveniently be processed more every now and again than existing month to month plans take into account. Wegman and Chapple (2013) utilized a little example of 338 Craigslist postings to consider the prevalence of optional dwelling units in the San Francisco Bay Area. At long last, Feng (2014) web-scratched 6,000 Craigslist postings to examine Seattle's lodging market.

3.1.3 SOCIAL MINING AND SENTIMENT ANALYSIS

Web based life is another wellspring of information that is essentially unique in relation to ordinary ones. Internet based life informational indexes are for the most part client produced, and are enormous, interlinked, and heterogeneous. Web-based social networking information can be acquired from openly

accessible sources by means of different methods, for example, scratching, utilizing website gave applications, and creeping. The chance of acquiring internet-based life information makes examine via web-based networking media information attainable. In Twitter Data Analytics, for instance, a start to-end process for Twitter information investigation is explained with four key advances: creeping, putting away, examining, and envisioning.

3.1.4 AUTOMATIC DATA COLLECTION

The developing significance of web-based exchanging requires considerably more value assortment from the web. Budgetary limitations, notwithstanding, require an increasingly proficient organization of existing HR to ace the extra remaining task at hand. Naturally extricated information, like the information gathered physically on the grounds that this considers a combination of the new technique into existing information cleaning, altering and coordinating procedures.

Internet based life information are much the same as regular information, which is potential fortune trove, yet requires information mining to reveal concealed fortunes.

3.2 LEGAL ASPECTS OF DATA EXTRACTION

3.2.1 LAW AREAS FOR WEB SCRAPER DEFENCE ACTION

The utilization of information got by Web Scraping is addressed by a few site proprietors. People or organizations dynamic in Web Scraping have a basic lawful resistance procedure. Sites are loaded up with information, which was distributed with the away from of being seen by others.

For instance, on account of the site called LinkedIn - the primary field-tested strategy is to distribute profiles and determine a pay through charging a segment of administrations offered to clients. The objective of clients is to open their CVs to an imminent business. In this manner, holding private information would not conform to their field-tested strategy, and by expansion any information accessible is consequently open. Web Scraping might break a few laws or agreement terms. The Table clarifies the law zones, where Web Scraper ought to be set up for a resistance activity.

3.2.2 WEBSITE TERMS AND CONDITIONS

Numerous organizations explicitly deny scratching inside their site terms and conditions. Regardless of whether they can authorize such terms is as yet muddled however relying upon conditions, a case for penetrate of agreement is conceivable.

3.2.3 COPYRIGHT, INTELLECTUAL PROPERTY RIGHTS

As scraping includes duplicating, it might prompt a case for copyright encroachment. Regardless of whether such a case has any benefits will rely upon the specific conditions because not all scratched information

meets all requirements for copyright insurance. Works, which are copyright, ensured incorporate unique scholarly and imaginative works, for example, PC programs, site designs, and photos. Files worked through computerized Web scratching can offer increasingly visit update interims.

3.2.4 DATABASE RIGHTS

A database right is encroached when all or a generous piece of a database is separated or re-used without the proprietor's assent. The rehashed extraction or re-usage of meagre pieces of a database which clashes with the ordinary utilization of the database may likewise encroach database rights. Encroachment of database rights may likewise apply when scratching catalogues or postings from outsider sites if the proprietor has brought about expenses in creating and looking after them.

3.2.5 TRADEMARKS

On the off chance that the scrubber repeats a site proprietor's (enrolled or unregistered) trademarks without their assent, the site proprietor could make a move guaranteeing trademark encroachment as well as going off. Going off forbids an outsider from selling products or carrying on business under a name, imprint, depiction, or in whatever other way that is probably going to misdirect, beguile or befuddle people in general into accepting that the product or business is that having a place with the brand proprietor.

3.2.6 DATA PROTECTION

Organizations hoping to utilize mechanized scratching procedures to gather data about people ought to know that they chance breaking nearby information insurance law in the event that they gather "individual information" (that is, any data that recognizes a living person). The focal issue is whether people have assented to their own information being gathered. Although information got from one site may in seclusion not be close to home information, when it is amassed from numerous sites, a business may accidentally wind up possessing individual information without the assent of the individual concerned. Utilization of such close to home information will encroach information assurance laws

3.2.7 CRIMINAL DAMAGE

It is an offense to make criminal harm a PC (counting harm to information) or to utilize a PC to get to information without authorisation. In like manner, information scratching could be a criminal offense as the site proprietor has not approved access to the information.

3.3 GENERAL LEGAL ASPECTS

In outline, there is no bit of enactment which disallows Web Scraping to accumulate data. The site proprietors, nonetheless, may have legitimate rights against the organization under protected innovation law and agreement law. Each case will turn on its own realities however and this is a lot of ward upon what data is scratched from the sites. Organizations ought to be careful with authoritative arrangements which they have consented to in regard of a site's terms of utilization these may preclude the client from taking and utilizing the information off the site. The best way to be sure that the privileges of a site proprietor have not been encroached is to get their express agree to the screen scratching and ensuing utilization of the data. The general result of illegal Web Scraping could be for the Website proprietor negative. The chance of less guests, less connections from content aggregator sites and less salary from publicizing. In this way, Data hosts should possibly utilize legitimate activities against scrubbers when the scrubber presents a danger to the information host's centre business and the information have has a enough case to win lawfully against the scrubber. From the law point of view is important to alter the terms of utilization on the sites.

Contingent upon the main role, diverse Web Scraping strategies can be utilized, taken measure of information, periodicity and required result into thought. Web scrubbers have an expansive determination of apparatuses to choose from. The venture doesn't comprise just from the specialized arrangement and execution. By all Web Scraping tasks ought to be the lawful part of the employment analysed and essential advances distinguished. Information hosts ought to consistently survey the advantage scrubbers can give and adopt a sober minded strategy to the individuals who scratch their information. Web Scrapers should hold the association with the Data has and permit ID of the Data Host as a wellspring of introduced data.

3.4 METHODS OF WEB SCRAPING

The techniques for Web Scraping advanced along with the World Wide Web. Not every single recorded technique was accessible toward the start. There are two guides to make reference to, on the grounds that these are by and by the most utilized strategies. Since 2000 the Document Object Model (DOM) turned out to be progressively mainstream in DHTML. A more extensive acknowledgment later permitted the HTML Parsing procedure to develop to DOM Parsing. Second model are Application Programming Interfaces (APIs). This strategy is the most youthful on the rundown, the development of accessible substance APIs is dated from 2005. Agreeing to ProgrammableWeb.com the quantity of APIs includes developed inside 8 years from 0 to 10302.

3.4.1 MANUAL SCRAPING

Manual scratching is yet a choice in explicit circumstances. These circumstances are:

- When the measure of information is negligible,
- When the information being scratched doesn't require a redundant assignment,
- When setting up robotized scratching would take longer than the information assortment itself.
- Possibly safety efforts or explicit attributes of the site don't permit mechanized techniques.

3.4.2 HTML PARSING

Sites don't generally give them confine agreeable organizations, for example, .csv or .json documents. HTML Pages are made by the server as a reaction to a client 's demand. Now server programming isn't pertinent, rather the yield in the program is significant. Examination of the HTML structure (basic page test gave in the Figure 3) in the gave page will show rehashed components. With a programming language content or Web Scraping apparatus, each page with comparative example can be utilized as a hotspot for information.

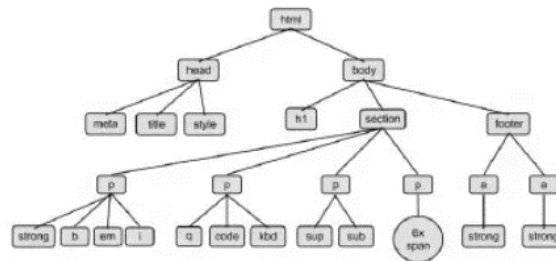


FIGURE 2: HTML PARSING

3.4.3 DOM PARSING

Archive Object Model (DOM) Parsing is an advancement of HTML Parsing dependent on improvements of the language and programs which lead to the presentation of the Document Object Model. DOM is vigorously utilized for Cascading Stylesheets (CSS) and JavaScript. Incorporation of DOM uncovered additional opportunities for tending to some pieces of the website page. Figure 4 shows compartments with their own DOM addresses. These are utilized in Web Scraping for simpler route through page content.

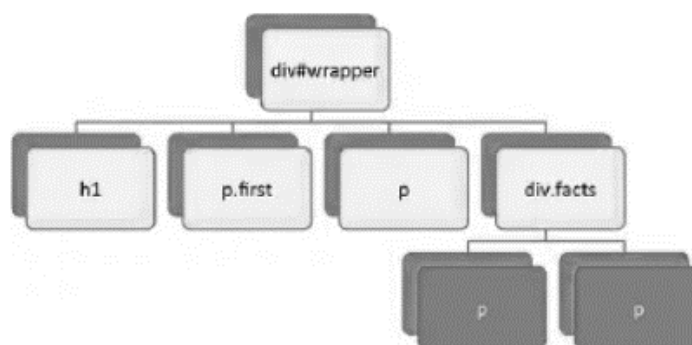


FIGURE 3: DOM PARSING

3.4.4 XPATH

Comparable tending to probability as DOM gives XPath (XML Path Language). The name proposes a use for XML archives. It is pertinent additionally to HTML design. Way requires a more definitely organized site page than DOM and has a similar chance to address fragments inside the website page. Figure 5 shows the report structure as deciphered in XPath.

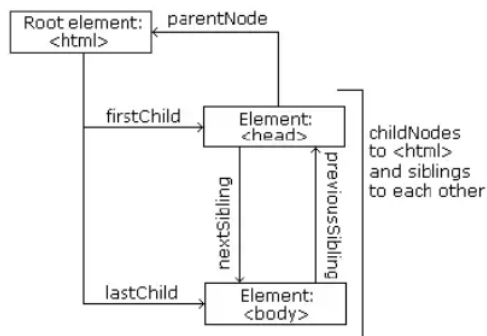


FIGURE 4: XPATH

3.4.5 API'S

While the past strategies work to scratch intelligible yields, Application Programming Interface (API) expects an application as a correspondence accomplice. In this manner, APIs are frequently named as machine-discernible interfaces (versus intelligible). Indeed, even APIs were presented a lot later than the WWW, and their development was quick. The universe of APIs is divided. For a basic review and direction were API Directories made. The vast majority of the accessible APIs are enrolled and portrayed in the index with important connects to the sources. Two instances of such indexes are - Programmable Web(<https://www.programmableweb.com>) and APIs (<https://apis.guru/>).API Directories likewise give their own API, which permits clients to look in their database for API Sources. A standard HTTP Request sent to an API Endpoint restores an answer from server. Every apus has its own determination and choices. The arrangement of the appropriate response can be set as alternative in the solicitation. The most broadly utilized organization for API correspondence is JSON.

3.5 AVAILABLE SOFTWARE TOOLS

This section will present a few distributed programming devices for Web Scraping. We can accept, that there are more devices which have been produced for inner reasons for people and organizations and along these lines have not been distributed. Along these lines, this rundown does exclude all accessible programming, because of the multiplication of utilizations and stages. Following areas dependent on programming type are shaped:

- Cloud Software
- Desktop Software
- Programming libraries
- Browser Extensions

3.5.1 CLOUD SOFTWARE

Cloud arrangements give a User Interface through a Web program while the application backend dwells on the cloud server. Such a design constrains the client end equipment necessities to their base. Huge measure of information can be scratched without having own PC on. Bigger tasks can be made without extra equipment or broadened web transmission capacity. In certain circumstances, is significant, where is the cloud found. For instance, some website pages may not be available from Europe in their full range. A Cloud arrangement associated through a US based ISP might have the option to get to different areas of such a site.

3.2.1 DESKTOP SOFTWARE

Site information are downloaded, parsed and spared locally. Work area Applications require a broadband web association. Web connection of the PC will legitimately impact the preparing time of Web Scraping task. Contrasted with Cloud Software for Desktop Software further developed equipment is required.

3.5.1 PROGRAMMING LIBRARIES

Webpage data are downloaded, parsed and saved locally. Work territory Applications require a broadband web affiliation. Web association of the PC will really affect the getting ready time of Web Scraping task. Appeared differently in relation to Cloud Software for Desktop Software further created gear is required. Workstation with Random access memory (RAM) size greater than 8GB is proposed as least.

3.5.1.1 SCRAPY

Scrapy is one of the propelled web scratching systems accessible. The Framework is written in Python. It is an application structure and it gives numerous orders to make own applications and utilize those for Web Scraping ventures.

Scrapy is a web creeping structure for designer to compose code to make bug, which characterize how a specific website (or a gathering of destinations) will be rejected. The greatest element is that it is based on Twisted, an offbeat systems administration library, so Scrapy is executed utilizing a non-blocking (otherwise known as nonconcurrent) code for simultaneousness, which makes the insect execution is incredible.

Broad documentation is accessible to help engineers who wish to coordinate it into their own applications. The decision between a Local Desktop Application or a Cloud Solution is not a constraint, both can be fuelled by Scrapy code. One of the introduced Cloud Solutions urges the client to move their Scrapy applications over to cloud foundation.

3.5.1 WEB SCRAPER

Web scraper is a program augmentation accessible for Google Chrome, that can be utilized for web scratching. Sitemap characterizes on how a site ought to be explored and what information ought to be separated. It can scratch various pages at the same time and even has dynamic information extraction capacities. Web scrubber can likewise deal with pages with JavaScript and Ajax, which makes it all the more impressive. The instrument removes information to a CSV record or CouchDB. For bigger Web Scraping occupations, the designer offers a cloud rendition. The Google Chrome extensionist additionally utilized as a test domain, before scratching occupations are relegated to the cloud variant.

It is hard to contrast the recorded apparatuses with one another. Some are intended for specialists and some are appropriate for undertakings. Program expansions have a place with the specialist's class. To gather information from a couple of sites fora brief research or venture, these apparatuses are all that could possibly be needed. Work area Applications have an expansive arrangement of functionalities. That is reflected in the significant expense. Cloud arrangements offer a comparative arrangement of capacities. Contrasted with Desktop Software Cloud Solution offer 3 fundamental favourable circumstances:

- Month to month charging dependent on asset use rather than introductory high venture
- low equipment necessities on own Server/PC
- Own data transmission is utilized uniquely for work arrangement and controlling cloud Software as of now offers the most available approach to start and test web scratching alternatives. The most mind-boggling alternative are systems. For systems, the used language and the accessible documentation is significant for the planned engineers.

3.6 ALGORITHM DESIGN

The most critical factor in the effectiveness of any open source intelligence project is the existence of a specific strategy—when you know what you are aiming to do and set goals accordingly, it would be far more possible to find the most effective methods and strategies.

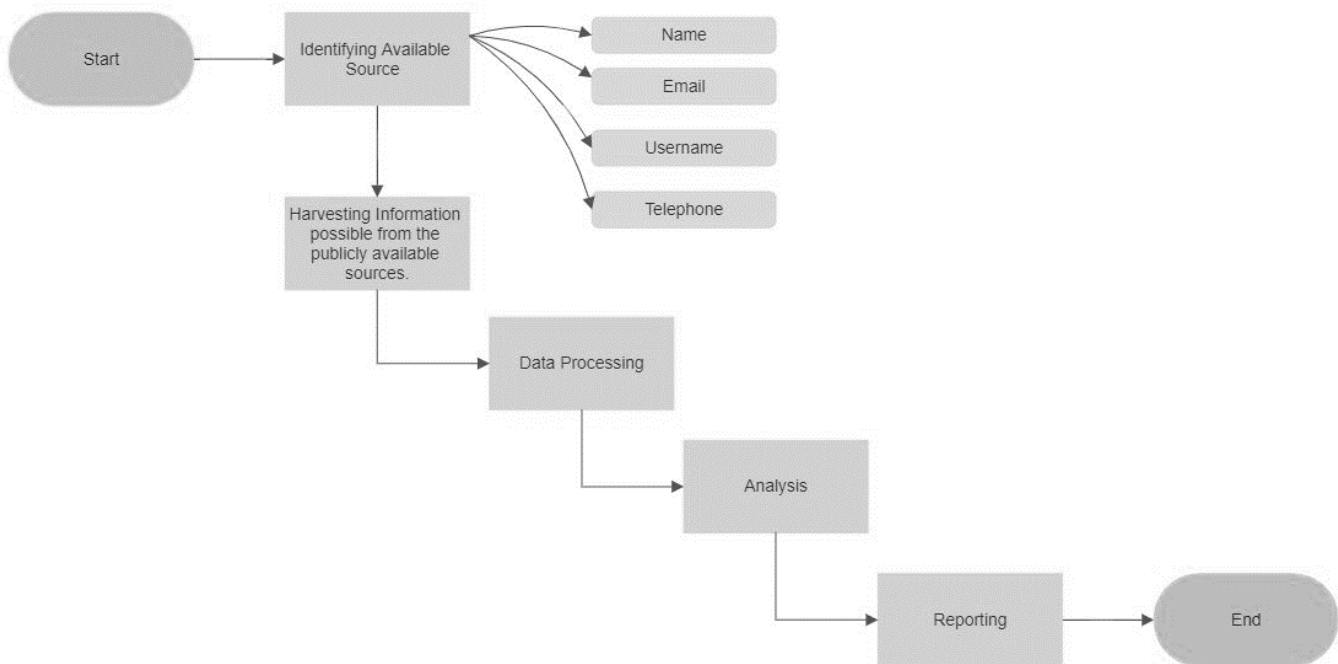


FIGURE 5: ALGORITHM

3.7 SYSTEM DESIGN

Whatever the aims, open source intelligence will be of great benefit to all security disciplines. At the end of the day though, discovering the best mix of resources and strategies for your needs will take time, as well as a degree of trial and error. The methods and strategies that you use to locate dangerous assets are not the same as those that will help you trace the danger warning or connect data points through a range of sources.

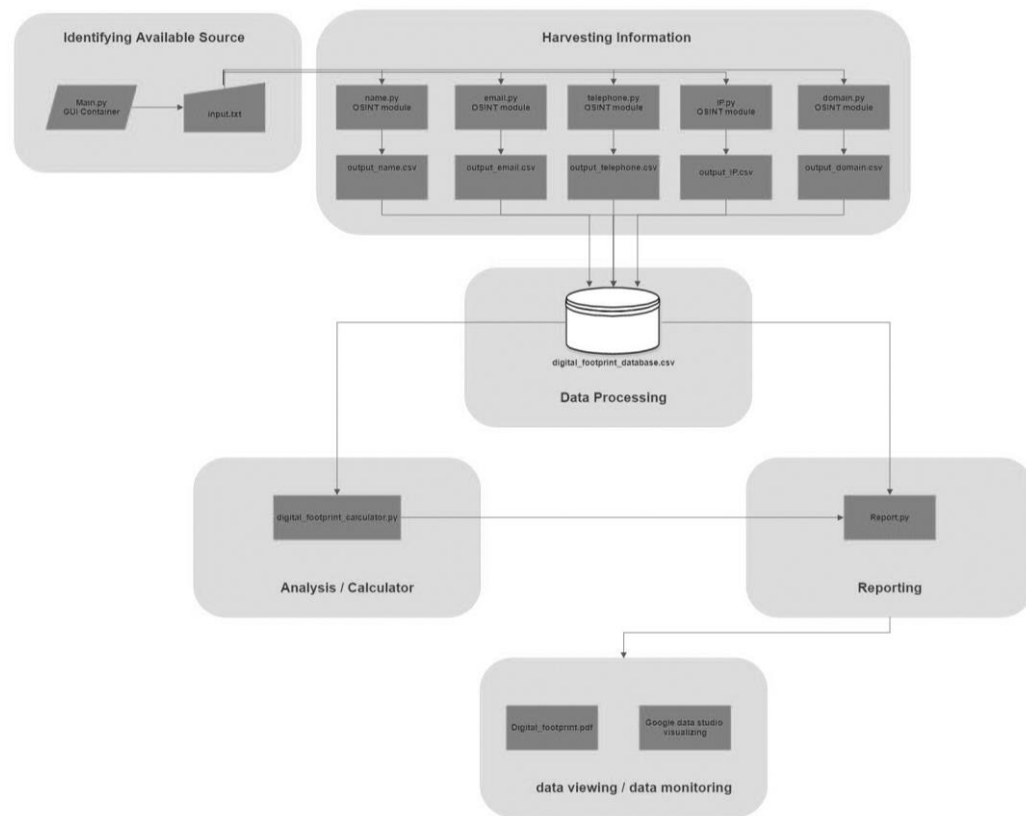


FIGURE 6: SYSTEM DESIGN

CHAP

PERFORMANCE ANALYSIS

4.1 REQUIREMENT ANALYSIS

A harvester tool to get email and domain-related information. One can be very helpful to get information. If the information is obtained, it allows links and can unmask secret relationships between names, email addresses, aliases, businesses, domains, record owners, affiliations and other information that may be valuable in the investigation, or to search into possible future issues. The software itself runs on python, meaning it operates on Windows, Mac and Linux systems.

4.1.1 Developers operating in Python have access to a popular Recon-ng tool written in that language. Its gui looks very similar to the common Metasploit System which is designed to reduce the learning curve for those who have familiarity with it. It also has an integrated support feature, which is absent in many Python modules, so developers should be able to pick it up easily.

4.1.2 Crafted to allow Python developers to scan publicly available information and return positive results, it has a very flexible architecture with a lot of built-in features. Popular activities such as streamlining performance, communicating with databases, making http requests, and handling API keys are all part of the gui. Instead of programming to perform searches, developers simply select which functions they want to execute and create an automatic module in just a few minutes.

4.2 TECHNICAL REQUIREMENTS

4.2.1 JUPYTER NOTEBOOK [12]

Jupyter Notebook is a web application used to create live code, visualizations, etc. using Python. On account of the blend of code and content components, these archives are the perfect spot to unite an examination depiction, and its outcomes, just as, they can be executed to play out the information investigation progressively.

4.2.2 NUMPY [13]

NumPy is the key group for legitimate enrolling with Python. It contains notwithstanding different things: an inconceivable N-dimensional group object, present day (telecom) limits, and diverse accommodating direct factor-based math, Fourier change, and discretionary number capacities.3.2.3 Pandas [14]

Pandas is an open source library for Python which gives an elite, simple to utilize information structures and information investigation instruments. Pandas has an element called Data Frames which stores the datasets and support various operations on the dataset.

4.2.4 MATPLOTLIB [15]

Matplotlib is a Python 2D plotting library that is utilized to create excellent figures and charts. It very well may be utilized to effectively create plots, histograms, bar diagrams, scatterplots, and so on in shortcodes. By and large, the PyPlot module is utilized to plot figures.

4.2.5 SEABORN [16]

Seaborn is a Python data portrayal library reliant on matplotlib. It gives a high-level interface to drawing engaging and edifying verifiable plans. This library is firmly incorporated with pandas information structures as well.

4.2.6 SCIKIT-LEARN [17]

Scikit-learn is a set of easy and powerful open-source software for data mining and data processing. It includes various modules for pre-processing data, machine learning models and metrics.

4.2.6 REPORTLAB [18]

Lab Report more than 18 years of involvement with creating PDF assets online utilizing a scope of devices, like JSON and XML. Over years, we have assembled a fast method to give organizations an archive creation framework: you make another parcel in JSON design at that point present it on a web URL that changes it to a PDF.

4.2.6 PYQT [19]

PyQt is a collection of Python version 2 and 3 dependencies for the Qt Company application system and runs for all platforms supported by Qt, like Windows, MacOS, Linux, iOS and Android. PyQt5 is provided by Qt v5. PyQt4 accepts Qt v4 and builds toward Qt v5. Strings are implemented as a collection of Python modules which contain more than 1,000 classes.

4.2 REQUIREMENTS FOR USER INTERFACE

There are many graphical user interface (GUI) toolkits which you can use in the Python language. The major three among them are Tkinter, WxPython, and PyQt. Each of these development tools will operate with Windows, MacOS and Linux, and PyQt will have the optional ability to work on a smart phone.

A interface is also an application that has keys, windows, and several other widgets that the user can use to communicate with the software. A clear example of this was a web browser.

4.2.1 APPLICATION FRONTEND EDITOR

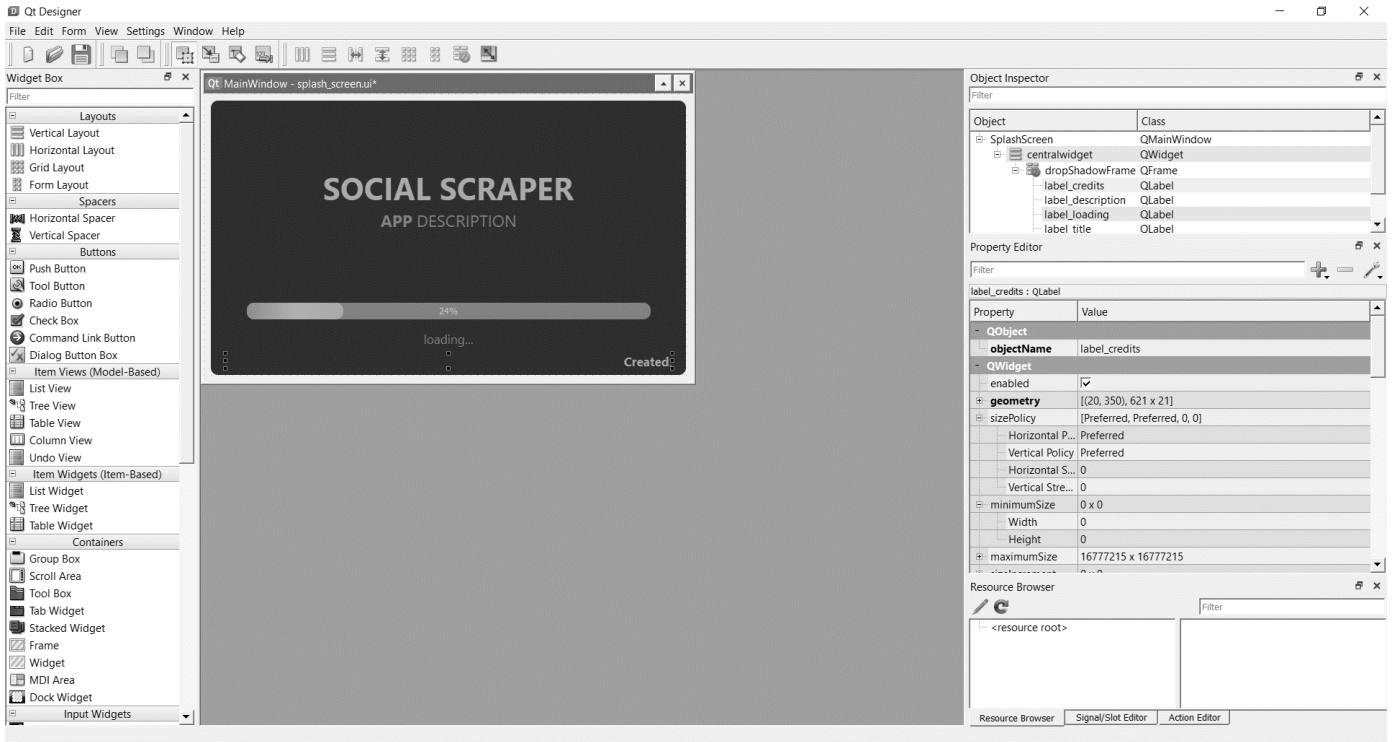


FIGURE 7: APPLICATION PRE-LOADER

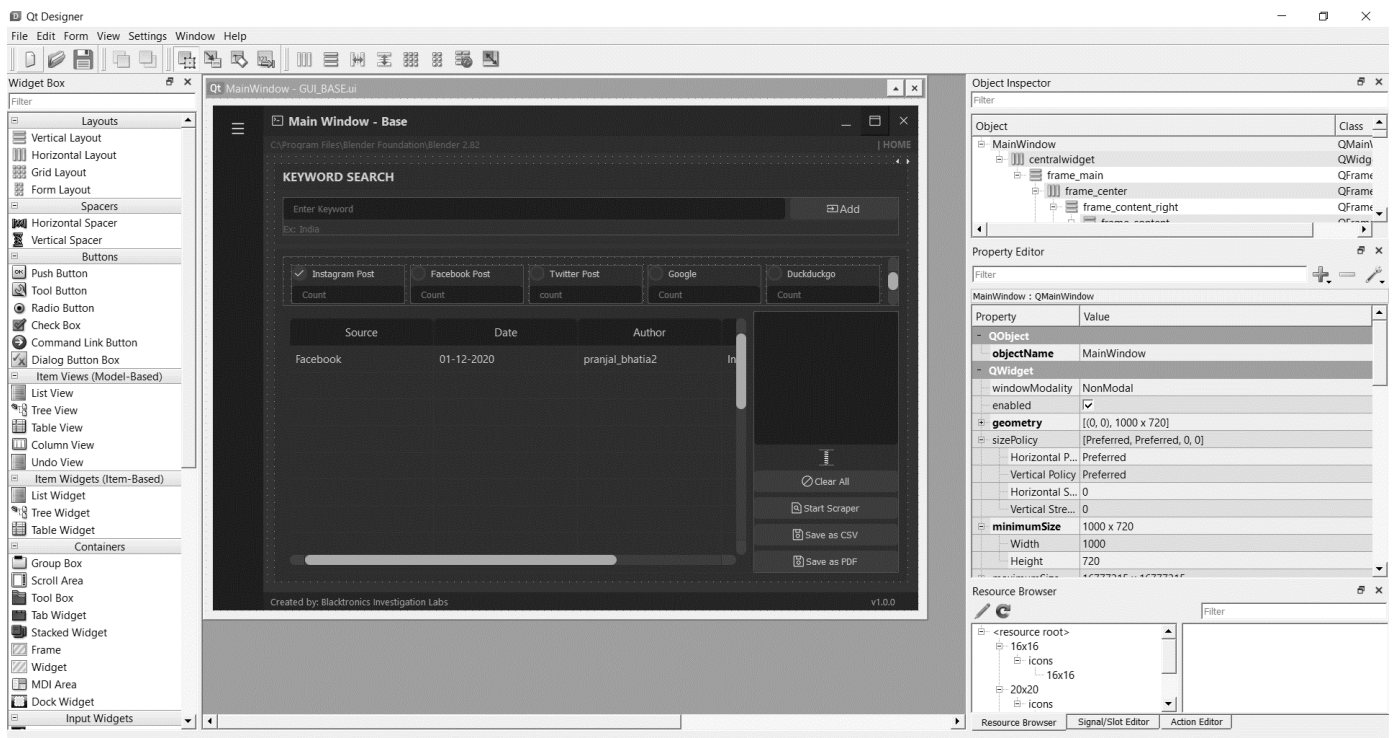


FIGURE 8: APPLICATION MAIN SCREEN

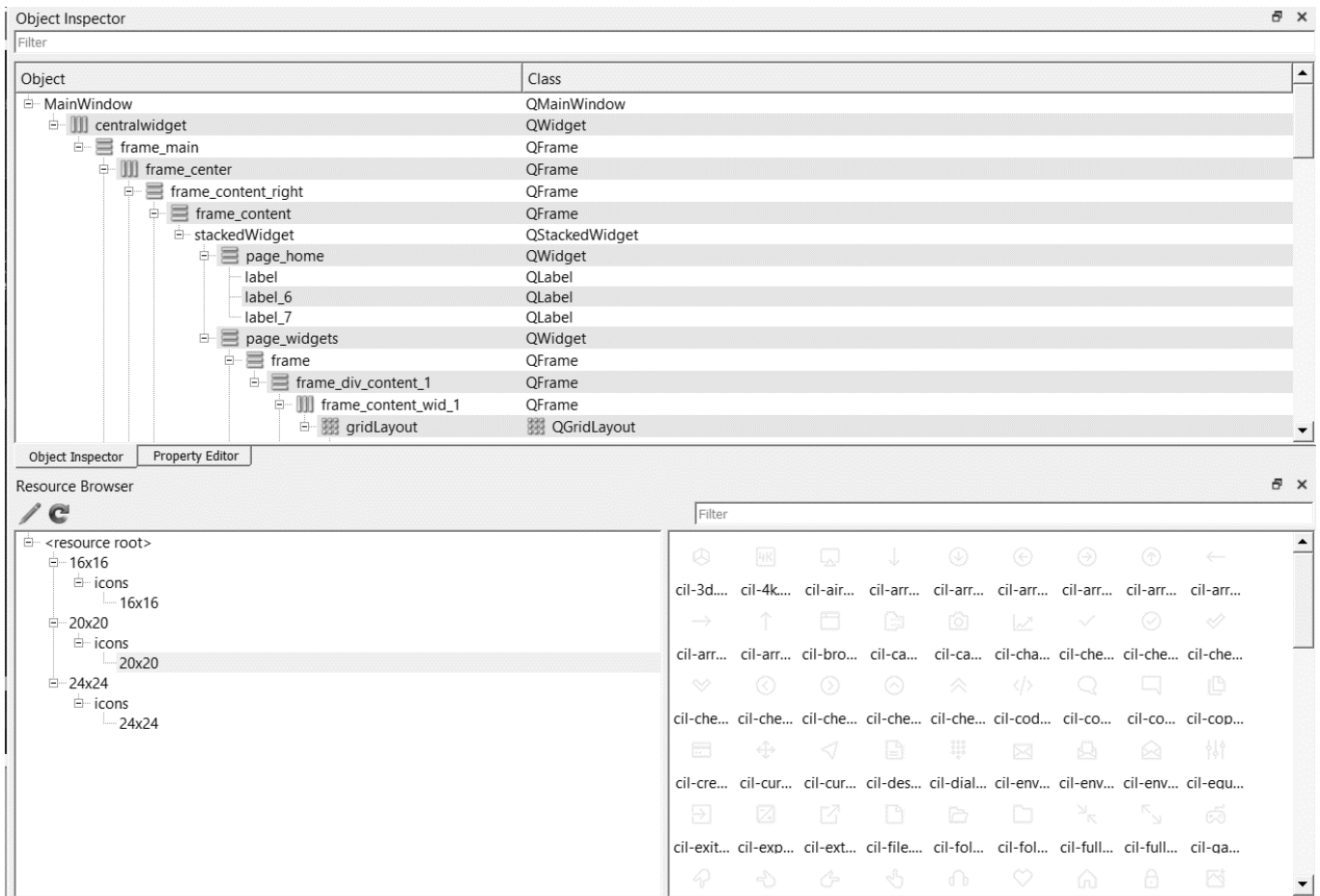


FIGURE 9: OBJECT INSPECTOR

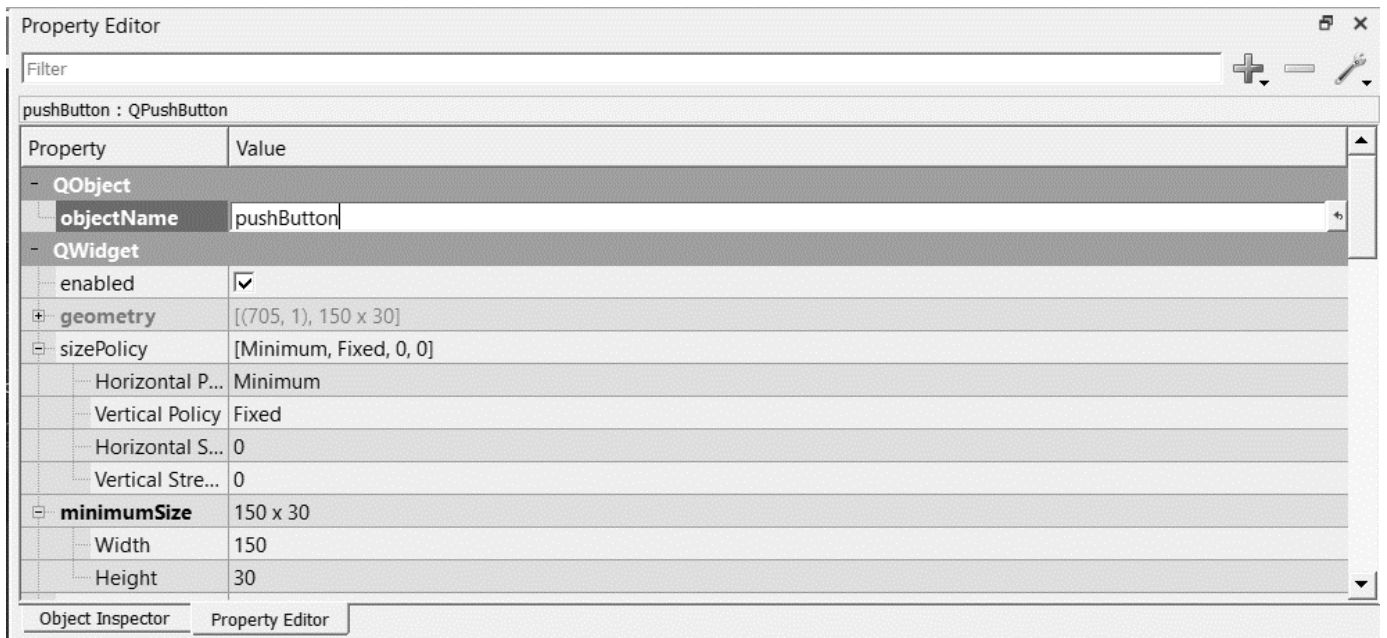


FIGURE 10: PROPERTY EDITOR

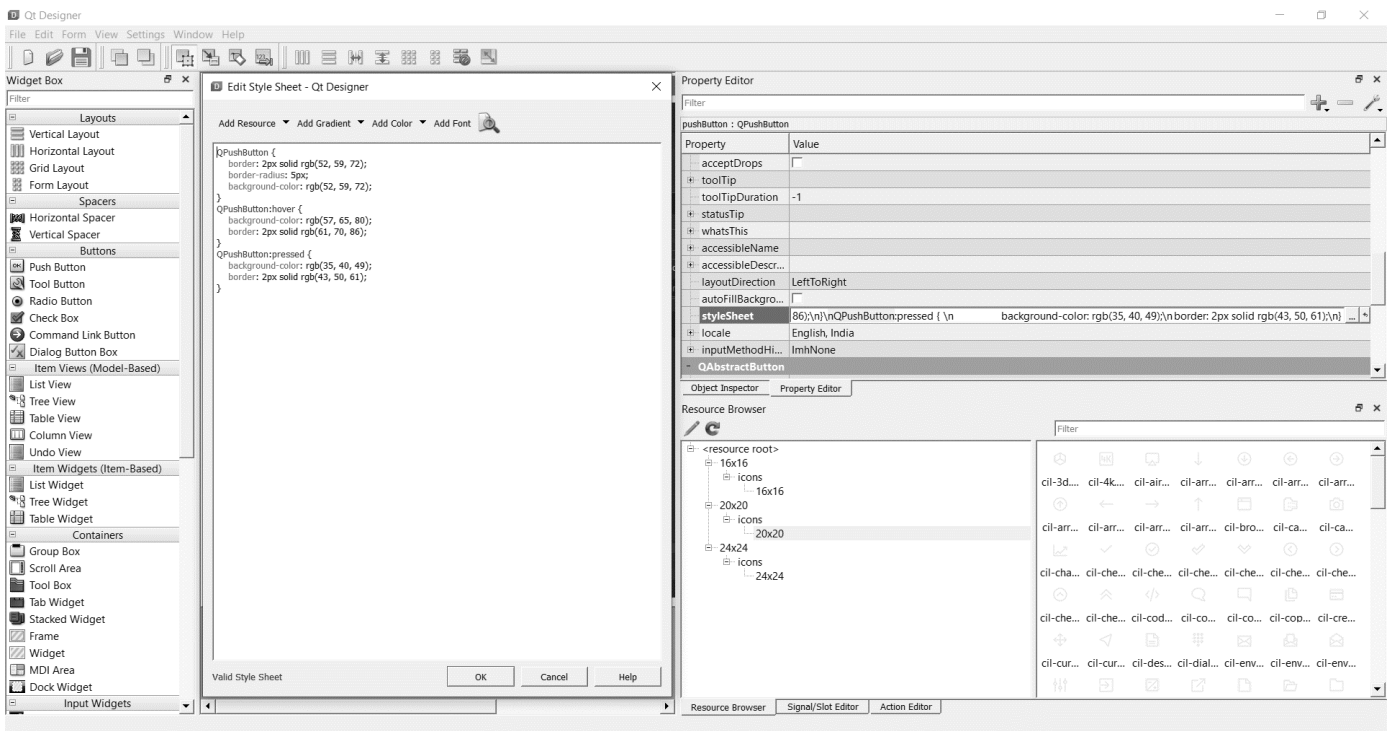


FIGURE 11: STYLE SHEET

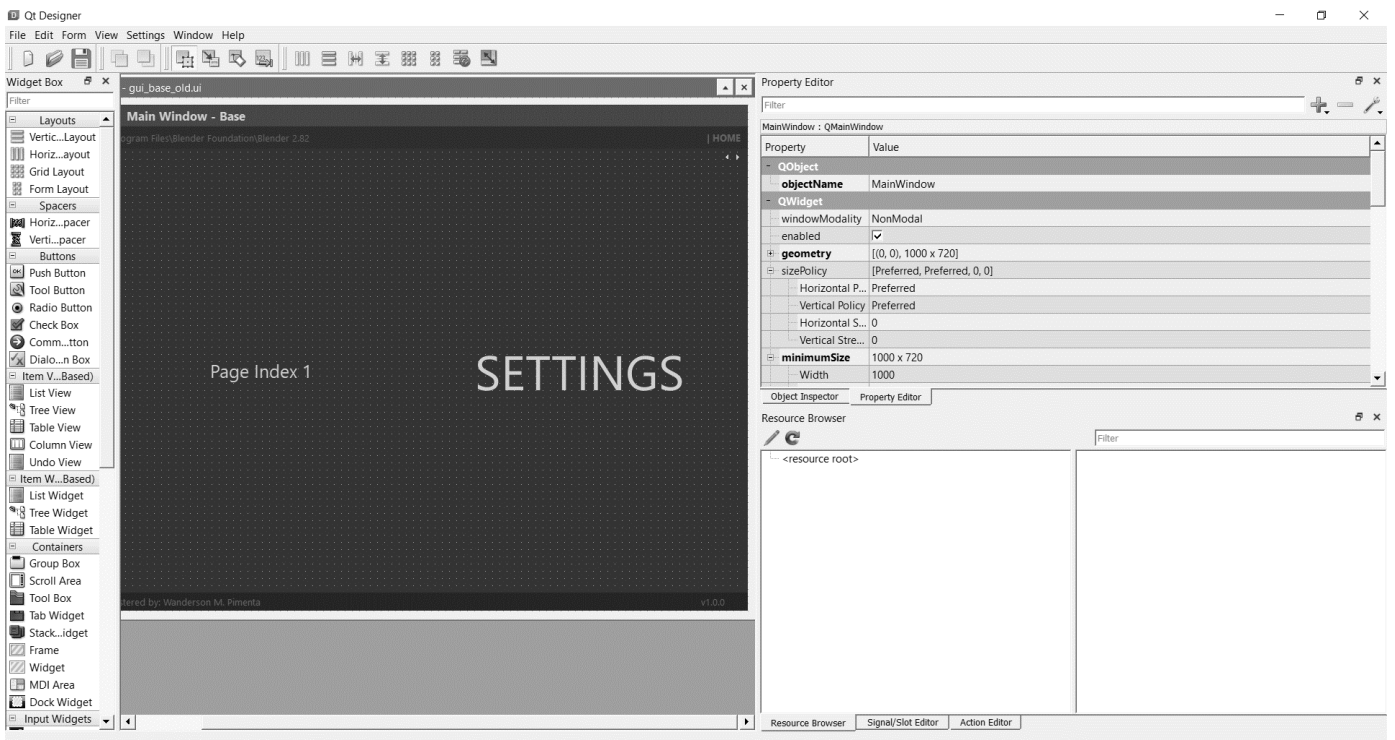


FIGURE 12: SETTINGS PAGE

The application is built upon PyQt python module can be launched in platform like windows, MacOS. While these resources have a plethora of OSINT data, there are several other methods and strategies ready to support you fully know the public footprint of your organisation. An ideal platform for finding more resources is the OSINT System which provides a web-based interface that disintegrates numerous topics of interest to OSINT investigators and links you with tools to help you sniff out the knowledge they need.

CHAPTER 5

CONCLUSION & FUTURE DIRECTIONS

This project uses technique like search engine dorking queries and other OSINT information gathering techniques to acquire data from the internet. Modules including social media post scrapers and public personal information scrapers, structure all the information into a dataset which is available to be saved as a CSV file or PDF report.

This project is platform independent because it is built on python and can be run on Windows and MacOS. There are a few pages of findings that are seen when we search the internet. We just look at the first page and research. But if you've ever wondered what the rest of the hundred pages of the result is? "Information!" open source intelligence (OSINT) gathering and analysis toolkit gets this information using a range of modules.

Many people think using their favorite Internet search engine is enough to find the data they need and do not realize that most of the Internet is not indexed by search engines.

With this toolkit we will implement effective methods of finding these data. By integration of automation OSINT collection and deep data analysis scripts this toolkit will provide processed report.

To create automated OSINT data gathering and analysis scripts and integrating all into one toolkit this project is divided into four phases Research and analysis, Structuring and optimizing the process of data collection, Designing and deploying of scripts, Integration of scripts into one toolkit.

This project uses technique like search engine dorking queries and other OSINT information gathering techniques to acquire data from the internet.

Modules including social media post scrapers and public personal information scrapers.

5.1 CONCLUSION

This tool can perform optimized data scraping and orient it in structural formats like CSV and PDF. The user interface used is of PyQt v5 and is platform independent.

5.2 FUTURE SCOPE

Furthermore, modules will added in the project, Not each user requires threats or intense, specialized penetration. There really is no reason to continue to clamp down on data protection over a series of months of work if the information they seek is made available via a freely open channel.

These resources will help organisations understand what data is required about everyone, their channels, data and customers. Finding the information quickly is important since it will cause it to be deleted so anyone can misuse it.

REFERENCES

- [1] Andrea Picasso, Simone Merello, Yukun Ma, intelligence in the internet age: the emergence and evolution of open source intelligence (OSINT) Elsevier, Volume 35, 2019, <https://doi.org/10.1016/j.eswa.2019.06.014> , pp.60-70.
- [2] Aparna Nayak, Prediction Models for OSINT, Procedia Computer Science, Elsevier, Volume 89, 2016, <https://doi.org/10.1016/j.procs.2016.06.096> , pp. 442-450.
- [3] Hiransha M, NSE Stock Market Prediction Using OSINT Models, Procedia Computer Science, Volume 132, 2018, <https://doi.org/10.1016/j.procs.2018.05.050> , pp.1351-1362.
- [4] Huiwen Wang, Shan Lu, Aggregating multiple types of complex data in stock market prediction, Knowledge-Based Systems, Elsevier, <https://doi.org/10.1016/j.knosys.2018.10.035> , pp. 193-204.
- [5] Yefeng Ruan,, Using Twitter trust network for stock market analysis, Knowledge Based Systems, Elsevier, vol 125, April 2018, <https://doi.org/10.1016/j.knosys.2018.01.016>, pp. 207-218.
- [6] Saishruthi Swaminathan, Linear Regression – A Detailed View, Medium, February 2018 <https://towardsdatascience.com/linear-regression-detailed-view-ea73175f6e86>
- [7] Aparna Nayak, Prediction Models for Indian Stock Market, Procedia Computer Science, Elsevier, Volume 89, 2016, <https://doi.org/10.1016/j.procs.2016.06.096> , pp. 442-450.
- [8] Hiransha M, NSE Stock Market Prediction Using Deep-Learning Models, Procedia Computer Science, Volume 132, 2018, <https://doi.org/10.1016/j.procs.2018.05.050>, pp.1351-1362.
- [9] Shan Lu , Huiwen Wang, Aggregating multiple types of complex data in stock market prediction, Knowledge-Based Systems, Elsevier, <https://doi.org/10.1016/j.knosys.2018.10.035> , pp. 193-204.
- [10] Yefeng Ruan,, Using Twitter trust network for stock market analysis, Knowledge Based Systems, Elsevier, vol 125, April 2018, <https://doi.org/10.1016/j.knosys.2018.01.016>, pp. 207-218.
- [11] Shashank Gupta, Sentiment Analysis: Concept, Analysis and Applications, <https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17>
- [12] Saishruthi Swaminathan, Linear Regression – A Detailed View, Medium, February 2018 <https://towardsdatascience.com/linear-regression-detailed-view-ea73175f6e86>
- Twitter API, <https://developer.twitter.com/en/docs>
- Jupyter Notebook, www.jupyter.org
- NumPy, <https://numpy.org/devdocs/user/quickstart.html>
- Selenium, **Error! Hyperlink reference not valid.**
- Pandas, <https://pandas.pydata.org/pandas-docs/stable/>
- Reportlab, <https://www.reportlab.com/docs/reportlab-userguide.pdf>
- PyQt, <https://wiki.python.org/moin/PyQt>

APPENDICES A

A.1 CODE TO INPUT CREDENTIALS AND KEYWORDS

```
with open('facebook_credentials.txt') as file:
    EMAIL = file.readline().split('"')[1]
    PASSWORD = file.readline().split('"')[1]

with open('input.txt') as file:
    KEYWORD = file.readline().split('"')[1]
    COUNT = file.readline().split('"')[1]

def _extract_post_text(item):
    actualPosts = item.find_all(class_="a8c37x1j ni8dbmo4 stjgntxs l9j0dhe7")
    text = ""
    for actualPost in actualPosts:
        text = actualPost.get_text()
        #print(text)
    text.encode('utf-8')

    return text
```

A.2 CODE OT EXTRACT POST ID

```
def _extract_post_id(item):
    postIds = item.find_all(class_="oajrlxb2 g5ia77u1 qu0x051f esr5mh6w e9989ue4
r7d6kgcz rq0escxv nhd2j8a9 a8c37x1j p7hjl8o kvgmc6g5 cxmmr5t8 oygrvhab hcuky3x
jb3vyjys rz4wbd8a qt6c0cv9 a8nywdso ilao9s8h esuyzwwr flsip0of lzcic4wl gmql0nx0
p8dawk7l")
    post_id = ""
    for postId in postIds:
        try:
            #"https://www.facebook.com" +
            post_id = f"{postId.get('href')}"
            #print(post_id)
        except:
            post_id = "NA"

    return post_id
```

A.3 CODE TO EXTRACT LINKS

```
def _extract_link(item):
    postLinks = item.find_all(class_="oajrlxb2 g5ia77u1 qu0x051f esr5mh6w e9989ue4
r7d6kgcz rq0escxv nhd2j8a9 nc684nl6 p7hjl8o kvgmc6g5 cxmmr5t8 oygrvhab hcuky3x
jb3vyjys rz4wbd8a qt6c0cv9 a8nywdso ilao9s8h esuyzwwr flsip0of lzcic4wl oo9gr5id
gpro0wi8 lrazzd5p dkezs63")

    link = ""
    for postLink in postLinks:
        link = postLink.ge
```

A.4 FUNCTION TO RETURN POST ID

```
def _extract_post_id(item):
    postIds = item.find_all(class_="oajrlxb2 g5ia77u1 qu0x051f esr5mh6w e9989ue4
r7d6kgcz rq0escxv nhd2j8a9 a8c37x1j p7hjln8o kvgmc6g5 cxmmr5t8 oygrvhab hcukyx3x
jb3vyjys rz4wbd8a qt6c0cv9 a8nywdso ilao9s8h esuyzwwr flsip0of lzcic4wl gmql0nx0
p8dawk7l")
    post_id = ""
    for postId in postIds:
        try:
            #"https://www.facebook.com" +
            post_id = f"{postId.get('href')}}"
            #print(post_id)
        except:
            post_id = "NA"

    return post_id
```

A.5 MAIN FUNCTION

```
if __name__ == "__main__":
    parser = argparse.ArgumentParser(description="Facebook Page Scraper")
    required_parser = parser.add_argument_group("required arguments")
    #required_parser.add_argument('-page', '-p', help="The Facebook Public Page you
want to scrape", required=True)
    #required_parser.add_argument('-len', '-l', help="Number of Posts you want to
scrape", type=int, required=True)
    optional_parser = parser.add_argument_group("optional arguments")
    optional_parser.add_argument('-infinite', '-i',
                                help="Scroll until the end of the page (1 = infinite)
(Default is 0)", type=int,
                                default=0)
    optional_parser.add_argument('-usage', '-u', help="What to do with the data: ")

    #TODO: rewrite parser
    if args.usage == "WT":
        with open('output.txt', 'w') as file:
            for post in postBigDict:
                file.write(json.dumps(post)) # use json load to recover

    elif args.usage == "CSV":
        with open(KEYWORD+".csv", 'w', encoding="utf8") as csvfile:
            writer = csv.writer(csvfile)
            writer.writerow(['Post Link', 'Profile Link', 'Post', 'Image'])

            for post in postBigDict:
                writer.writerow([ post['PostId'], post['Link'], post['Post']
,post['Image']])

    else:
        for post in postBigDict:
            print(post.encode("utf-8"))

    print("Finished")
```

A.6 PYTHON CODE TO DISPLAY INTERFACE

```
self.dropShadowFrame.setFrameShape(QFrame.StyledPanel)
self.dropShadowFrame.setFrameShadow(QFrame.Raised)
self.label_title = QLabel(self.dropShadowFrame)
self.label_title.setObjectName(u"label_title")
self.label_title.setGeometry(QRect(0, 90, 661, 61))
font = QFont()
font.setFamily(u"Segoe UI")
font.setPointSize(40)
self.label_title.setFont(font)
self.label_title.setStyleSheet(u"color: rgb(85, 170, 255);")
self.label_title.setAlignment(Qt.AlignCenter)
self.label_description = QLabel(self.dropShadowFrame)
self.label_description.setObjectName(u"label_description")
self.label_description.setGeometry(QRect(0, 150, 661, 31))
font1 = QFont()
font1.setFamily(u"Segoe UI")
font1.setPointSize(14)
self.label_description.setFont(font1)
self.label_description.setStyleSheet(u"color: rgb(98, 114, 164);")
self.label_description.setAlignment(Qt.AlignCenter)
self.progressBar = QProgressBar(self.dropShadowFrame)
```

A.7 CODE FOR PRE-LOADER

```
class Ui_SplashScreen(object):
    def setupUi(self, SplashScreen):
        if not SplashScreen.setObjectName():
            SplashScreen.setObjectName(u"SplashScreen")
        SplashScreen.resize(680, 400)
        self.centralwidget = QWidget(SplashScreen)
        self.centralwidget.setObjectName(u"centralwidget")
        self.verticalLayout = QVBoxLayout(self.centralwidget)
        self.verticalLayout.setSpacing(0)
        self.verticalLayout.setObjectName(u"verticalLayout")
        self.verticalLayout.setContentsMargins(10, 10, 10, 10)
        self.dropShadowFrame = QFrame(self.centralwidget)
        self.dropShadowFrame.setObjectName(u"dropShadowFrame")
```

A.8 CODE FOR LOAD UI

```
def retranslateUi(self, SplashScreen):
    SplashScreen.setWindowTitle(QCoreApplication.translate("SplashScreen",
u"MainWindow", None))
    self.label_title.setText(QCoreApplication.translate("SplashScreen",
u"<html><head/><body><p><span style=\" font-size:26pt; font-weight:600;\">SOCIAL
SCRAPER</span></p></body></html>", None))
    self.label_description.setText(QCoreApplication.translate("SplashScreen",
u"<strong>APP</strong> DESCRIPTION", None))
    self.label_loading.setText(QCoreApplication.translate("SplashScreen",
u"loading...", None))
    self.label_credits.setText(QCoreApplication.translate("SplashScreen",
u"<html><head/><body><p><span style=\" font-size:9pt; font-weight:600;\">Created by
```

A.9 CODE TO SETUP UI

```
self.progressBar.setValue(24)
self.label_loading = QLabel(self.dropShadowFrame)
self.label_loading.setObjectName(u"label_loading")
self.label_loading.setGeometry(QRect(0, 320, 661, 21))
font2 = QFont()
font2.setFamily(u"Segoe UI")
font2.setPointSize(10)
self.label_loading.setFont(font2)
self.label_loading.setStyleSheet(u"color: rgb(98, 114, 164);")
self.label_loading.setAlignment(Qt.AlignCenter)
self.label_credits = QLabel(self.dropShadowFrame)
self.label_credits.setObjectName(u"label_credits")
self.label_credits.setGeometry(QRect(20, 350, 621, 21))
self.label_credits.setFont(font2)
self.label_credits.setStyleSheet(u"color: rgb(85, 170, 255);")
self.label_credits.setAlignment(Qt.AlignRight|Qt.AlignTrailing|Qt.AlignVCenter)

self.verticalLayout.addWidget(self.dropShadowFrame)

SplashScreen.setCentralWidget(self.centralwidget)

self.retranslateUi(SplashScreen)

QMetaObject.connectSlotsByName(SplashScreen)
# setupUi
```

A.10 APPLICATION FUNCTION

```
from main import *

class Functions(MainWindow):
    pass
```

A.11 UI FUNCTIONALITY

```
rom main import *

## ==> GLOBALS
GLOBAL_STATE = 0
GLOBAL_TITLE_BAR = True

## ==> COUT INITIAL MENU
count = 1

class UIFunctions(MainWindow):

    ## ==> GLOBALS
    GLOBAL_STATE = 0
    GLOBAL_TITLE_BAR = True

    #####
    ## START - GUI FUNCTIONS
    #####

    ## ==> MAXIMIZE/RESTORE
    #####
    def maximize_restore(self):
```

```

global GLOBAL_STATE
status = GLOBAL_STATE
if status == 0:
    self.showMaximized()
    GLOBAL_STATE = 1
    self.ui.horizontalLayout.setContentsMargins(0, 0, 0, 0)
    self.ui.btn_maximize_restore.setToolTip("Restore")
    self.ui.btn_maximize_restore.setIcon(QtGui.QIcon(u":/16x16/icons/16x16/cil-
window-restore.png"))
    self.ui.frame_top_btns.setStyleSheet("background-color: rgb(27, 29, 35)")
    self.ui.frame_size_grip.hide()
else:
    GLOBAL_STATE = 0
    self.showNormal()
    self.resize(self.width()+1, self.height()+1)
    self.ui.horizontalLayout.setContentsMargins(10, 10, 10, 10)
    self.ui.btn_maximize_restore.setToolTip("Maximize")
    self.ui.btn_maximize_restore.setIcon(QtGui.QIcon(u":/16x16/icons/16x16/cil-
window-maximize.png"))
    self.ui.frame_top_btns.setStyleSheet("background-color: rgba(27, 29, 35,
200)")
    self.ui.frame_size_grip.show()

## ==> RETURN STATUS
def returStatus():
    return GLOBAL_STATE

## ==> SET STATUS
def setStatus(status):
    global GLOBAL_STATE
    GLOBAL_STATE = status

## ==> ENABLE MAXIMUM SIZE
#####
def enableMaximumSize(self, width, height):
    if width != '' and height != '':
        self.setMaximumSize(QSize(width, height))
        self.ui.frame_size_grip.hide()
        self.ui.btn_maximize_restore.hide()

## ==> TOGGLE MENU
#####
def toggleMenu(self, maxWidth, enable):
    if enable:
        # GET WIDTH
        width = self.ui.frame_left_menu.width()
        maxExtend = maxWidth
        standard = 70

        # SET MAX WIDTH
        if width == 70:
            widthExtended = maxExtend
        else:
            widthExtended = standard

        # ANIMATION
        self.animation = QPropertyAnimation(self.ui.frame_left_menu,
b"minimumWidth")
        self.animation.setDuration(300)
        self.animation.setStartValue(width)
        self.animation.setEndValue(widthExtended)
        self.animation.setEasingCurve(QQtCore.QEasingCurve.InOutQuart)

```

OPEN SOURCE INTELLIGENCE (OSINT) GATHERING AND ANALYSIS TOOLKIT

ORIGINALITY REPORT

20%

SIMILARITY INDEX

15%

INTERNET SOURCES

4%

PUBLICATIONS

13%

STUDENT PAPERS

PRIMARY SOURCES

1

www.recordedfuture.com

Internet Source

2%

2

Submitted to Universiti Teknologi MARA

Student Paper

2%

3

www.coursehero.com

Internet Source

2%

4

www.sans.org

Internet Source

1%

5

btech.mit.asia

Internet Source

1%

6

predictiveanalytics.pnl.gov

Internet Source

1%

7

Submitted to The British School

Student Paper

1%

8

Submitted to LNM Institute of Information Technology

Student Paper

1%

content.sciendo.com