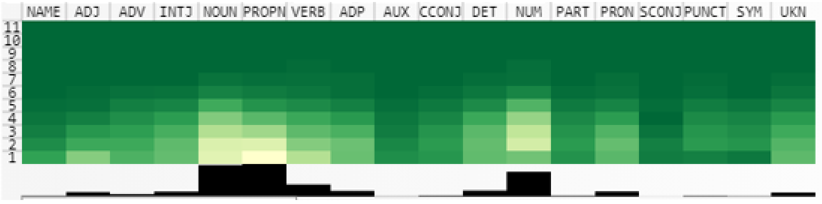
Interactive Visualisation of English and French Semantic in Passwords

CO880 : Dissertation

Author
Maël Brocher

Supervisor
Shujun Li



Total word count : 8300 words

School of Computing
Msc Networks and Security
University of Kent
Academic Year 2020 - 2021

UNIVERSITY OF KENT

ACCESS TO A MASTER'S DEGREE OR POSTGRADUATE DIPLOMA DISSERTATION

In accordance with the Regulations, I hereby confirm that I shall permit general access to my dissertation at the discretion of the University Librarian. I agree that copies of my dissertation may be made for Libraries and research workers on the understanding that no publication in any form is made of the contents without my permission.

Notes for Candidates

1 Where the examiners consider the dissertation to be of distinction standard, one copy will be deposited in the University Library.

2 The copy sent to the Library becomes the property of the University Library. the copyright in its contents remains with the candidate. A duplicated sheet is pasted into the front of every thesis or dissertation deposited in the Library. The wording on the sheet is:

> "I undertake not to use any matter contained in this thesis for publication in any form without the prior knowledge of the author."

Every reader of the dissertation must sign and date this sheet.

3 The University has the right to publish the title of the dissertation and the abstract and to authorise others to do so.

SIGNATURE **DATE**
05/09/2021

FULL NAMES
Maël Brocher

CERTIFICATE ON SUBMISSION OF DISSERTATION

I certify that:

1 I have read the University Degree Regulations under which this submission is made;

2 In so far as the dissertation involves any collaborative research, the extent of this collaboration has been clearly indicated; and that any material which has been previously presented and accepted for the award of an academic qualification at this University or elsewhere has also been clearly identified in the dissertation.

SIGNATURE **DATE**
05/09/2021

This form should be completed and submitted with your dissertation to the Course Administration Office Room S132 School of Computing

Declaration of Originality

I confirm with my signature that the submitted work is my own work and nothing has been previously submitted for any assessments. Furthermore, I have clearly identified and referenced any work done by authors other than myself using the Harvard referencing system. I agree that the University of Kent may use Turnitin to checks if any plagiarism has been made. I understand that any work considered as plagiarism will be punished.

signed :

date : 24/08/2021

Acknowledgments

I would like to thank my Supervisor and Professor Shujun Li, who guided me and devote a significant amount of his time to doing this project a success. Thanks to his comments, suggestions and views of the project, it permits to do my thesis an interesting and complex challenge to overcome.I also wish to thank my friends Louis, Nicolas and Octave who were able to give good advice and share their views about programming whenever I needed. And most importantly, I thank my family for their support and the good conditions they provided me to succeed.

Abstract

This report documents modifications of Gary Read work which made an interactive visualisation of passwords based on characters used in passwords. It was meant to detect patterns in passwords which couldn't be seen on raw data. I choose to improve his software with the approach of semantic in passwords in different languages. I decided to work with English and French as it's the two languages I am the most experimented with. Gary Read's Software is well built it has been relatively easy to reach my objective to build heatmaps of semantics in French and English. And because JavaScript is limited concerning data processing a Python server was set up to handle data parsing, detection of languages and part of speech tagging. All requests to the said server are asynchronous making the user free to do parallel analysis while the server processes the database of passwords.

Contents

List of Figures

List of Tables

Listings

1 Introduction

Today passwords are the most common way to authenticate to a service offered on the internet or on our personal devices and we know that this technology has a limit, the human. And sadly, people create their passwords based on their environment and persons they know in order to memorize them better. In 2012 a study published that password cracking software is between 300% and 600% more effective if it takes into account the personals information put on social networks by a user(al. 2016). It is becoming obvious that in the near future we will all have to use a password manager. That will permit to no longer leave the choice of the password to the human to keep it secret and random. The only breach for someone using a password manager will come from the master passwords or an error coming from the website or services he was authenticated to. A graphical representation of users passwords could make them aware of common behaviour when it comes to the creation of a password for a new website. Furthermore making a password graphical representation could be also helpful for the research community as it could let them experiment and observe various conduct of users through different database from all over the internet and the world. Moreover visual representation of data is a huge center of interest as, for example, the forum of Reddit ”r/dataisbeautiful” is counting more than 16 Millions users(*r/dataisbeautiful stats* 2021). Another major of this trend of data representation is the website informationisbeautiful.net which also has a very nice visualization about the 500 most used passwords and they could clearly separate them in 11 categories. (Figure 1).

1.1 Project Context

At first my thesis was supposed to be about passwords visualisation but it would have a smaller impact in the passwords security area. It would create a universe-like representation of passwords, sadly the student that was maintaining the project before couldn’t deliver in time and a new subject should be found for me to work on a thesis. Thankfully, my supervisor, Professor Shujun Li suggested me to work on the Heatmap and Parallel coordinates subject. I was very happy to work on this subject as I really enjoyed recreating the Pin Analysis Heatmap (Berry 2012) during cyber Security Class. It was started by Gary Read in 2016. It is a powerful tool that create heatmaps based on characters and their positions in passwords. These 2D heatmaps permit to see patterns that couldn’t be seen in raw data or with pure statistics. For example with upper cases, we can clearly see how they are used in passwords. In general, it’s at the start of the password because that’s how people write proper nouns. In 2018 Stephanie Schmid has taken over the work to add a Regular Expression dimension to the tool. This approach can be very useful when looking for a specific pattern in passwords. After some testing it was clear that using Stephanie

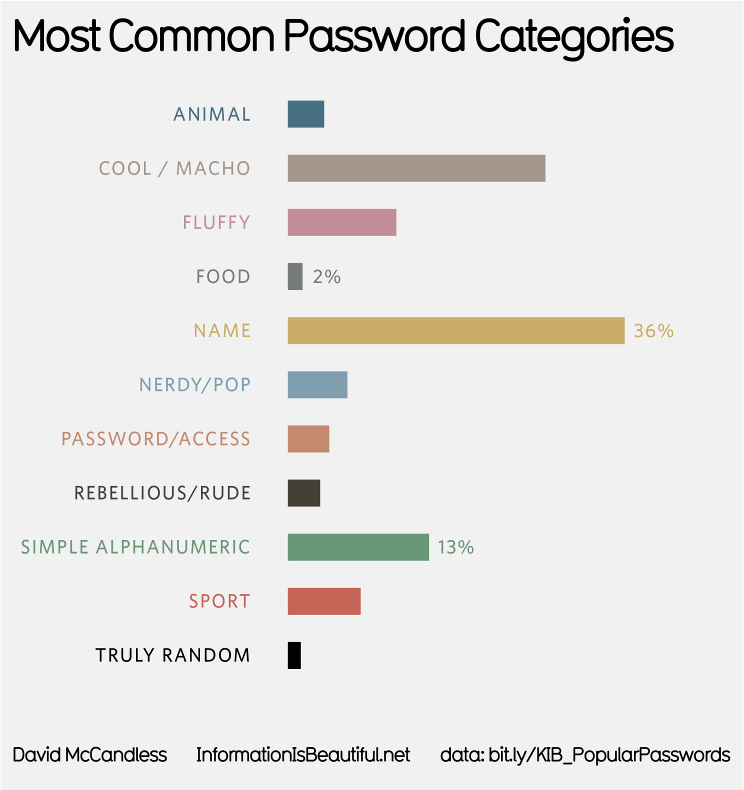


Figure 1: Categories found by David McCandless from informationisbeautiful.net

Schmid's code was not the best idea as many of the usability issues had been introduced and it would have slowed me down to understand her base code and Gary Read's base code. This section is also made to explain my objectives and the approach dedicated to improvements of Gary Read project.

1.2 Project Objective

This project has many objectives as it is possible to improve and add interesting functionalities. The main objectives here are to make the tool more oriented around the semantic in passwords.

- The first thing to keep in mind is to **add functionality while keeping Gary's work functional**. It would be embarrassing to re-do the problem encountered with Stephanie Schmid work and throw my work away because of it instability.
- Further more, I have to **review what has already been made** in the field of semantic visualisation of passwords to not make anything already existing and also creating something as relevant as possible. Hopefully I don't have to look for another visualisation technique as

Gary's work is already providing me an intuitive and well made program.

- Moreover, **implementation of new functionalities around semantic** and finding the best way to segment and process the data the most accurately possible will be necessary.
- And finally, some **test and evaluating** of the programs created would be necessary to ensure the quality of the project. It would permit to check if the requirements stated are met or not to measure the success of the project

1.3 Project Approach

The project starts with a lot of article literature readings as I am not familiar at all with the field of passwords, semantic analysis and human conduct about passwords despite being a really interesting but complex area of work. Doing this and testing Gary's work permits me to understand basic bias present during the process of the password creation. Moreover having Professor Shujun Li as my supervisor was very helpful as he is a great researcher concerning cyber security, it will permit me to have the point of view of a potential future user of my work. His participation makes my thesis run smoothly with good criticisms and suggestions to create the best version of the project. Thanks to his views of the subject and scientific literature, requirements needed could be identified to succeed in this project. Thereafter, functionalities will be gradually implemented while following advice received through meetings. Once most of the software is finished, tests will be run to check if the requirements previously defined are met. Finally, a review of the project will be made to get the full picture and concludes with possible additions.

2 Literature review

In order to have a global view on the progress of the research on password analysis and mostly password visualisation and semantic about passwords, a literature review was necessary. Thanks to the knowledge collected about human conduct, passwords visualisation and semantic in passwords, I have a better comprehension of requirements and objectives. It will set a good environment to conduct a successful project.

2.1 Human behavior

This section reviews literature about the human conduct when it comes to create a password. Nowadays the research is focused on this bias made by humans creating and using their passwords and why some patterns are so much widespread. A better understanding of this bias could significantly improve password cracking software.

A study made in 2016 was asking to people to create passwords following specific format guidelines in order to comprehend the common human thought process (Shay et al. 2016). They consider four types of characters someone can use in English in passwords: Lowercase, Uppercase, Digits and symbols. The study asks to create passwords and compare the result of combinations of theses type of characters with a wide range of length, but it should not be a word present in classic dictionary. They also requested passwords composed at of least two words to encourage passphrase. It appears that the strongest passwords were those made of non-existent words, long and composed of three types of characters. In general, subjects have a hard time creating passwords based on these complex criterias and it could go up to 2.4 attempts in average to create one. But the important part of this study, for this project is the usability of those passwords, and it has been shown that longer passwords are perceived harder to create and remember. People still manage to put easy words and subsets like ”1234”, ”passwords” or even a year. It shows that the context of study doesn't remove these classic errors.

Another study conducted in 2017 with 154 participants and 4057 passwords shows that the average subject reused totally or partially 79% of passwords (Pearman et al. 2017). They also demonstrated that there is a direct link between the reuse of passwords partial or complete and the number of domains visited daily. The participants have clearly different strategies of passwords, from people that enter passwords in websites less than one per month to people that have more than nine entries a day. I really like this study because they compared data from queries of users estimating their relation to passwords and data when they observed participants and their usage of passwords. It is clear that most of the people underestimate the time they spent on internet, as the data from active days (when participants are observed) are always superior to their idea of their daily password usage. The only downside of this study is the sample, it's composed of 60.4% females but it is counteracted by the large number of passwords given and the variety of domains and the difference in education levels of participants.

2.2 Password Visualisation

This section reviews literature about passwords visualisation, but it's mostly dedicated to passwords visualisation made after 2016 as Gary did an incredible job and any new way of visualisation is welcome.

In this study (Yu and Liao 2019), an interesting visualisation is made with links between prefixes, word and postfix. However, it makes the result difficult to see and there isn't the dimension of frequency like an heatmap does. But heatmaps made in this paper concerning number after city and people name give this time a good-looking result and we can see that it is mostly keyboard patterns oriented, we can find number such as *123*, *1010*, *1020*, *1212* or *919*.

This paper is mainly focused on gesture and path the hand could make on a keyboard to type a password (Schweitzer et al. 2011). It has the advantage to show how passwords that looks complicated to remember are simply a shape on the keyboard. For example the passwords *6TFCVBNHY6* is a triangle in the middle of the keyboard. This could be a great use for my thesis as the work will consist of comparing French and English language and both language have different keyboard layout, respectively AZERTY and QWERTY. In this study they could identify 11 patterns clearly cause by keyboard logic. However, the paper takes in account as keyboard pattern when two keys are next to each other. This one isn't really relevant as it quite common to find words that have two contiguous keys. Combination of *'ed'* is common in English, *'oi'* is common French and *'er'* is common in both. But it could be interesting to implement this combined to a dynamic heatmap on a keyboard with an animation and a slide bar to progress through characters positions.

2.3 Semantic in Password

The group of researchers Veras, Collins and Thorpe did and incredible work via their two papers based on the semantic dimension of passwords.

Back in 2014 they realised a study focused on the security impact of the semantic in passwords, and they already demonstrated that more than 73% passwords are composed of a noun segment (Veras, Collins, and Thorpe 2014). They also made a list of the most common combination of segments in passwords. The top 5 most used segments are mixs of numbers, male names and female names.

In the study published in April 2021, they could create a list of most frequent semantic group and the first one display is the 46th most probable and his probability is incredibly low $1.9x10^{-3}$ (Veras, Collins, and Thorpe 2021) . It means that my semantic heatmap cannot be made of semantic groups and a POS (Part of Speech) group could be more adapted to my situation. Moreover they did a great visualisation of dependencies between high-level grammatical classes, this kind of spiderweb graph is great to demonstrate links between common segments in passwords. In this study they aimed to make an augmented version of the PCFG, a program made to perform guessing attacks on a database of passwords. With the semantic approach they hoped to make something better than the latest neural network models (Melicher et al. 2016). Their semantic PCFG (Probabilistic Context-Free Grammar) is very situational as it gives different results, worse than Melicher for LinkedIn and better one for 000webhost data leak.

3 Problem analysis and Requirements

This section is mostly about the requirements inherited from Gary Read and the new ones coming with the modification made. The tools are highly focused to be interactive and intuitive in order to get started quickly for the user. Functional and non-functional requirements are listed, described and the thought process is explained. Some usability requirements, therefore, follow to keep the whole coherent. Based on what we learn during the literature review we can deduce what the user wants and what is best to display a good result for the user. Again, thanks to the feedback from my supervisor Shujun Li, a better view of the users needs could be defined. During our meetings, we could discuss about the features, and we were able to create something between his expectations and the technical limitations. Due to these constant adjustments, some features might have been drop during the process.

3.1 Functional Requirements

Here are described functional requirements needed to successfully create features according to needs extracted from conversation with Shujun Li and my literature review. Each requirement is given a priority in order for me to have a better management of my time during the implementations of features. Functional requirements are explained in the Table 1.

No.	Description	Priority
1	keep previous Gary Read code working	Must Have
2	Semantic Heatmap	Must Have
3	Analysis in different languages	Must Have
4	Set analysis	Must Have
5	Heatmaps are interactive and responsive.	Must Have
6	the program supports multiple file upload.	Should have
7	The program notifies the user during long processes.	Should have
8	The program allows for the comparison of visualisations.	Should have
9	Fix previous bug	could have

Table 1: Functional requirements

The idea of the keyboard heatmap have to be drop as it is out scope and maybe will took to much time to implement it.

3.2 Non-Functional Requirements

This subsection is about requirements concerning the conduct of the software, how the features will be implemented and how they will be designed. Non-Functional requirements are explained in the Table 2, still ordered from the most important to the least one, to keep a good work rhythm.

No.	Description	Type
1	New features should be intuitive	Usability
2	work on different Operating System	Compatibility
3	work on various browser	Compatibility
4	Queries process should be short or asynchronous if long	Usability

Table 2: Non-Functional requirements

3.3 Usability

Usability is a term well defined through *Usability Inspection Methods* from Jakob Nielsen(Nielsen 1994). Based on his few principles, it will help to create a good interface. The part that interests us is the following : Heuristic evaluation, where it guides us to other works he previously did in the most informal way possible. These requirements where previously specified in Gary work (Read 2016). To put it in a nutshell, it explains how the software has to be user friendly. Firstly, by making it clear what is happening after a user action. Then, be comprehensible for everyone, describe every specific vocabulary if necessary. Moreover, the software will need options to go back if he made a mistake. And if the mistake come from the software, he will need to be alerted to avoid confusion. Finally, the interface has to stay as simple as possible, we don't want to lose users attention looking at something else other than the core functionalities of the software.

4 Methodology

This part focuses on the methods used to implements correctly features based on the requirements listed before. Through weeks of meetings with my supervisor, it became natural to set up a rapid prototyping method. Each week we discuss about new functionalities created past week and how to improve it. Some new features are suggested, and I try my best to implement them for the next week and then we repeat the process the next week. Sometimes those suggestions cannot be made as the program is sometime limited by the computational speed of my computer or more simply the programming language I'm using isn't made for the feature suggested.

4.1 Concepts

4.1.1 Set Management

One of the first objective of the project is to make more set operations. Gary Read's version of work only permit to merge two databases and make one containing words of both databases. In order to call this feature of merging databases, the user has to drag and drop on the button. Shujun suggested that set difference and set intersection could be interesting to see. The Figure 2 is a scheme of what the user interface would look like when set operations will be added in the program, two more button to drag and drop one database into another. For the set difference it would create an Heatmap containing words from the database A and database B and as a result we would have words of:

$$A - B$$

And for the set intersection a new heatmap containing words from the database A and B and it will contains words from:

$$A \cap B$$

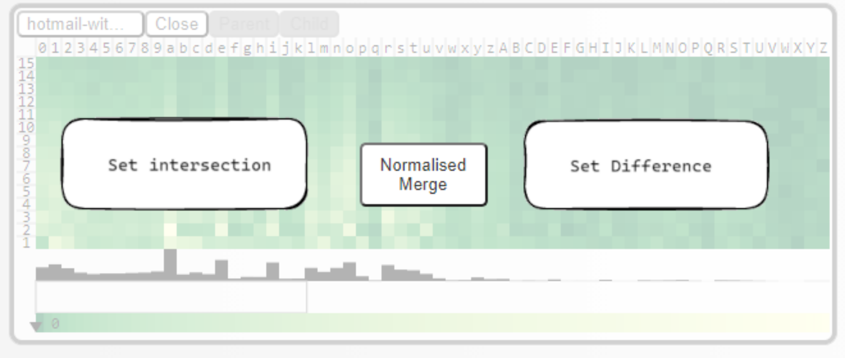
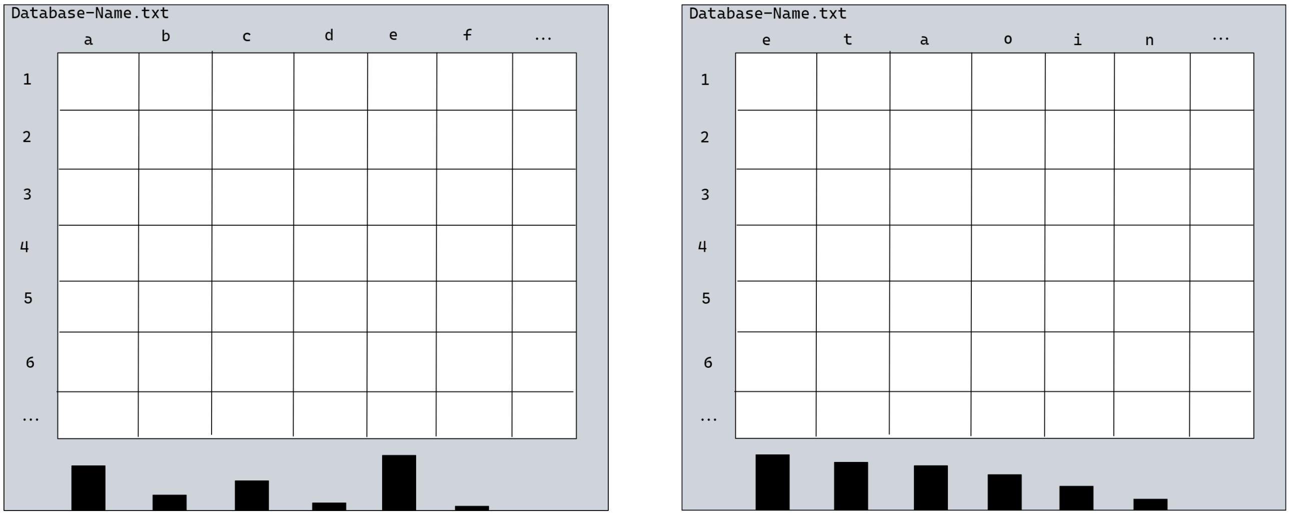

hotmail-wit...
Close
Parent
Child
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
Set intersection
Normalised Merge
Set Difference
0

Figure 2: User interface

4.1.2 Axis Management

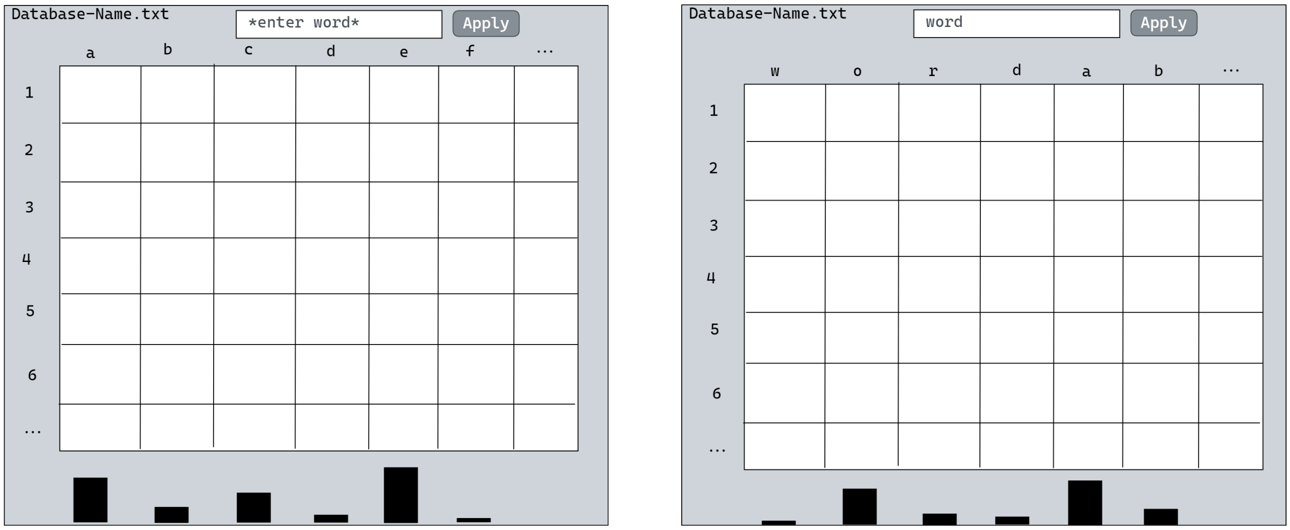
During the test phase of Gary Read project, some functionalities were not working and the Axis Management was part of it. A feature in the options was meant to change the X axis order from alphabetical order to frequency order, and it was disabled. Even if it was force enabled, nothing in the algorithm was programmed to sort characters of the Character set by frequency. Adding a specific section in the parameters will permit the user to change the order of characters with these two options, alphabetic and frequency giving respectively the result of Figure 3 figures (a) and (b). Also, another possibility will be to sort the x Axis with a text input. It will permit to the user to play with differents words and find some patterns based on his custom Character set. It will be implemented directly on the heatmap or in the parameters section the input could be added. It could look like Figure 4. In the case where character is present two or more only the first iterations of each duplicate will be saved in the Character set.



(a) Alphabetical Order

(b) Frequency Order

Figure 3: Different ordering options possible



(a) Base conduct of custom charset

(b) Word ”word” use to change the charset

Figure 4: Custom X axis with text input

4.1.3 Semantic Heatmap concept

For this part of the project, the same architecture used for normal heatmaps based on characters will be used for the semantics one, it will give as a result the Figure 5.

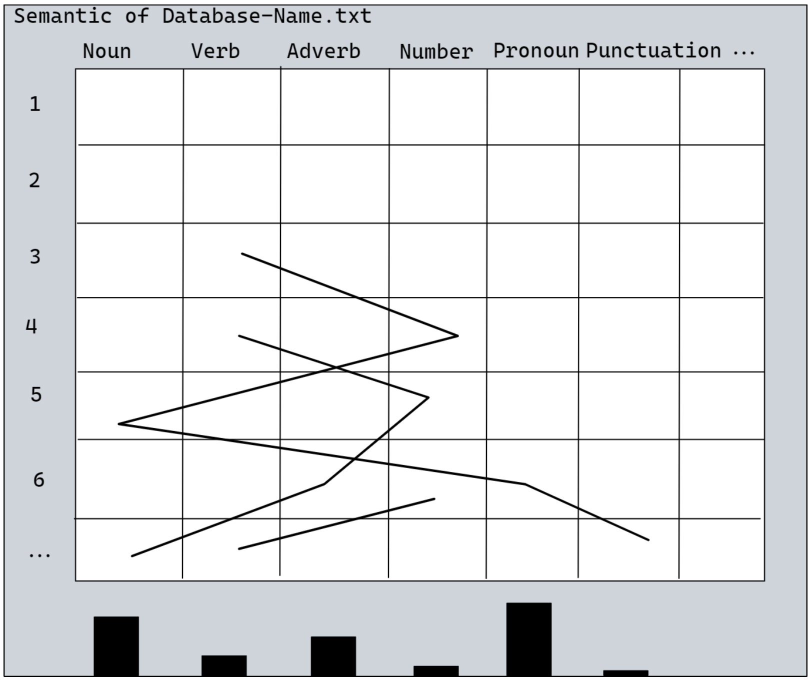


Figure 5: Concept of semantic heatmap

As Gary Read did most of the work to create them, my part here is to ensure that most of the features of normals heatmaps are done for the semantic heatmaps. All the data of each heatmap uploaded have to be processed before and then create a semantic copy, or an option to switch from normal to semantic will be needed. As seen with the paper about security of semantic, semantic tag of a word is too specific is too precise and POS (Part of Speech) will be used instead. In this paper when a semantic heatmap will be mentionned, it's in reality a POS heatmap as we stated it would be not effective. The parallel coordinates options have their importance as it shows to users the most used combinations of POS groups. It is important that all the variables created for characters heatmaps have their semantics version to permit the switch between semantic and characters modes. Furthermore, semantic analysis will be performed in various languages, and the language wanted for the analysis will be added in the name of the heatmap.

4.2 Overview of the architechture

During the project the concept was quickly evolving due to the rapid prototyping method. Towards the middle of the development, the Python Flask server had to be setted up as it wasn't necessary before. Javascript is limited in terms of external library for language analysis and a lot of modules Shujun advices me were in Python. In the end, the flask server will process queries processing data from files, determine language and split passwords in a list of words and detect part of speech these chunks. It became logic to deploy the HTML and Javascript file with the server to make queries

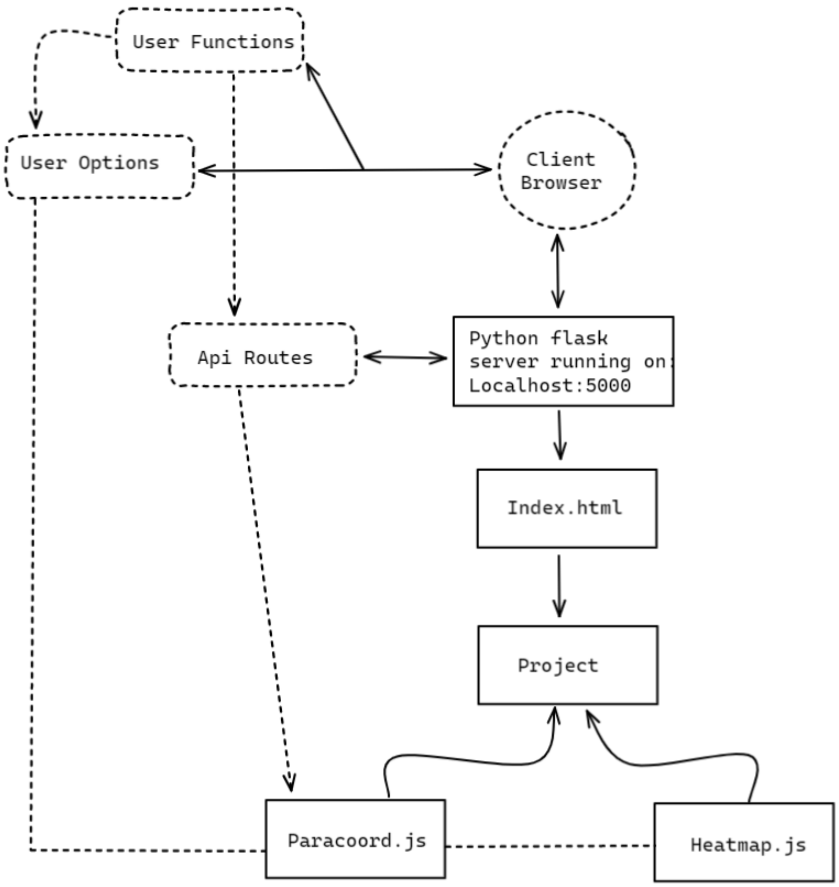


Figure 6: Architechture overview

faster. The last version of my project has an architecture that looks like the Figure 6. Most of Gary Read's works didn't change, whenever a file is uploaded the Heatmap class handle the data and the Paracoord is here to act on the visualisation of the heatmap. Moreover, when a file is uploaded, it is transmitted to the server and start to process it asynchronously. New features added in the Paracoord class are calling the Python server to get analysis result or even extract a great variety of subsets. Some elements and variables have been duplicated in the Heatmap class to enable the operations of a semantic view of the heatmap. The project part of the Figure 6 is functions present in both class that permit the communication between these two.

5 Implementation

This section describes the implementation process and how every new functionality has a place in the program of interactive visualizer of Gary Read. Firstly about the environment used and what kind of library made the project successful and then new functionalities will be explained, how they work and why it's this way.

5.1 Development Environment and Tools

The way Gary Read made the project permit to be usable on every platforms possible as it is made in HTML/CSS combined with Javascript. A web based program have this advantage but have some downsides as browsers have a maximum limit concerning the maximum RAM a tab can use, my test were run with Chrome and the limit is set to 50MB but for most of the time it is enough as the major part of databases used aren't bigger than 5MB. Moreover, during the development the project needed some Python to run on a local server, a language heavily supported and updated which wont affect the compatibility nor the maintainability of the project. Furthermore, to develop the project Visual Code Studio was mainly used to write code of Python, HTML/CSS and Javascript.It has the advantage to adapt himself in function of languages used thanks to extensions. Also, the Google Chrome's developer console was used to debug network exchanges and make adjustments for the visual sides of the web page.

5.2 External Library

Here is a full list of components, languages, libraries used for the project:

- Javascript
- HTML5
- CSS3
- SVG
- d3.js version 4.2.2
- date.js
- python3
- C++ 2017
- spacy version 3.1.1
- fasttext version 0.9.2
- flask version 2.0.1
- Selenium version 3.141.0

The first six elements were mandatory in the Gary Read version and they were kept them as it was unnecessary to find something else. Python3 is needed to run the local server and Spacy, Fasttext, Flask and Selenium are python modules. C++ is only needed because of python using it by calling methods to make Fasttext works. Spacy and Fasttext are made to process rapidly passwords, Flask is the module charged to deploy locally the server. Selenium is a library used to scrap web page, it has been really helpful when data from Wikipedia not available on Wiktionary were needed.

5.3 Difference between two versions

Gary Read project serves me as base code, followings subsections describes the implementation of news components and can be used as a documentation of these functionalities.

5.3.1 Functionality from Stephanie Schmid work and minor UI changes

Despite having some bugs making the project unstable, Stephanie Schmid did implement a lot of interesting that can be easily imported without causing any harms to my actual project. The "Copy" shown on Figure 7 functionality was really useful as it permits the user to actually have a duplicate of the heatmap he is currently working on. It is great to see differences between different sub-sets. Moreover, Stephanie did two button in the parameters sections (Figure 8) that permit the user to increase and decrease both cell width and height at the same time. It could be a nice feature to implement, and be useful if the user wants to zoom in or zoom out on heatmaps selected. Furthermore, in her version of the project, when a file is uploaded, the corresponding heatmap is considered as selected, which avoid some confusion for the user if he tries to interact with the heatmap and nothing really happens.

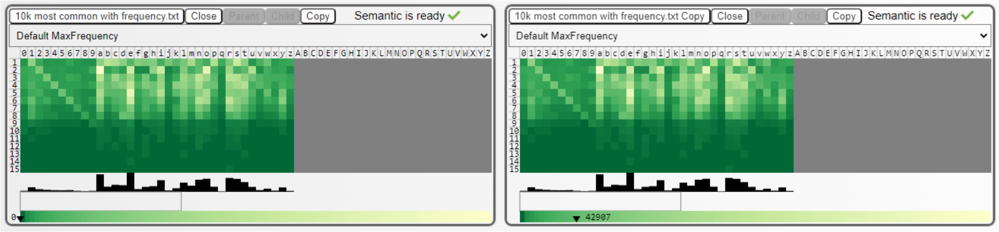
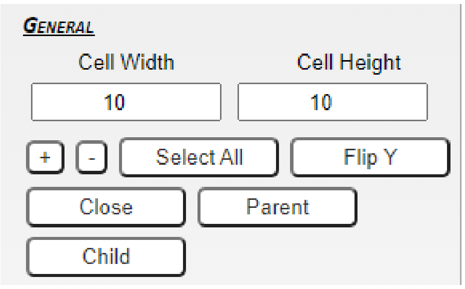

10k most common with frequency.txt
Close
Parent
Child
Copy
Semantic is ready ✓
Default MaxFrequency
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0
10k most common with frequency.txt Copy
Close
Parent
Child
Copy
Semantic is ready ✓
Default MaxFrequency
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
42907

Figure 7: Example of the Copy of a heatmap



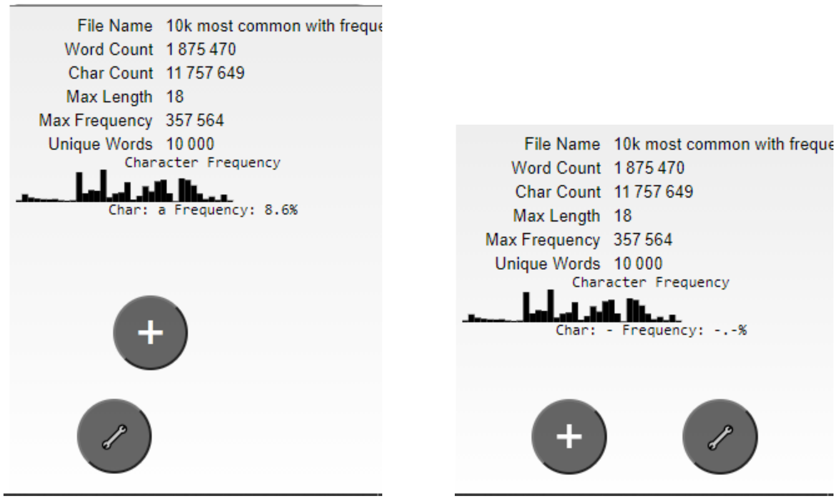
GENERAL

Cell Width: 10

Cell Height: 10

Figure 8: General section of parameters

There were also some features that wasn't very clear like the drag and drop functionality on the upload and option buttons. If the user tries to interact with the button and he moves even a little his mouse the button will not performs his actions and move the button clicked. This feature was removed to make it works all the time Figure 9. Also as more sections in the parameter container will be added so the height ratio between the parameter containers and the statistics container change from 60% and 40% to 70% and 30% of the size of the screen.



File Name 10k most common with freque

Word Count 1 875 470

Char Count 11 757 649

Max Length 18

Max Frequency 357 564

Unique Words 10 000

(a) Before the fix

File Name 10k most common with freque

Word Count 1 875 470

Char Count 11 757 649

Max Length 18

Max Frequency 357 564

Unique Words 10 000

(b) After the fix

Figure 9: Remove of the drag and drop feature

5.3.2 Set Differences and Intersection

One of the first functionality implemented was everything concerning the set management. Gary Read already made something to merge two heatmaps and it is from this point that I have based myself to continue the management of the sets. When the user drags and drops an heatmap into another, a button appears over the targeted heatmap. Buttons for set difference and set intersection are added next to the normalised merge buttons Figure 10. To set this up it was fairly easy as the set manipulation only needed to extract words from both dragged and target heatmap and compare them. It would keep common passwords for the set intersection as shown in listing 1 and remove passwords from heatmap B to heatmap A for the set difference shown in listing 2. And the outcome would look like Figure 11 for both set differences, there is two different database possible for result, A-B and B-A which explains the differences of the database 3 and 4. Moreover, the names of these new heatmaps are of the format "operation + Heatmap Name 1 + Heatmap Name 2" and we could have as an example "Difference CSDN_passwords_frequencies_greater_than_100.txt faithwriters-withcount.txt"

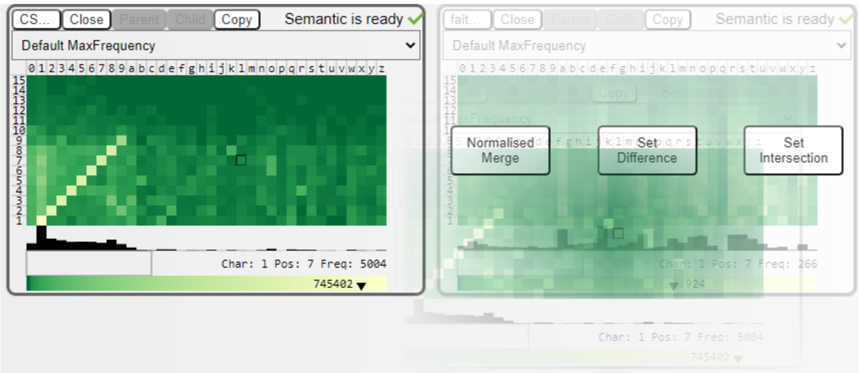


Figure 10: Drag and drop interaction showing new buttons

```
var intersection = new Set([...heatmapdragged].filter(x =>
    targetHeatmap.has(x)));
```

Listing 1: Javascript code that extract common passwords between two data bases

```
var difference = new Set([...heatmapdragged].filter(x => !
    targetHeatmap.has(x)));
```

Listing 2: Javascript code that remove passwords of database dragged from the target heatmap

5.3.3 Axis Management

For this functionality I had to understand more deeply how the heatmap was construct, before that I was simply manipulating words from different heatmaps. When a file is uploaded, the heatmap is drawn column per column and each column is made for every element of the character set. If the character is a string, it will be one character per column, but the management of the character set works in such a way that it is possible to make an array of string work and each column will have a string assigned. Furthermore, some new characters set were added to bring new points of views during heatmaps analysis. New character set are mainly made to compare

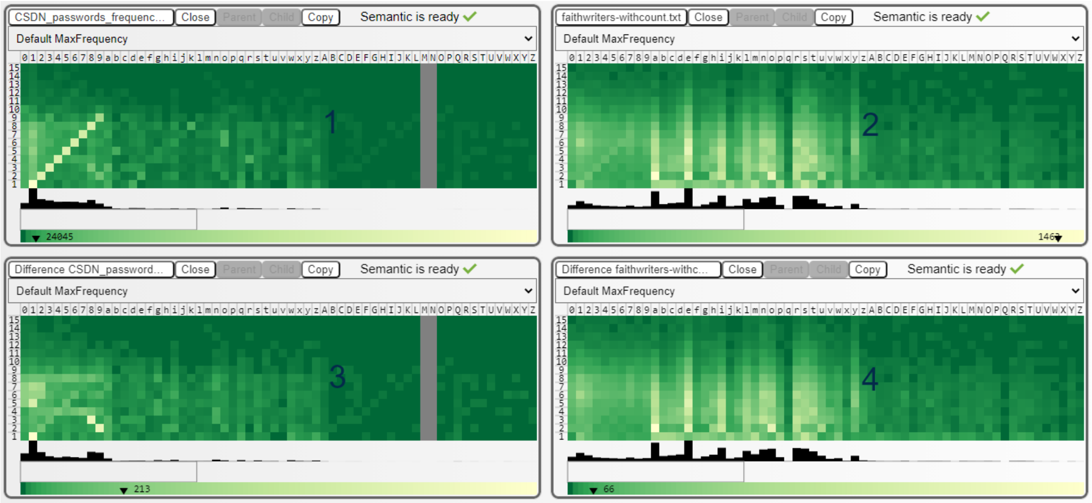


Figure 11: Heatmap 3 is the set difference of 1-2 & heatmap 4 is the set difference of 2-1

lowercase and upper case and also vowels and consonants. To implement them the defaultConfig.js file just needed to be edited and add these news character set like listing 3.

```
"charSet"                : {
    "lowAlpha"          : "abcdefghijklmnopqrstuvwxyz",
    "highAlhpa"         : "ABCDEFGHIJKLMNOPQRSTUVWXYZ",
    "alpha"             : "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ",
    ...
    "lowHighCompare": "AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz",
    "vowelsConsonantsalpha" : "aeiouybcdfghjklmnpqrstvwxzAEIOUYBCDFGHJKLMNPQRSTVWXZ",
    "vowelsConsonantsLow" : "aeiouybcdfghjklmnpqrstvwxz",
    "vowelsConsonantsHigh" : "AEIOUYBCDFGHJKLMNPQRSTVWXZ",
}
```

Listing 3: Updated charset config in defaultConfig.js file

```
var data = [];
for (var i = 0; i < charSet.length; i++) {
    var c = charSet[i];
    var cfreq = (state.charSet[c] == undefined) ? 0 : state.charSet[c];
    data.push({ value: cfreq, name: c });
}
data.sort(function (a, b) { return b.value - a.value; });
for (i = 0; i < charSet.length; i++)
    newCharSet += data[i].name;
```

Listing 4: Updated charset config in defaultConfig.js file

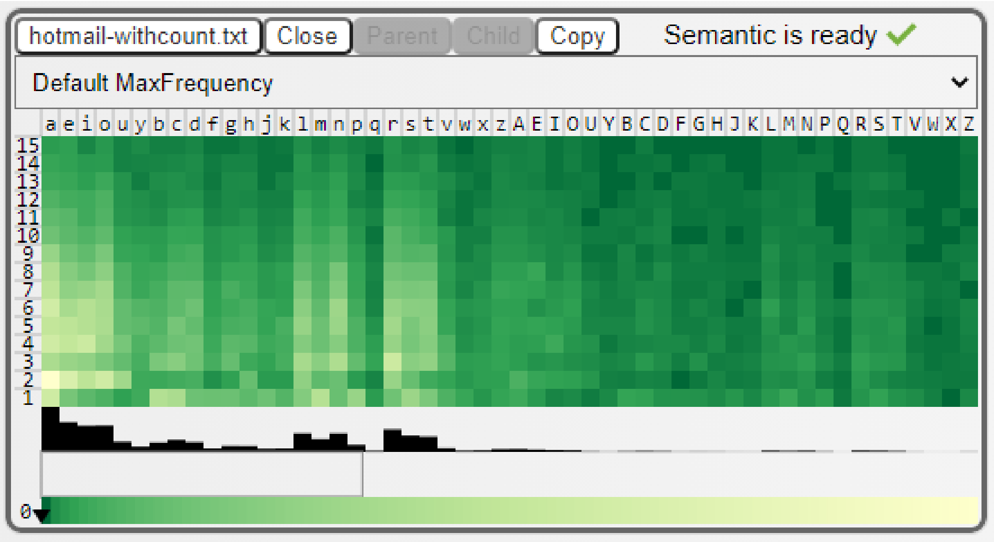

hotmail-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
Default MaxFrequency
a e i o u y b c d f g h j k l m n p q r s t v w x z A E I O U Y B C D F G H J K L M N P Q R S T V W X Z
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0

Figure 12: Example of charset vowelsConsonantsalpha on the hotmail dataleak

The Figure 12 is showing the effect of the character set vowelsConsonantsalpha which highlights the use of vowels in passwords. Moreover an Axis management section shown on Figure 13 has been created in the parameter container for some advanced manipulation of the X axis. In Gary Read's version the options that would have permit to switch from alphabetical order to frequency order was created but the code actually sorting elements in the character set was never made. Here whenever the user switch from alphabetical to frequency the actual character set is reordered thanks to this code in listing 4. For each element in the actual charset, the sum of frequency of all positions is added in an array, and the array is sort with the JavaScript system function, a new charset is then recreated. It gives as a result the Figure 14. Finally the Re Order axis button and the input next to it as shown in the Figure 13 is made to add at the start of the X axis some character a user might want to study, it will permit them to create some custom X Axis. Also, it will delete every duplicate character, for example if my character set is "abcdefghijklmnopqrstuvwxyz" and the input words is "cybersecurity" the new character set will be "cybersuitadfghjklmnopqvwxz". It is also possible to add character not present in the character set, here is the visual result of the reorder feature in Figure 15.

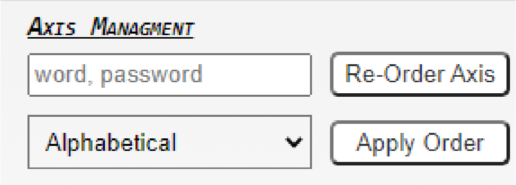
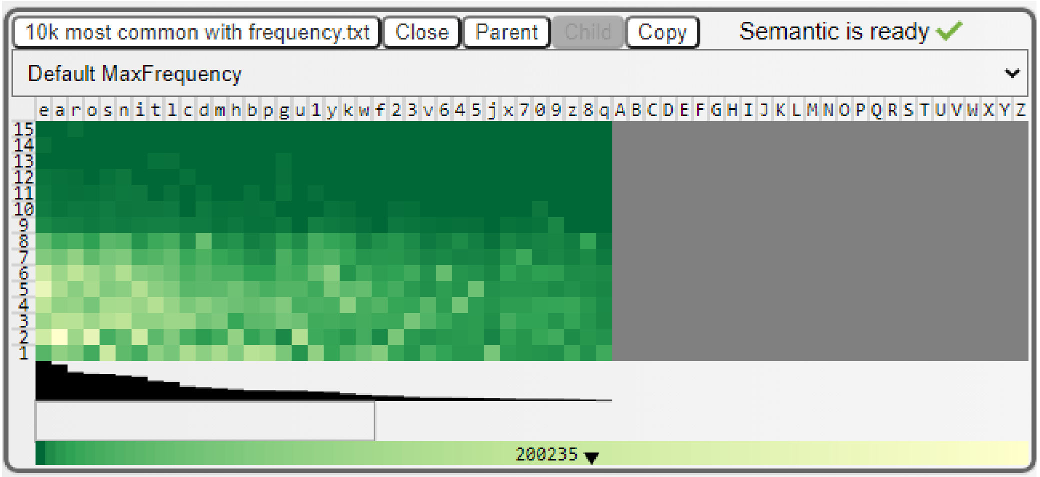
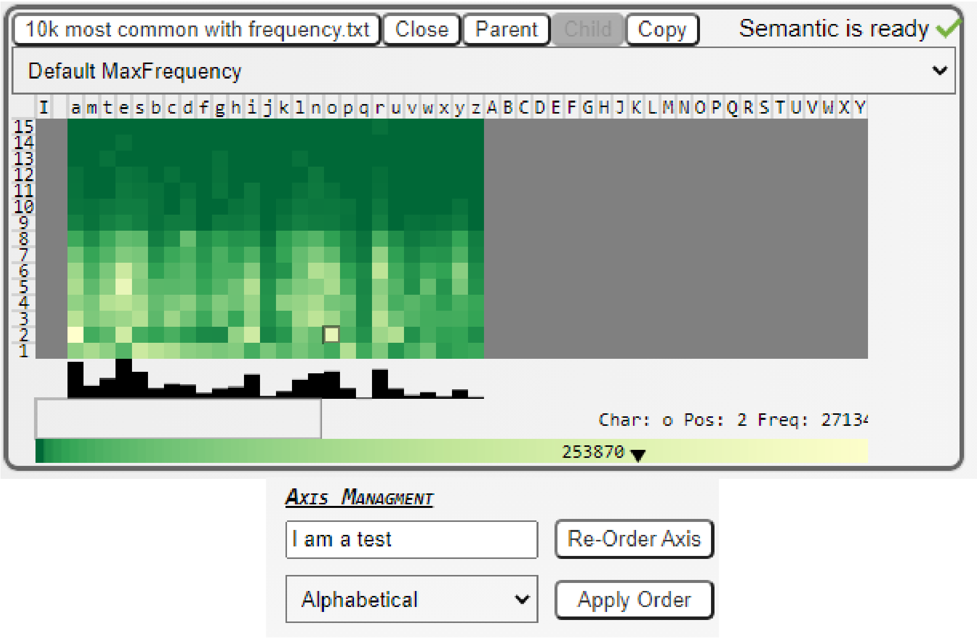


Figure 13: New section in parameter concerning X Axis Management


10k most common with frequency.txt
Close
Parent
Child
Copy
Semantic is ready ✓
e a r o s n i t l c d m h b p g u 1 y k w f 2 3 v 6 4 5 j x 7 0 9 z 8 q A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
200235

Default MaxFrequency

Figure 14: 10k most common passwords database ordered by frequency


10k most common with frequency.txt
Close
Parent
Child
Copy
Semantic is ready ✓
I a m t e s b c d f g h i j k l n o p q r u v w x y z A B C D E F G H J K L M N O P Q R S T U V W X Y
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
253870
Axis Managment
Re-Order Axis
Apply Order

Default MaxFrequency

Char: o Pos: 2 Freq: 2713

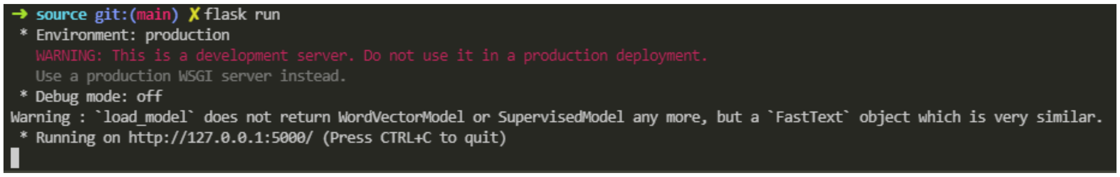
I am a test

Alphabetical

Figure 15: 10k most common passwords database with ”I am a test” as word input

5.3.4 Flask server

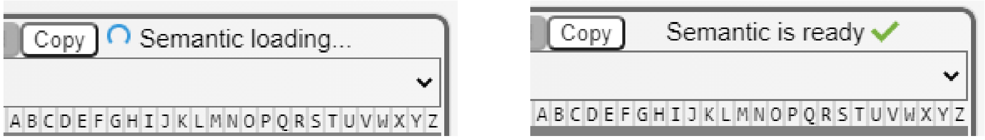
For further analysis a server processing the data is becoming mandatory as JavaScript don’t have fast processing library meant to analyse words in different language, but Python does. To permit JavaScript and Python to communicate, options are limited, a server is needed and Python will host it while JavaScript will send AJAX calls to transmit data to the server. In order to deploy a server easily, the Python module Flask is used. Also to lighten up the program and the server processing request the web page is deployed at the address **http://localhost:5000/**. If all the dependencies are installed to run the server it is only needed to type ”flask run” in a linux console and ”python3 run.py” for the windows powershell and wait to have your console to look like Figure 16 to call anything from the server.



```
→ source git:(main) ✗ flask run
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
Warning : `load_model` does not return WordVectorModel or SupervisedModel any more, but a `FastText` object which is very similar.
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Figure 16: Console ready to receive request

5.3.5 Semantic Heatmap


Copy
Semantic loading...
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Copy
Semantic is ready ✓
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

(a) Semantic Animation loading

(b) Semantic ready

Figure 17: Quick animation to inform the user

In order to get the semantic heatmap a lot need to be done, firstly in JavaScript every variable made to store data about the heatmap have to be duplicated to store semantic heatmap data as column will be displayed with POS tag (Part of Speech) as character set. Whenever a file is uploaded to the webpage, words and their frequency are sent to the server via an AJAX call to process them, when the data is ready the server will send back a confirmation request with the name of the heatmap uploaded. The user is notified directly on the heatmap if the semantic analysis is ready or not as shown on Figure 17. When the server receives the data, the first thing it does is to detect the lang used in the passwords thanks to the module fasttext that is an offline language recognition algorithm based on trained data loaded at the server start. The language of spacy and wordninja is set according to the language detected by fasttext. Before splitting the passwords in segments one last check is made to be sure that the passwords isn't a French nor a English first name. Data were found respectively on the **French government website** and on github for the English ones at **https://github.com/smashew/NameDatabases/**. To set up the module that will split the words some modification needed to be done as wordninja is encoded in utf-8 instead of latin1 that support every possible French character. Moreover a Regular Expression is present in wordninja that prevent any special characters that could appears in french such as à, é, è, ù or ê, so it has been reworked to accept them. One of the

Position	English	French	English translation
1	the	je	I
2	of	de	of
3	and	est	is
4	to	pas	not
5	in	le	the
6	I	vous	you

Table 3: Most common words in French and English languages according to wikipedia

reasons why wordninja was chosen is because it is possible to change the dictionary on which it bases its word cuts, which is perfect for us has we will switch quickly between two languages. To avoid closing and re-opening files quickly it as been decided that two version of wordninja will be created, one managing French language and the other English language. Wordninja dictionaries are a bit special, to have a faster program the dictionary is ordered by frequency and according to **wikipedia** in Table 3 those are the most common words.

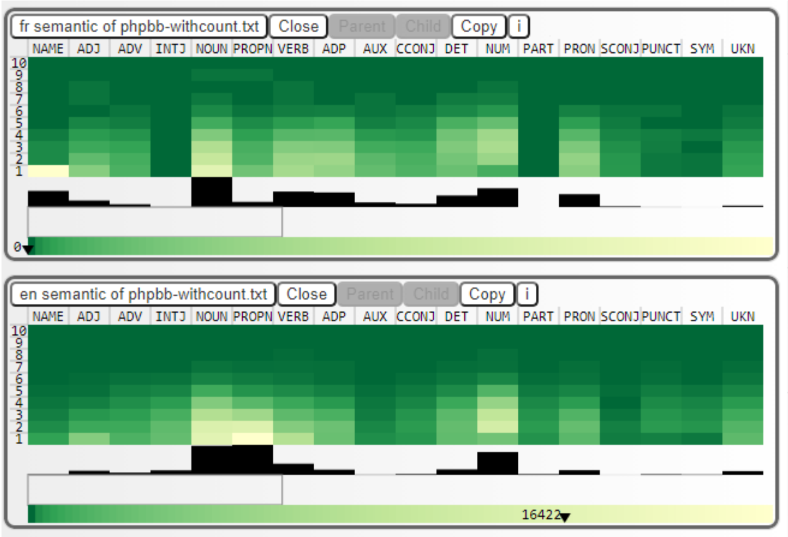


Figure 18: Semantic analysis in English then French made on phpbb dataleak

A small script made in python with selenium permits to scrap first 40 000 most used words in English and 20 000 for French languages. Then words are stored in a compressed file ready to be used by wordninja. Right after splitting the passwords, the program makes sure that the split went well, if more than four chunks of one character exists in a word it is considered as invalid and skipped. If it isn't the case, that is where the POS tagging made by Spacy happen with a check if a segment discovered isn't a name. And finally passwords composed of digits only are rejected as no language can be determined. All the passwords will be processed this way and the results is saved on the server, if at any times a database already processed is uploaded the confirmation message will be instantly sent. Now that the database is ready, the user can call semantic data in the language wanted and we see Figure 18 as a result. Columns have been expanded to fit POS tag label and every feature such as histogram, parallel coordinates (Figure 19) or even X axis management (Figure 20) are functional. For files bigger than 2MB it can take more than 2 minutes with a AMD Ryzen 5 3600 6-core as a CPU and a NVIDIA Geforce RTX 2060 for the GPU. Otherwise for files from 100KB to 500KB the process time should vary from 7 seconds to 30 seconds. A section made for semantic is added to performs AJAX calls as shown on Figure 21.

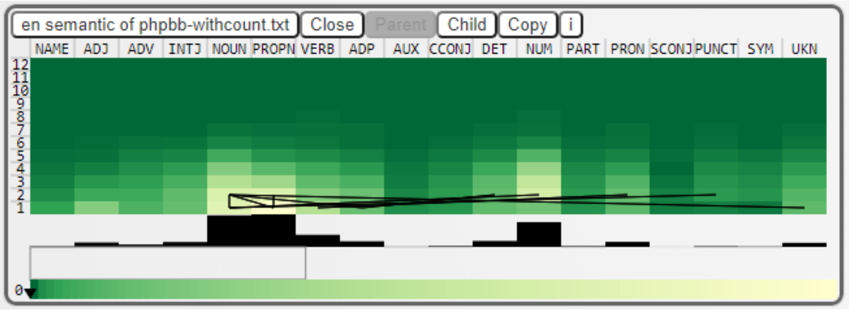

en semantic of phpbb-withcount.txt
Close
Parent
Child
Copy
i
NAME ADJ ADV INTJ NOUN PROPN VERB ADP AUX CCONJ DET NUM PART PRON SCONJ PUNCT SYM UKN
12
11
10
9
8
7
6
5
4
3
2
1
0

Figure 19: Parallel coordinates on semantic analysis of phpbb dataleak

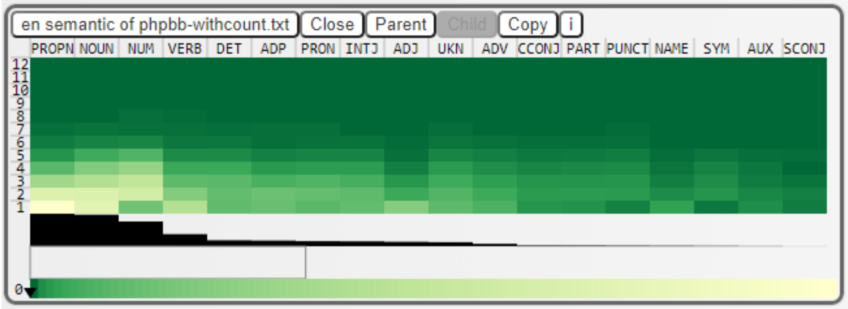
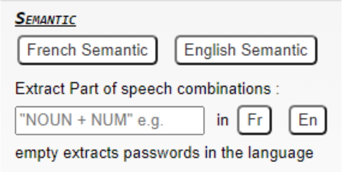

en semantic of phpbb-withcount.txt
Close
Parent
Child
Copy
i
PROPN NOUN NUM VERB DET ADP PRON INTJ ADJ UKN ADV CCONJ PART PUNCT NAME SYM AUX SCONJ
12
11
10
9
8
7
6
5
4
3
2
1
0

Figure 20: X axis ordered by frequency on semantic analysis of phpbb dataleak



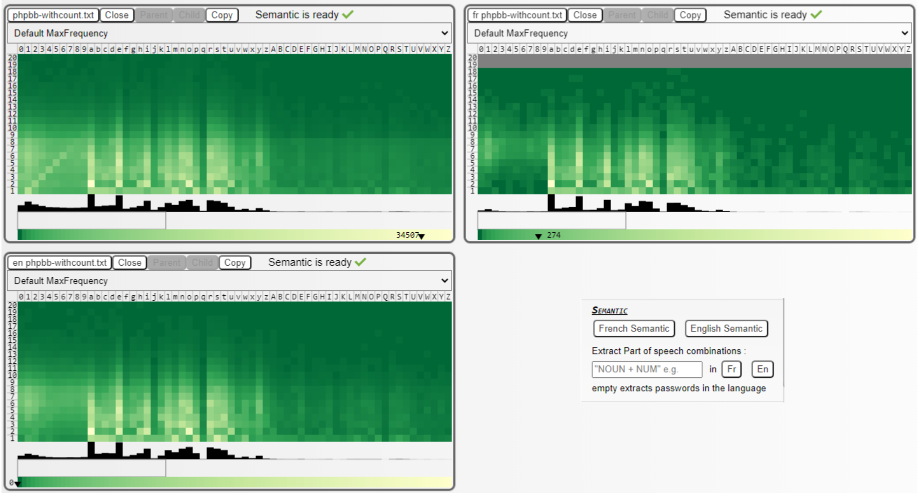
Extract Part of speech combinations :

empty extracts passwords in the language

Figure 21: Semantic section in the parameter container

5.3.6 Semantic and Language extract

Now that the semantic process is ready, more advanced filtering is possible. As every database has their passwords stored with a sequence of POS, a language and their frequency associated. As displayed on Figure 21, an input is added to get from the server passwords from a combination of POS tag from every selected heatmap in different language. Pressing ”Fr” or ”En” with an empty input will return passwords in the wanted language from selected heatmaps. These heatmap act like normal heatmap, the user can perform any test he wants to explore this database. On phpbb database an empty input have a result looking like Figure 22 and with ”NUM” as an input the user will see the Figure 23. If the server returns nothing no new heatmaps will be created.



phpbb-withcount.txt — Semantic is ready ✓ — Default MaxFrequency — 34507

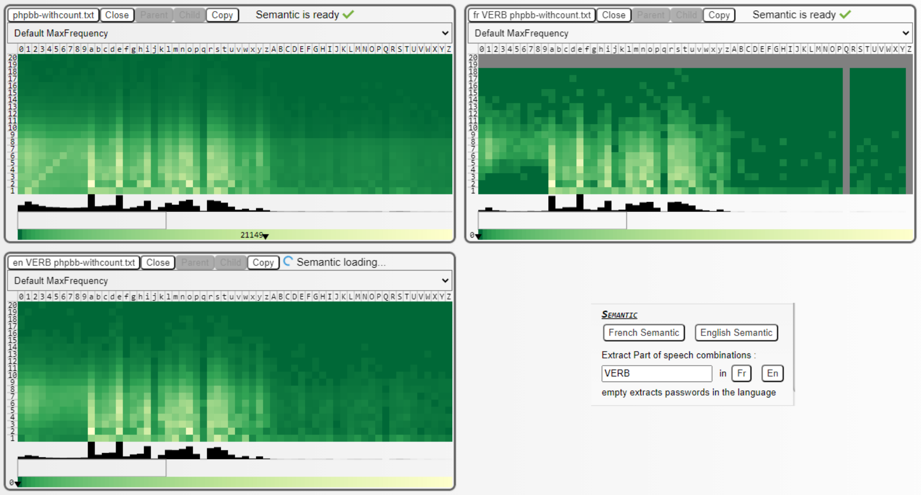
fr phpbb-withcount.txt — Semantic is ready ✓ — Default MaxFrequency — 274

en phpbb-withcount.txt — Semantic is ready ✓ — Default MaxFrequency — 0

Extract Part of speech combinations : "NOUN + NUM" e.g. in Fr En

empty extracts passwords in the language

Figure 22: phpbb extract of French(top right) and English(bottom left)



phpbb-withcount.txt — Semantic is ready ✓ — Default MaxFrequency — 21149

fr VERB phpbb-withcount.txt — Semantic is ready ✓ — Default MaxFrequency — 0

en VERB phpbb-withcount.txt — Semantic loading... — Default MaxFrequency — 0

Extract Part of speech combinations : VERB in Fr En

empty extracts passwords in the language

Figure 23: phpbb semantic extract of verbs in French(top right) and English(bottom left)

5.3.7 Max frequency Comparison

The last feature of this project is the max frequency comparison between heatmaps. A drop-down list is in every heatmap permitting to set the max frequency of an heatmap to another. By default when a semantic extract is performed, the max frequency is set to the parent value, permitting to see how much the database reduced in size. If the database is completely dark, it means that very few passwords are left after the semantic extract. In the case where the max frequency from another heatmap is lower than the actual one nothing will change and the max frequency will override the lower one. On the Figure 24 we can clearly see the effect of a higher max frequency, both are the same database but the first one have the parent max frequency. Each time the user click on the drop-down list, names of all heatmaps are loaded with their max frequency.

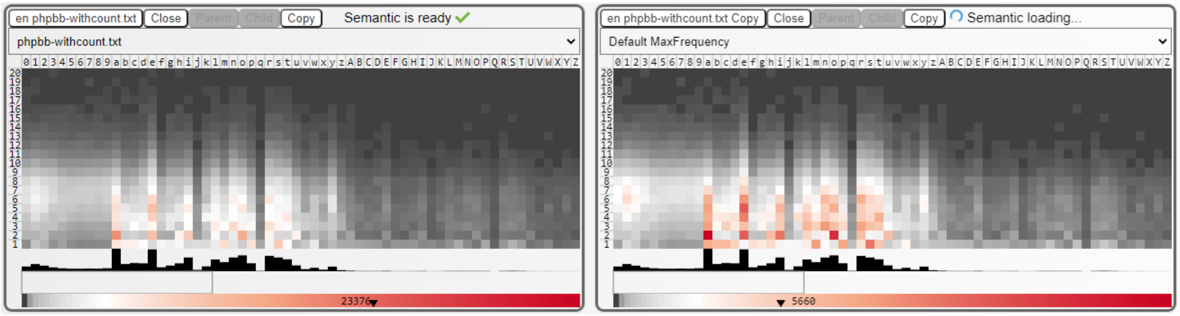

en phpbb-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
phpbb-withcount.txt
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
23376
en phpbb-withcount.txt Copy
Close
Parent
Child
Copy
Semantic loading...
Default MaxFrequency
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
5660

Figure 24: Effect of a bigger max frequency

6 Testing and results

In this section we find if the final project meets the requirements stated during the requirements creation process. If most of the requirements are implemented and working it would mean a successful project.

6.1 Functional testing

For functional tests, only the basic tests and their expectations and results are detailed in the Table 4, the more complex tests have their tests specified in their own sub-sections.

No.	Test	Expected result	Actual result
1	User want to perform analysis of the previous version	User should be able to manipulate data like before	Success, all the previous features are still working on the final version of the project
2	User want to perform an analysis in different languages	User should be able to test any language with the database	Partial success, analysis are only available in French and English
3	User attempt to interact with new semantic Heatmaps.	User can interact and semantic heatmap behave the same way as normal heatmap	Success, Semantic heatmap have been made to work like normal heatmap, all feature implemented in normal heatmap are present
4	User should be aware of the possible long waiting times	User see an notification or a message telling him to wait	Success, in addition to the loading icon on heatmaps, a popup message is displayed telling names of heatmaps where the semantic analysis is not ready yet
5	User want to compare differents heatmaps at the same time	User should expect to compare visual result and statistics of different heatmaps	Success, the user can see heatmaps side by side

Table 4: Tests, their expected result and results

6.1.1 Semantic difference between languages

One of the major features of this project is the Semantic heatmaps. They have been created to make shows the semantic differences in passwords in various languages. This feature is made to highlight those differences and permit the user to compare them through the visualisation. For example with the Figure 25 the user can clearly see that passwords in French and in English are not built the same way. In the phpbb database English passwords tends to use more proper nouns and but french and english has a similar relation with digits in passwords. Moreover french passwords lack of particle (for example 's or 't) as it is a specific trait of english language. The test is considered successful as differences between languages can be seen.

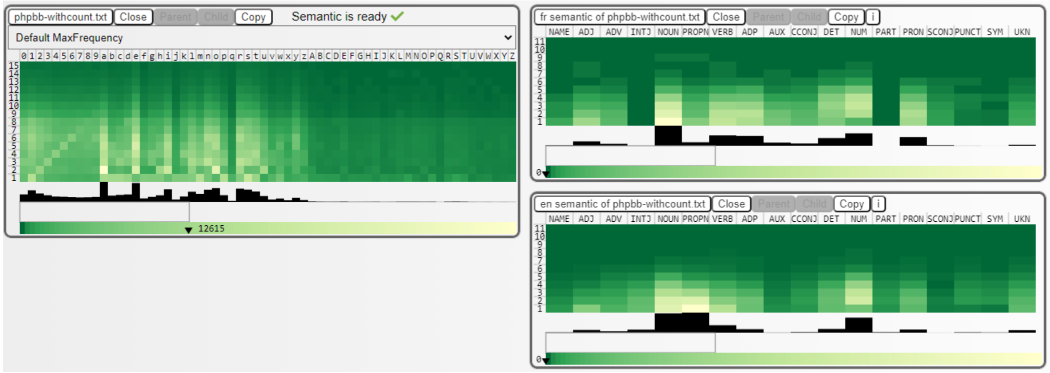

phpbb-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
Default MaxFrequency
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
12615
fr semantic of phpbb-withcount.txt
en semantic of phpbb-withcount.txt
i
NAME ADJ ADV INTJ NOUN PROPN VERB ADP AUX CCONJ DET NUM PART PRON SCONJ PUNCT SYM UKN
0

Figure 25: Difference between the semantic heatmaps according to their languages from phpbb data leak

6.1.2 Set Manipulation

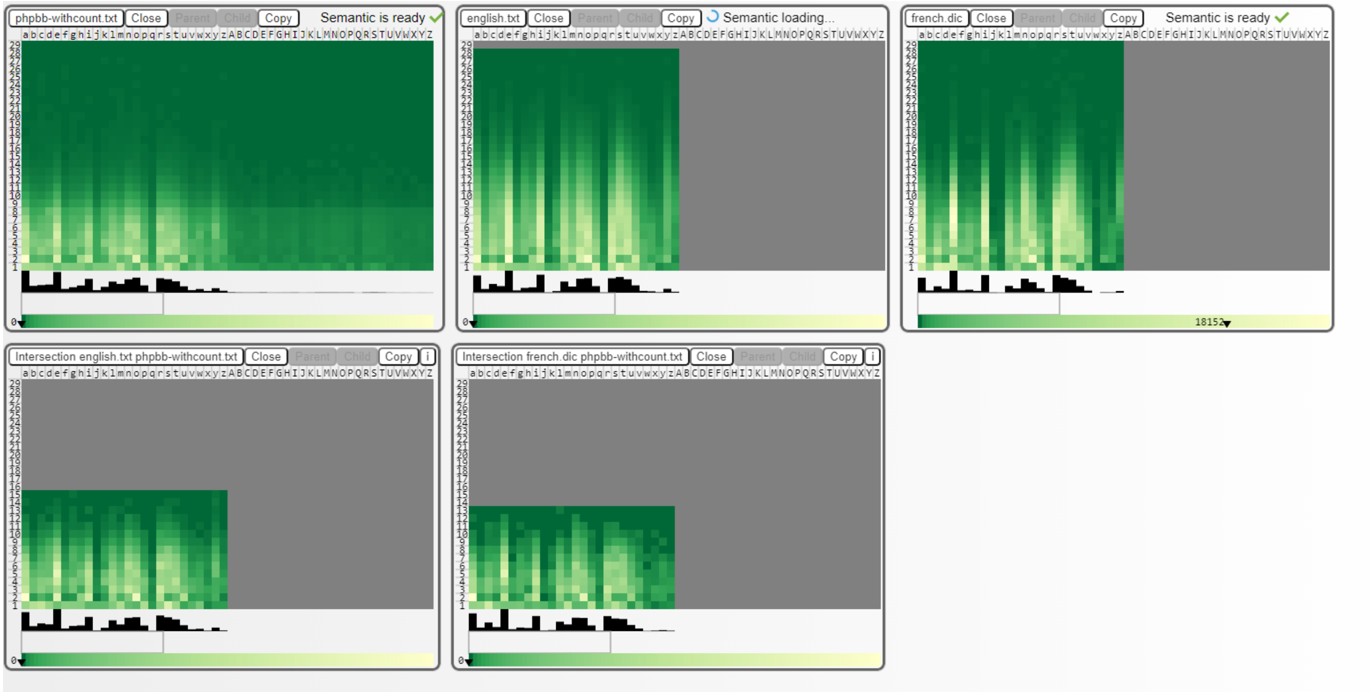

phpbb-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
english.txt
Semantic loading...
french.dic
a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
18152
Intersection english.txt phpbb-withcount.txt
Intersection french.dic phpbb-withcount.txt
i
0

Figure 26: intersection of french and english dictionary with phpbb data leak

It is important for the user to have a feature that makes him able to find similar passwords in two databases or remove common passwords to look for a common behavior between databases. Thanks to the set manipulation implemented with the drag and drop function he can do so. For example if the user wants to find dictionary words existing in a database, he only has to upload the said dictionary, drag and drop to see the difference analyse the result. A test conducted with the phpbb data leak Figure 26 to see if a specific behavior can show up. And we can clearly see that the common passwords in dictionaries are for most of them short words. As a behavior can be identified, this implementation is considered as successful.

6.1.3 Heatmap based on semantic

Finally, the last test concerns the heatmaps extracted from a combination of POS tags. Whenever the semantic analysis the user should be able to get a normal heatmap based on tags wanted to perform an analysis. Some tests were run to observe a possible conduct in the phpbb data leak when tags NOUN and NUM are presents, the Figure 27 is set with the parent Max Frequency and the Figure 28 is with their default Max Frequency. The test is successful as the user could extract a part of the heatmaps based on a POS tags, the max frequency also helps the user to understand how much the number of passwords has reduced.

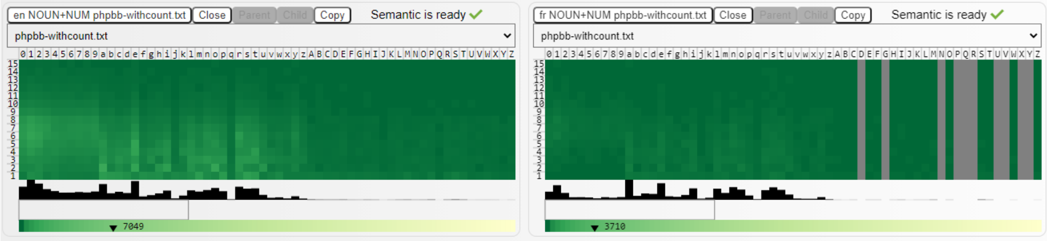

en NOUN+NUM phpbb-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
phpbb-withcount.txt
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
7049
fr NOUN+NUM phpbb-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
phpbb-withcount.txt
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
3710

Figure 27: NOUN+NUM extracted from phpbb with Max Frequency of phpbb

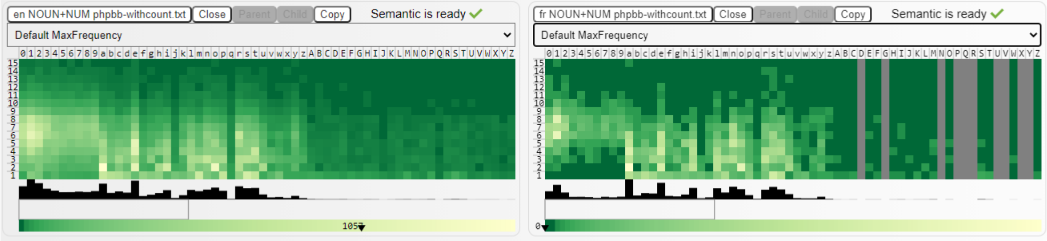

en NOUN+NUM phpbb-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
Default MaxFrequency
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
105
fr NOUN+NUM phpbb-withcount.txt
Close
Parent
Child
Copy
Semantic is ready ✓
Default MaxFrequency
0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
0

Figure 28: NOUN+NUM extracted from phpbb with their own Max Frequency

6.2 Compatibility testing

As stated by Gary Read, it is important that the project can work on various operating systems and need as little installation to be run by anyone, to make it as accessible as possible. After some tests the web page is actually working on Brave, Chrome, Firefox and Microsoft Edge and Safari, but the python server working only on operating system that can execute python code like windows, linux or macOS which limit the usage of the project to desktop only. To be usable on a mobile device, it would be necessary to deploy the server on internet, but it would cause some issue as the request would take more time as it is not anymore the CPU working on local file.

7 Conclusion

To put it in a nutshell, during this project how passwords are structured, designed and how to visualize them have been studied deeply, which allowed to understand and create an improved version of the interactive visualisation tools. The goal of this section is to assess whether or not if the project is a success. The following subsection is a critical evaluation of the project made by the author. And thereafter possible upgrades, idea to improve the project followed by technical limitations.

7.1 Self evaluation of the project

Considering only my part of the work the last version, the project is able to shown different patterns according to the language the user want to work with. The project started with a literature review to understand how people create their passwords and what is difficult to create and use every day, then research of new ways to do visualisation and thereafter study demonstrating the strength of passwords cracking using semantic analysis. Afterwards, the requirements of the project are discussed to create a basic concept of the project. The implementation phase has made me realise that some features were out of scope and would take too much time to do. As stated before the completions of the requirements will determine the successfulness of the project which are all achieved and met. We conclude that the project is ready to be used by a user and is fully functional and is a success in terms of objectives defined.

7.2 Future work

Even if we have taken into account all the new features, there are still many ways to improve the visualisation tool. To improve the portability of the project, a docker container could be set up, having instantly all the dependencies and installation ready to work. Moreover the work made by Stephanie could be implemented as her work is fascinating thanks to the Regular expression approach she took but sadly she created to much error that could slow the user. This has to be done cautiously as her work is including Regular expression and errors are easy to do with them. Furthermore, it could be possible on the server sides to add a verification if the heatmap hasn't changed between two upload. And at last an heatmap drawn on a keyboard pattern could be really interesting to see, for example each key would have a color and a slide bar could changer color according to his position in the string like stated in the literature review.

Also, the set intersection could be improved, as of today the intersection find passwords strictly equals and present in both databases. It should be possible to find passwords from Database A contained in a password from Database B. But in this case I am limited by JavaScript and Python, no system function doing this exist and functions I made myself are very slow and can take more than 20 minutes for a small database.

References

al., Li et (2016). “A study of personal information in human-chosen passwords and its security implications”. In: *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9. DOI: `10.1109/INFOCOM.2016.7524583`.

Berry (2012). *Pin number analysis.* [Online]. URL: `http://www.datagenetics.com/blog/september32012/`. [Accessed: 25 August 2021].

Melicher, William et al. (2016). “Fast, lean, and accurate: Modeling password guessability using neural networks”. In: *25th {USENIX} Security Symposium ({USENIX} Security 16)*, pp. 175–191.

Nielsen, Jakob (1994). “Usability inspection methods”. In: *Conference companion on Human factors in computing systems*, pp. 413–414.

Pearman, Sarah et al. (2017). “Let’s go in for a closer look: Observing passwords in their natural habitat”. In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 295–310.

r/dataisbeautiful stats (2021). [Online]. URL: `https://subredditstats.com/r/dataisbeautiful`. [Accessed: 25 August 2021].

Read, Gary (2016). “Interactive Password Visualisation”. In:

Schweitzer, Dino et al. (2011). “Visualizing keyboard pattern passwords”. In: *Information Visualization* 10.2, pp. 127–133.

Shay, Richard et al. (May 2016). “Designing Password Policies for Strength and Usability”. In: *ACM Trans. Inf. Syst. Secur.* 18.4. ISSN: 1094-9224. DOI: `10.1145/2891411`. URL: `https://doi.org/10.1145/2891411`.

Veras, Rafael, Christopher Collins, and Julie Thorpe (Jan. 2014). “On the Semantic Patterns of Passwords and their Security Impact”. In: ISBN: 1-891562-35-5. DOI: `10.14722/ndss.2014.23103`.

Veras, Rafael, Christopher Collins, and Julie Thorpe (Apr. 2021). "A Large-Scale Analysis of the Semantic Password Model and Linguistic Patterns in Passwords". In: *ACM Trans. Priv. Secur.* 24.3. ISSN: 2471-2566. DOI: `10.1145/3448608`. URL: `https://doi.org/10.1145/3448608`.

Yu, Xiaoying and Qi Liao (2019). "Understanding user passwords through password prefix and postfix (P3) graph analysis and visualization". In: *International Journal of Information Security* 18.5, pp. 647–663.