POLYTECH TOURS
64 avenue Jean Portalis
37200 TOURS, FRANCE
Tél +33 (0)2 47 36 14 14
www.polytech.univ-tours.fr

Rapport SEO

Mise en œuvre d'outils SEO pour la prédiction de trafic Web

Etudiant :
MOULINOU Robin
DAKROUB Mohamad Ali
Tuteur académique:
DELALANDRE Mathieu

Introduction

Ce projet s'inscrit dans le cadre des projets de programmation et génie logiciel de Polytech tours pour le Semestre 7, le but de ce projet est de nous faire gagner en autonomie et en gestion de projet, nous avons choisis ce sujet car nous étions tous les deux curieux d'apprendre des nouvelles choses sur le SEO pour mieux comprendre le fonctionnement d'internet et des recherches web.

Nous remercions Mr Delalandre pour nous avoir superviser sur ce projet et nous avoir aider à mener à bien ce projet. Nous remercions aussi le groupe de Aurélien et Bacari avec qui nous avons travaillé étroitement.

Table des matières

1 Besoin et objectif du projet

1.1 Introduction à la SEO et Contexte

Le SEO (Search Engine Optimization) est une stratégie digitale qui regroupe l'ensemble des pratiques visant à positionner son site dans les premiers résultats des moteurs de recherche de plus c'est une étape nécessaire pour que tout site soit visible et visité. Ceci définis l'ensemble des techniques pour améliorer le référencement (position du site web dans les moteurs de recherches).

Lorsque l'on parle de SEO, nous parlons généralement de Google, ce moteur de recherche étant celui privilégié par les internautes et recevant plus de 90 % de leurs requêtes, mondialement comme en France. Pour obtenir un bon positionnement sur les pages de google plusieurs critères (chargement de la page, notoriété du site...) sont prises en compte.

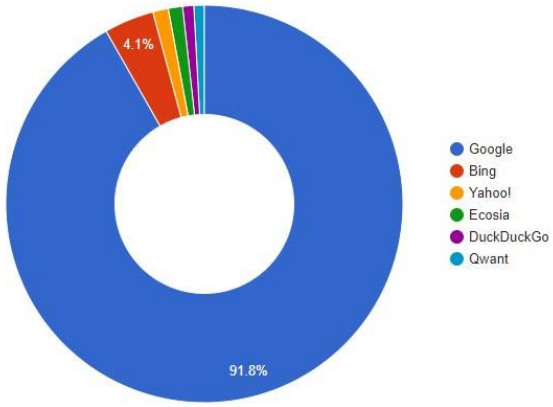

Figure 1) Statistique sur les moteurs de recherche

En effet, le choix des mots-clefs à mettre est aussi important pour bien satisfaire les utilisateurs qui veulent accéder aux informations.
Le but correspond aux actions mises en œuvre pour optimiser un site afin qu'il soit dans la première page de résultats d'un moteur de recherche tel que Google.

D'après des recherches sur la SERP, le premier de la liste correspond à 33% des clics sur ordinateur, le deuxième 15% et le troisième 10%.
Les trois premiers représentent donc presque 60% des clics.

La station TV est une plateforme de calcul pour l'intelligence artificielle et la télévision. Ses applications sont multiples du social à l'Analytics TV, en passant par le "data journalism" et les "second screen apps".

Le projet de Station TV, se décompose en plusieurs parties, La collection d'informations, catégoriser les données, enrichir les données d'origines, s'en servir dans les services numériques du projet Station TV.

1.2 Objectifs

Le but de notre projet est l'extraction des métriques d'autorité pour établir des courbes trafic/référencement. Autrement dit dans un premier temps il faudra recuperer la difficulté de placement d'un mot via un outil spécialisé en analyse SEO. Puis dans un second temps faire correspondre son score google trend et générer un nuage de point afin de pouvoir sélectionner les mots les plus pertinents.

2 Gestion de projet

2.1 Méthode de travail

Mettre en place une bonne stratégie de SEO est une des parties les plus importantes pour un site Web. Pour cela, nous avons décidé de suivre une méthodologie de développement agile nous permettant d'apporter des modifications et des améliorations tout au long des phases du développement du projet. Le tout associé a un git afin de suivre le travail de chacun. (Cliquez-ici pour y accéder)

2.2 Organisation

Après avoir analyser le sujet, nous avons pu établir et entretenir un Trello tout au long du déroulement du projet que vous trouverez ci-dessous.

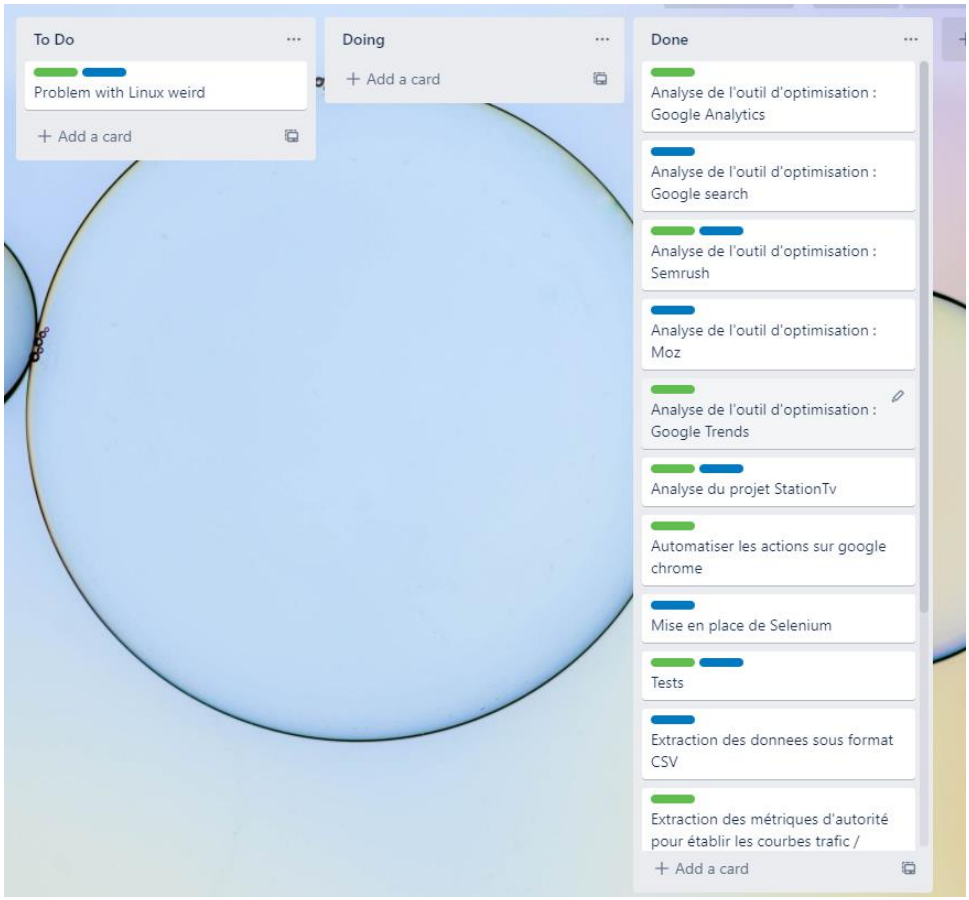

Figure 2) Trello du projet

3 Solution

3.1 Extraction de donné WEB

3.1.1 Stratégie

À la suite de l'analyse des besoins, nous avons effectué des recherches sur les outils de SEO disponible permettant de donner une notion de la difficulté à se placer pour ce mot. De nombreux outils sont disponible sur internet (Google Trends, Ubersuggest, Ahrefs, Screaming Frog, SEMrush, Moz, Clearscope / Yoast, Nightwatch) mais très peu le sont de façon gratuite.

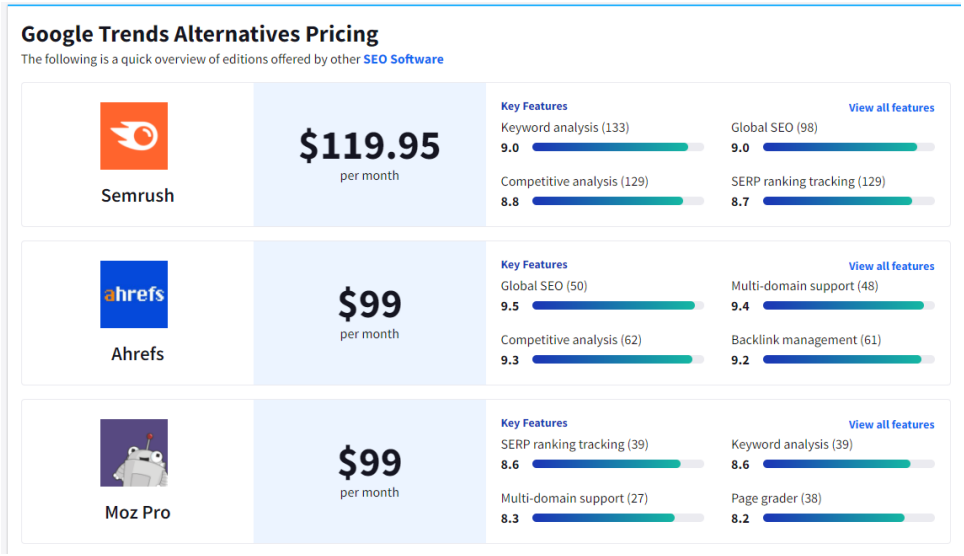

Figure 3) Prix des 3 meilleurs outils de SEO

A la fin des recherches l'outils qui nous sembla le plus pertinent fut Semrush.

L'outil propose également des fonctionnalités plus avancées. Il permet une analyse du trafic, des mots clés et des liens, ce n'est pas tout, l'outil permet également une analyse plus technique du site. Ce qui nous a permis de récupérer une analyse du positionnement du nom de domaine à une date donnée, l'évolution de la position sur 30 jours, la tendance de visibilité, le nombre de fois ou le mot clé qui a été utilisée dans une requête et encore énormément d'autres informations.
Notre seul obstacle fut que seulement les dix premières requêtes sont gratuites et même pour une analyse plus développée il fallait partir sur la version payante.
Une API existe pour récupérer ces données depuis Semrush, mais là aussi il faut payer.

La seule solution disponible du coup consiste à automatiser des actions sur un navigateur web pour récupérer les données.

Pour effectuer ceci il existe une librairie python qui permet d'utiliser google chrome et d'en prendre le contrôle et donc de naviguer, faire des requêtes, récupérer des données.
Selenium est un outil d'automatisation open source pour tester une application web. Nous avons choisi Selenium puisqu'il est capable d'effectuer des tests sur plusieurs navigateurs et capable d'effectuer des tests sur plusieurs systèmes d'exploitation.
Les données fournis par l'autre groupes sont stockées dans un fichier CSV avec le format suivant : '' mot-clé ;score_googleTrend ''. Python permet facilement d'ouvrir un fichier CSV pour récupérer ses données et d'écrire dans un csv grâce à une librairie intégrée.

3.1.2 Analyse Technique

La première partie du programme va recuperer les arguments passés lors du lancement, c'est-à-dire un email, un mot de passe ainsi que le chemin du fichier contenant les données à traiter.

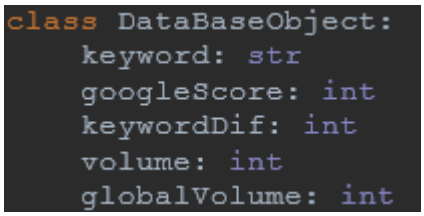

Figure 4) Architecture de la classe DataBaseObject

Il faut ensuite lire les données contenues dans le CSV est les insérer dans une liste de classe ''DataBaseObject'' qui va contenir le mot, le score google trend et une succession de donnée que l'on peut recuperer via SEMRUSH (difficulté de placement du mot, volume en France et volume global). Si la valeur est de -1 cela signifie que le mot n'a pas encore été traite par notre logiciel et si la valeur est de -2 cela signifie que le mot a était traiter mais que SEMRUSH n'a pas d'information sur ce mot.

Nous choisissons maintenant les 10 premiers mots ayant des valeurs égales à -1. Car SEMRUSH en version gratuite a une limite de 10 mots par 24h.
Maintenant nous ouvrons chrome à l'aide de ChromeDriver qui va permettre avec sélénium de prendre le contrôle de l'instance Google Chrome (ci-dessous).

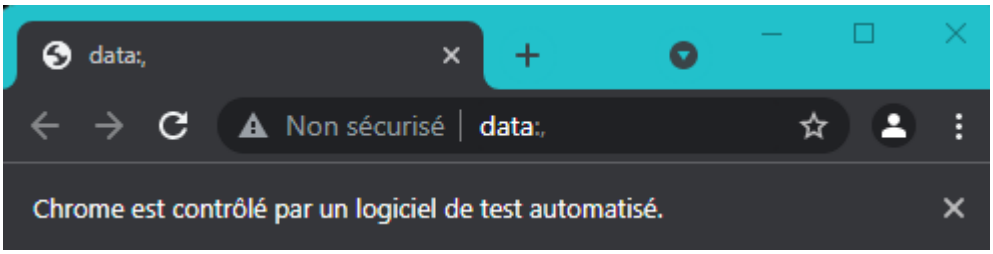

Figure 5) Capture d'écran ChromeDriver

Puis nous allons naviguer à l'adresse http://semrush.com/projects/ il est possible qu'il y'ai des cookies qui soit encore présent et qu'un compte sois déjà connecté, si c'est le cas on continue sinon on navigue à la page http://semrush.com/login/ et l'on va rentrer les logins passés au lancement du programme.

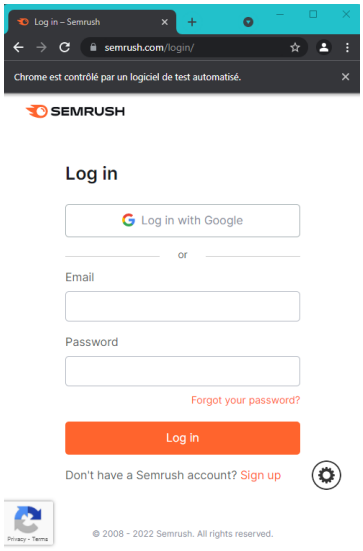

Figure 6) Page de connexion Semrush

Afin de détecter des éléments tel que des champs d'entre, des boutons, ou du texte. Nous utilisons la chemin XPATH des éléments. Le XPATH permet de naviguer dans l'architecture HTML d'un site WEB.
Voici un exemple pour mieux comprendre :

Pour recuperer le XPATH du champ d'entré de l'adresse électronique, il faut via Google Chrome faire clique droit inspecter l'élément qui nous donné le code HTML de la page (voir ci-dessous).
Il est bien sur recommandé d'avoir quelques connaissances en HTML pour comprendre l'architecture d'un site WEB et pouvoir donc retrouver les bons éléments.

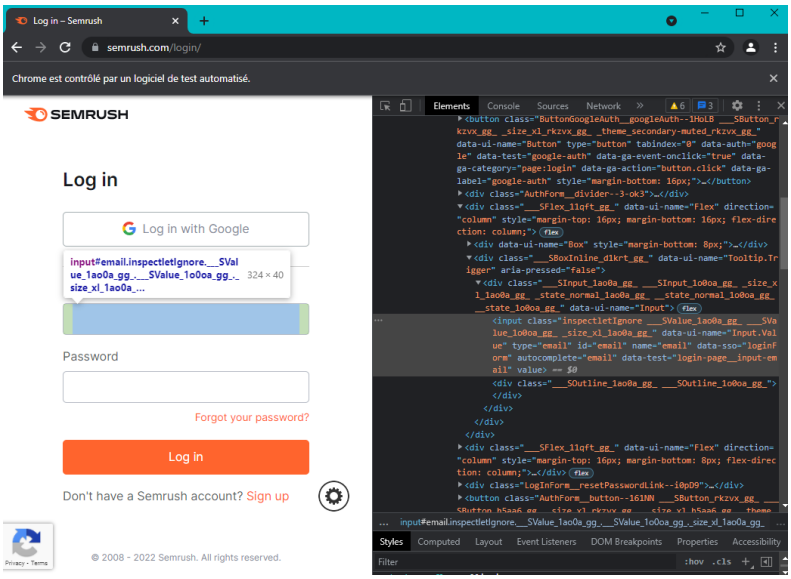

Figure 7) Outils de développement Google Chrome

Le bout de code HTML correspondant au champ d'entré et est déjà sélectionné il faut faire clique droite et choisir copier tout le XPATH qui va nous donner cette chaine :

/html/body/div[1]/main/div/main/div/div/form/div[2]/div[2]/div/input

A l'aide de sélénium nous pouvons ensuite lui demander de recuperer cet objet et de lire sa valeur ou en l'occurrence envoyer du texte.

Notons que parfois il est possible que le système captcha anti-robot de SEMRUSH s'active ce qui nous empêche de pouvoir y accéder mais cela peut être résolu en attendant quelques heures ou avec un changement d'adresse IP (voir ci-dessous).

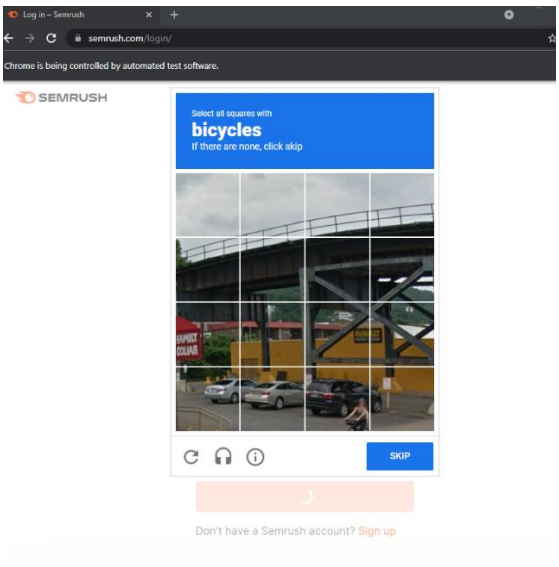

Figure 8) Système anti-robot SEMRUSH

Il va ensuite falloir attendre que la page soit bien chargée, une fois cela fait nous allons pouvoir naviguer à l'adresse suivante :
https://www.semrush.com/analytics/keywordoverview/?q=[LE_MOT]&db=fr
Elle correspond à l'adresse pour une requête SEMRUSH (voici un exemple ci-dessous avec le mot Polytech).

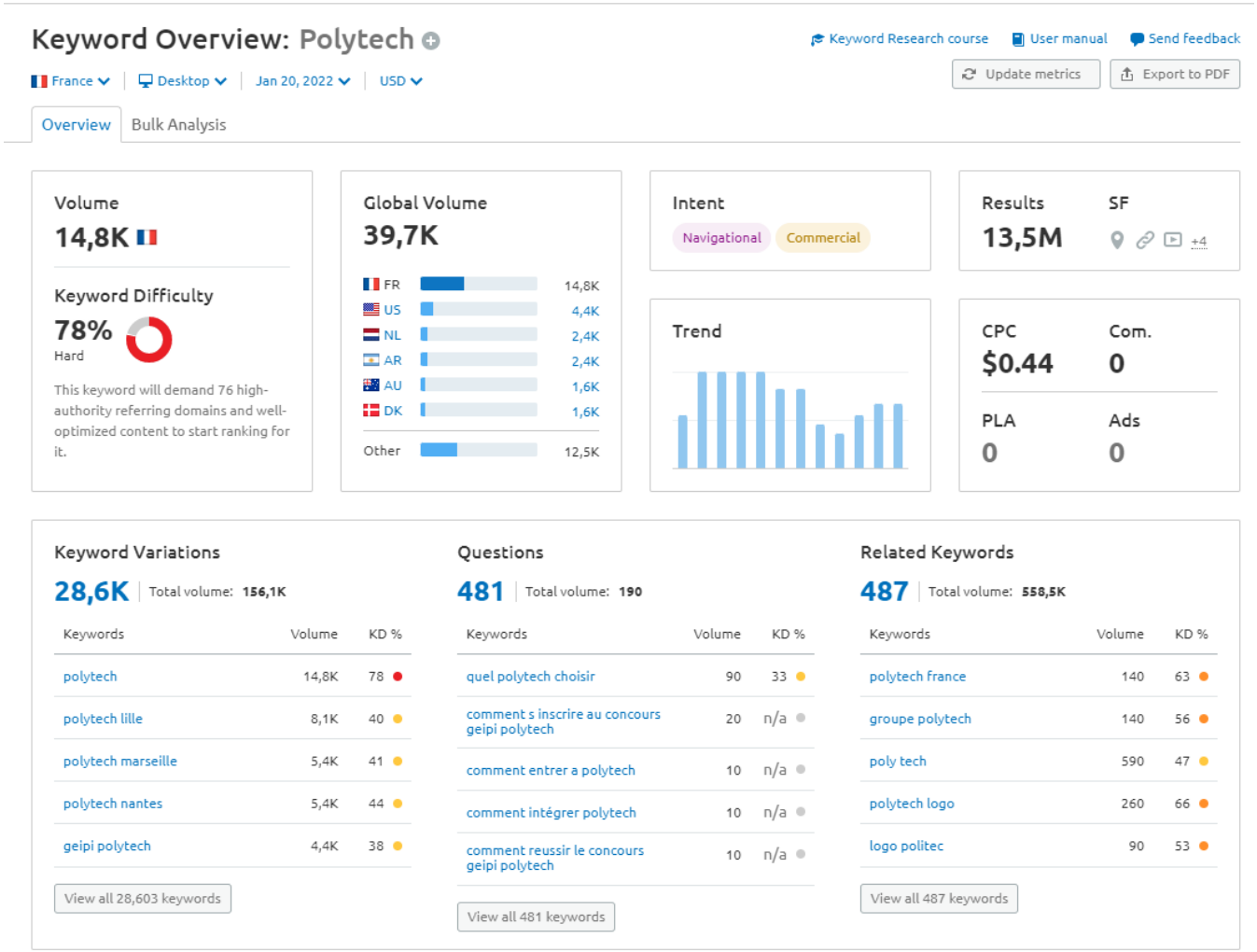

Figure 9) Résultat de recherche pour le mot Polytech

Une fois cette page chargée nous récupérons les données pour la difficulté du mot, le volume local et le global volume grâce au XPATH que nous avons cherché(notons qu'il est possible de recuperer d'autres donné plus tard en rajoutant des éléments à la classe DataBaseObject et des chemins XPATH).
Les données recuperer sont sois une valeur en pourcentage, soit un nombre de ce style : 10K, 20M, ou alors la valeur n/a.

Nous avons donc créé une fonction stringConverter qui s'occupe de faire les transformations nécessaires pour avoir une valeur adéquate (en l'occurrence -2 si aucune donné existe ou alors un nombre au bon format (10k -> 10 000).

Une fois toutes nos donné recuperer il faut exporter les données dans le fichier csv de base. Pour ce faire il faut transmettre un tableau, nous avons donc fait une fonction ToExportFormat qui va transformer un objet DataBaseObject en un tableau.
Ce tableau est ensuite ajouté à un tableau qui contiendra tous les nouveaux tableaux créer. Voici un exemple :

[[mot1,score google, score semrush, ...],[mot2,score google, score semrush, ...]]

Une fois les données préparer il suffit de les exporter dans le fichier csv. Et le travail du programme est fini.

Nous avons implémenté un petit script en batch (Windows) qui va exécuter le programme plusieurs fois avec plusieurs faux comptes créés pour le projet afin de traiter plus de données par 24h.

3.2 Visualisation des données

3.2.1 Stratégie

Pour cette partie du projet il est nécessaire d'afficher un graphique de point avec des annotations, heureusement une librairie très puissante existe pour faire des tracés de courbes et de point : MatPlotLib.

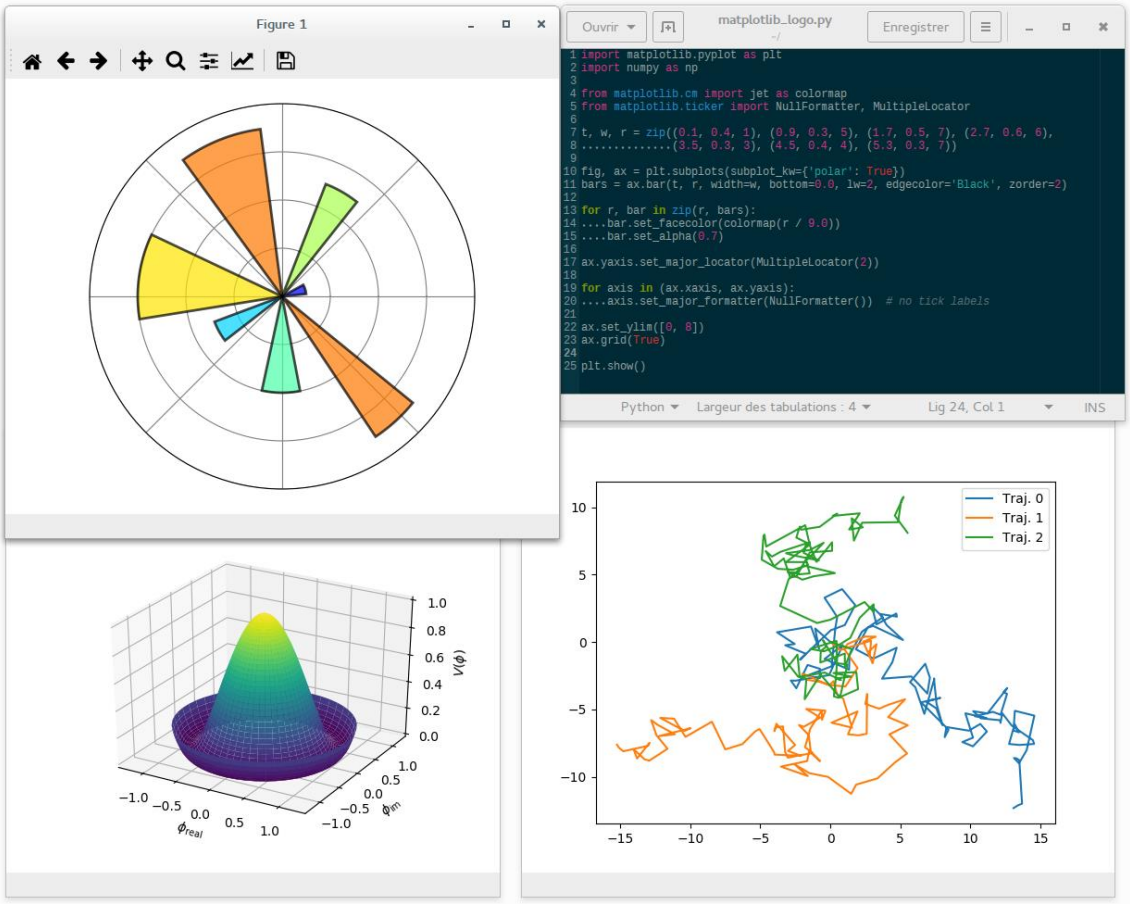

Figure 10) Exemple de tracé avec MatPlotLib

Le but est donc de récupérer toutes les données depuis notre fichier CSV, les stocké dans des tableaux et enfin tracer les points avec PyPlot (module de la librairie MatPlotLib très simple qui permet des tracés simple) et y ajouter des annotations pour savoir quel mot correspond à quel point.

3.2.2 Analyse Technique

Tout d'abord ce programme peut prendre au démarrage un argument qui correspond au chemin du fichier contenant les données traitées par le programme précédant.

Une boite de dialogue permet de confirmer si oui ou non ceci est bien le chemin souhaité. Si le programme démarre sans argument alors le chemin proposé est vide.
Vous pouvez donc choisir d'écrire le chemin que vous désiré et valider.
Une vérification est effectuée pour s'assurer que le fichier est bien un fichier CSV et qu'il existe bien.

À la suite de cela le programme va ouvrir le fichier CSV (si le fichier est déjà ouvert par un autre logiciel il est possible qu'une exception sois généré). Puis il va traiter ce fichier CSV pour en extraire toutes les données (mot clé, score google trend, score Semrush) qui vont être utile au traitement. A noter que les données de volume et volume global sont disponible et pourrait être traiter mais ceci ne fait pas partie du périmètre de notre projet. Les objets sont stockés dans un tableau contenant des objets de type :

```
class ExtractedData:
    keyword: str  # the keyword
    googleTrend: int  # google trend score
    SEMrsuh: int  # Semrsuh score
```

Figure 11) Architecture de la classe ExctractedData

Une fois les données récupérer il est temps de le préparer pour le traitement. À la suite de recommandations sur l'utilisation du module PyPlot nous séparons les données en 3 tableaux (un tableau contenant les mots clé, un contenant le score google trend et enfin un contenant le score Semrush).

Les données sont maintenant dans un format qui convient à PyPlot, il faut maintenant configurer la figure en elle-même a l'aides des fonctions de base du module (taille de l'abscisses et ordonnée, ainsi que l'ajout de label).

Pour créer le tracer des points il suffit d'utiliser la fonction *« scatter »* (qui signifie littéralement dispersion). Pour notre usage nous allons simplement fournir nos deux tableaux contenant le score google trend et le score Semrush, cependant la fonction accepte beaucoup plus de paramètre de façon à personnaliser le tracé (forme des points, couleur, etc.).

Nos données sont donc maintenant affichées mais seul problème nous ne savons pas quel point appartient à quel mot clé. Pour ce faire après quelques recherches nous sommes tombés sur une discussion Stackoverflow qui explique comment mettre en place

un système d'annotation quand la souris passe au-dessus d'un point.(cliquez ici pour le lien)
Nous avons donc implémenté cette solution et l'avons adapté pour correspondre à nos besoins.

Le principe est de créer une annotation avec un style que nous avons définis et par la suite cacher cette annotation.
Il ne reste ensuite qu'à récupérer les évènements de la souris dans la fenêtre PyPlot. Quand la souris passe au-dessus d'un point nous récupérons sa position, faisons apparaitre l'annotation en changeant sa position et son texte pour correspondre au donné et l'emplacement du point au-dessus duquel se trouve la souris.

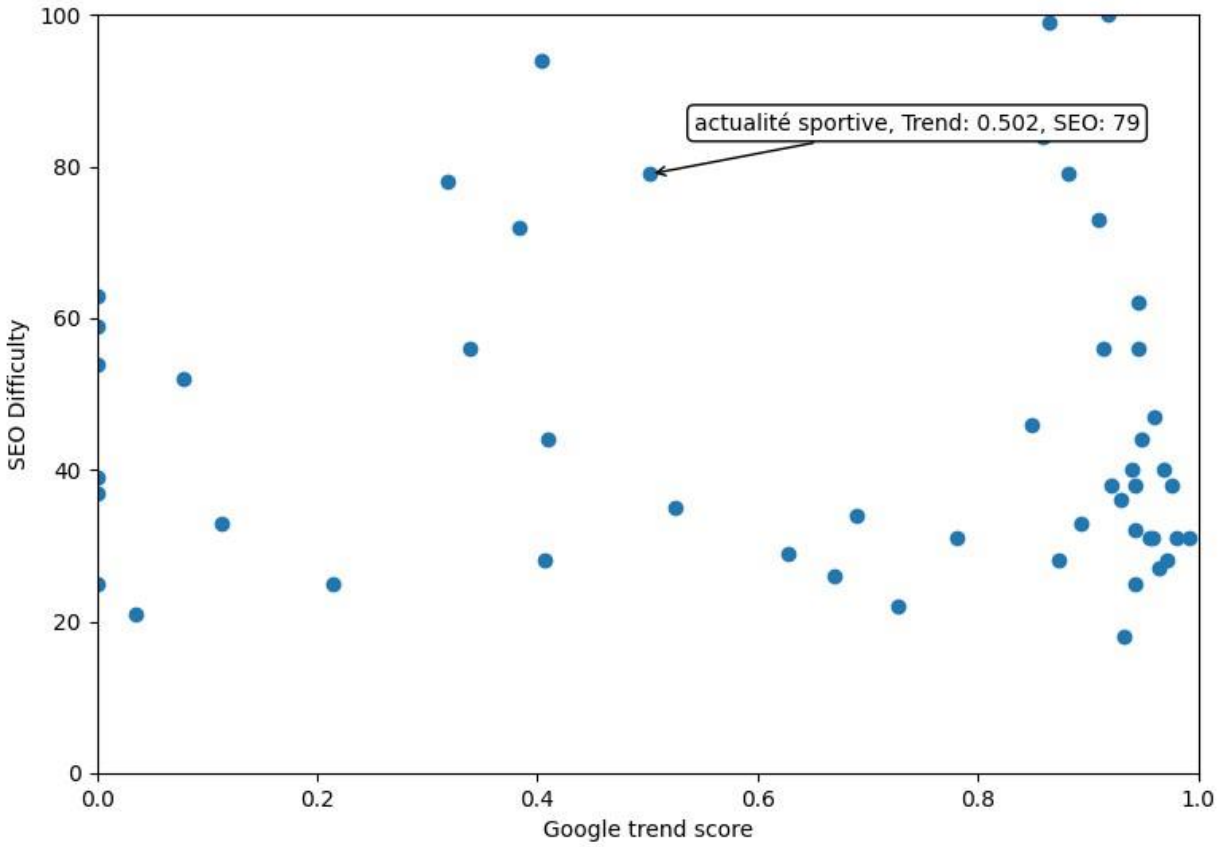

Figure 12) Résultat des données

Il est donc maintenant possible de trouver les mots les plus intéressant pour se placer en choisissant des mots ayant un faible score semrush (donc faible compétition de placement sur ce mot) et avec un fort score google trend (donc très recherché).

4 Conclusion

Pour conclure, ce projet nous as apporté énormément de connaissances, que ce soit en gestion de projet, en SEO ou bien même en python (qui n'est pas un langage que nous maitrisons particulièrement).
L'objectif était très clair dès le début, travailler de pair avec le groupe de la partie extraction de donnée et utilisé leurs données pour y agréger nos données de SEO et enfin obtenir une courbe Traffic / référencement. Il nous aura fallu un peu de temps pour bien comprendre la base même de ce projet qu'est le projet de Station Tv.

Là où ce projet fut complexe pour nous fut dans le fait d'avoir une très grande liberté et donc un grand choix de Framework, de module, d'outils possible. Nous devions donc faire le meilleur choix possible pour avoir un projet le plus performant possible.
Mais une fois le choix fait pour Selenium et PyPlot alors le projet fut assez rapide à développer car Python possède énormément de petite fonction pour nous faciliter la tâche.

Le seul vrai problème qui pourrait survenir serait que l'architecture du site Semrush change et il faudrait donc revoir tous les Xpath.
Il pourrait donc être intéressant d'utiliser des plugins de type : barre d'outils de SEO comme Seoquake qui éviterai ce problème.

Finalement nous avons un projet fonctionnel qui correspond à ce qui nous as était demandé et qui correspond à ce que nous voulions faire.

5 Tables des figures