REDEV

Rapport de projet

Sujet 6

La sciences des données au service des VTubers français : notebook python et API Twitch/Youtube

Encadrants : Morgan MAGNIN, Tony RIBEIRO

28 mars 2022 (projet mené de septembre 2021 à mars 2022)

Khaled EL GHAMMARTI - Constance PHILIPPE

SOMMAIRE

Remerciements

Nous tenons à remercier vivement nos deux encadrants, Morgan MAGNIN et Tony RIBEIRO qui nous ont accompagnés lors de ces 6 mois de projet. Il nous ont fait découvrir un sujet original et très intéressant dans lequel nous avons pu pleinement nous investir avec enthousiasme. Nous leur sommes reconnaissants de leur disponibilité pour nous aider lors de problèmes techniques en programmation et pour avoir partager avec nous des informations sur le milieu et des outils qui nous serviront à l'avenir pour faire de l'analyse de données. Leur accompagnement nous a permis de nous améliorer tout au long du projet.

Enfin, nous les remercions pour la confiance accordée tout au long du projet et le soutien pour rendre ce travail accessible au public, notamment à la communauté des VTubers francophones, principaux concernés.

Lexique

VTuber :

“ V-tuber (parfois écrit Vtuber sans le tiret) est l'abréviation de Virtual Youtuber ou Youtubeur Virtuel. La culture et l'activité en tant que Vtuber/Vtubeuse est fréquemment appelée Vtubing. Si la définition précise ne fait pas toujours consensus, il est communément admis qu'un vtuber est un avatar virtuel 2D ou 3D proposant un contenu de divertissement par le biais de vidéos (ou de lives) via des plateformes en ligne telles que Youtube, Twitch etc. ”

Selon https://vtubers-fr.fandom.com/

Visual novels :

“ Un visual novel, ou roman vidéoludique en français, est un genre de jeu vidéo assez populaire au Japon, mais moins connu dans le reste du monde. ”

Selon https://fr.wikipedia.org/

IDE :

“ Un environnement de développement intégré, ou IDE, est un logiciel de création d'applications, qui rassemble des outils de développement fréquemment utilisés dans une seule interface utilisateur graphique (GUI). ”

Selon https://www.redhat.com/

Scraping :

“ Le scraping définit de façon générale une technique permettant d'extraire du contenu (des informations) d'un ou de plusieurs sites web de manière totalement automatique. Ce sont des scripts, des programmes informatiques, qui sont chargés d'extraire ces informations. ”

Selon https://www.journaldunet.fr/

Subahton :

“ Le concept d'un subathon sur Twitch provient de la combinaison des mots "abonné" et "marathon". Son but est d'aider une chaîne à créer un afflux d'abonnés grâce à un long stream. Ce concept a été largement popularisé par un événement que Ludwig, désormais streamer sur YouTube, a organisé en 2021. ”

Selon https://dotesports.com

API :

“ Une API pour application programming interface permet à deux applications de communiquer entre elles. Elle permet de rendre disponibles les données ou les fonctionnalités d’une application existante afin que d’autres applications les utilisent. ”

Selon https://www.agencedebord.com/

I. Introduction

Dans le cadre de notre dernière année d'études à Centrale Nantes, nous avons choisi de faire l'option professionnelle "Recherche & Développement". Dans cette option, un projet de R&D nous est attribué sur lequel nous devons travailler tout au long de l'année en binôme. Nous avons eu pour sujet "La science des données au service des VTubers français : notebook Python et API Youtube/Twitch".

Tout d'abord, avant de débuter avec les aspects techniques du projet, nous allons faire un petit avant propos sur le phénomène VTubers.
Les VTubers sont des diffuseurs de divertissement qui utilisent différentes plateformes comme Youtube, Twitch, TikTok, etc... Leurs particularités est qu'ils utilisent des logiciels générant des avatars 2D, 3D qui vont collés aux mouvements des avatars à ceux des VTubers.

En ce qui concerne le phénomène VTuber, nous pouvons faire un bref historique mettant en avant l'émergence de ce dernier ainsi qu'une description brève de son histoire. Tout commence en 2010, avec la première publication d'un avatar 3D, Super Sonico, mascotte de la chaîne YouTube de visual novels : Nitroplus. Ensuite, en 2011, nous avons Ami Yamato qui met en ligne sa première vidéo dans laquelle un avatar virtuel animé parle à la caméra. Puis en 2014, le lancement du premier programme hebdomadaire de météo en direct avec l'avatar Airi. Finalement, en 2016, nous avons la première youtubeuse virtuelle, Kizuna AI que l'on peut voir Figure 1, qui a atteint une popularité internationale.

Figure 1 - Kizuna AI est la première VTubeuse à s'être lancée au Japon en 2016. (Capture d'écran YouTube)

Tous ces grands événements ont permis à ce phénomène de connaître une explosion au Japon. Cette forte popularité s'est soldée par un nombre d'abonnés à la chaîne de Kizuna AI atteignant les 2 millions et un nombre de Vtubers répertoriés sur la plateforme YouTube de 16000 en 2021 [1]. Ce phénomène commence à attirer l'attention des occidentaux, d'autant plus avec l'arrivée de la Covid, avec le nombre d'internautes augmentant énormément.

Ce rapport décrit notre travail permettant d'analyser les statistiques des **VTubers français** et de proposer des sortes de fiches VTubers".

II. Objectifs de l'étude

Il s'agira dans ce chapitre de présenter les grands objectifs de notre étude. L'idée principale est de pouvoir évaluer les statistiques des streamers sur deux principales plateformes : **Youtube** et **Twitch**. Comme présenté dans la figure 2, l'idée générale est la suivante : nous allons diviser l'étude en deux : l'API Twitch et l'API YouTube. Puis, nous croiserons les données collectées.

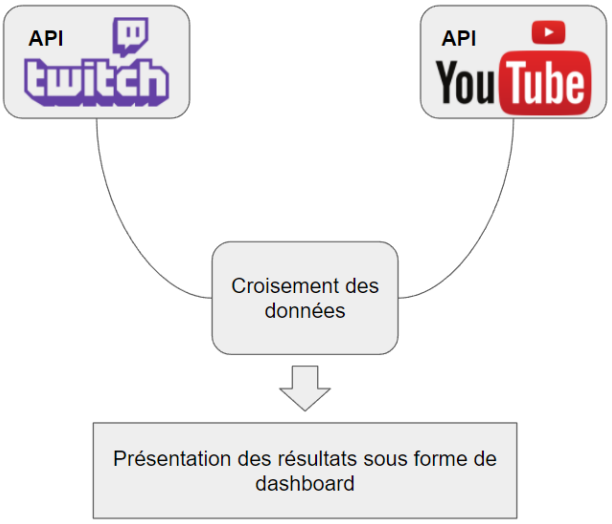

Figure 2 - Schéma du principe de l'étude

Pour entrer légèrement dans le détail, notre principal objectif est de se former sur les technologies utilisées pour pouvoir faire ces analyses. Ensuite, nous devrons rédiger du code nous permettant de récupérer toutes les informations nécessaires à l'analyse. Puis, nous créerons du code utilisant les technologies vues lors de notre apprentissage nous permettant de mettre en place les outils générant les graphiques mettant en évidence les résultats de l'analyse. Finalement, ces résultats permettront une analyse sur l'aspect financier et voir à quel point ce phénomène génère de ressources.

Dans la suite de ce rapport, ces différents points seront expliqués le plus précisément possible.

III. Etat de l'art

Il s'agit d'un sujet d'étude assez novateur dans le fait qu'on étudie d'une part des VTubers et d'autre part le fait que l'étude soit ciblée sur des francophones. Il n'y a donc pas d'études qui ont été faites jusqu'à présent. En revanche, nous pouvons présenter des analyses qui ont été faites, assez similaires à la nôtre, sur un domaine d'étude assez proche.

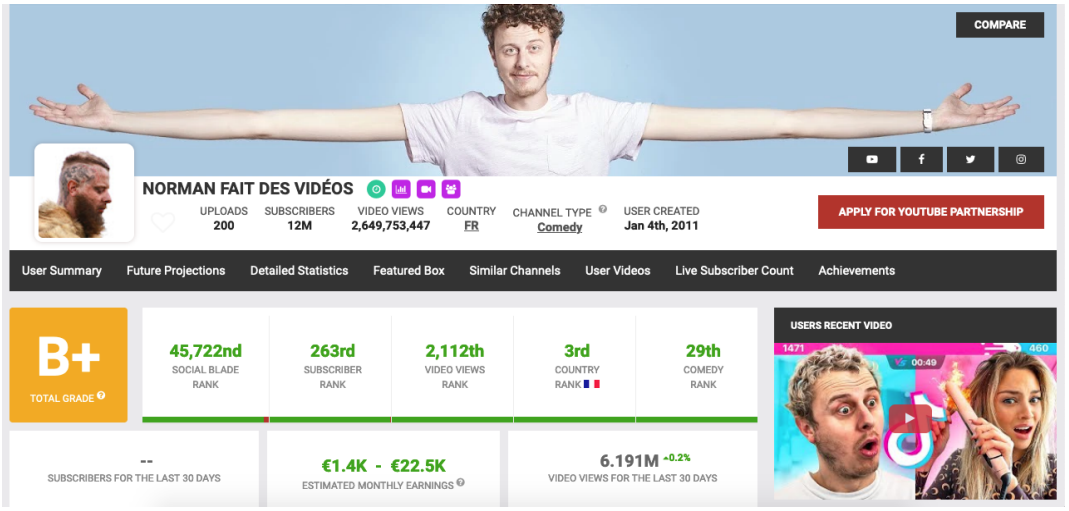

Figure 3 - Aperçu de la page contenant des statistiques de Youtubeur (ici Norman)

Il existe par exemple une plateforme internet appelée *SocialBlade* [2], créée en 2008 par Jason Urog, qui permet d'avoir une étude statistique sur n'importe quel utilisateur de Youtube, Instagram, Twitch, Facebook, Twitter, TikTok, pour ne citer que les plus connus des réseaux. En entrant le nom d'une chaîne Youtube par exemple, on accède à diverses informations : date de création de la chaîne, nombre d'abonnés, nombre de vidéos, etc. Il est également possible d'avoir une estimation du revenu mensuel que touche une personne sur une plateforme spécifique.

Les données sont très intéressantes et informatives. On a en effet un rapide aperçu d'un créateur de contenu, sur sa popularité, ses potentiels revenus. On peut également voir s'il y a une baisse de popularité en regardant le nombre d'abonnés au cours du temps, le nombre de vues, etc.

Ce faisant, cette plateforme met dans le même temps à disposition une expertise visant un public de professionnels souhaitant par exemple dynamiser leur chaîne Youtube et leur communauté. Il y a donc derrière un intérêt économique majeur, les réseaux représentant une grosse part du marché.

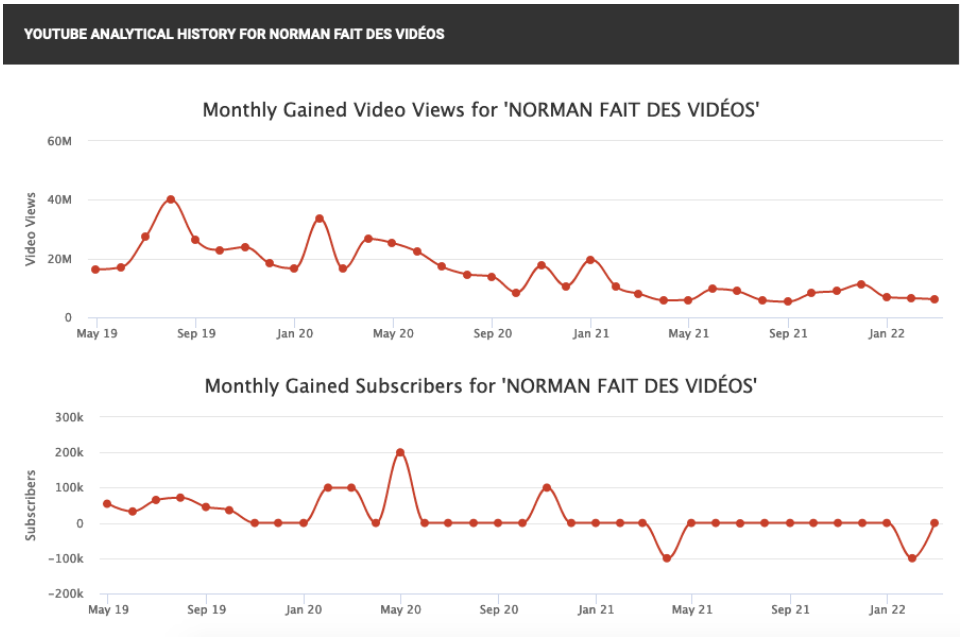

Figure 4 - Analyse sur Youtube sur SocialBlade (évolution du nombre de vues et abonnés)

IV. Methodologie

A. Outils utilisés

1. Langage de programmation et bibliothèques

a) Python et Jupyter Notebook

Pour ce projet, nous avons utilisé le langage de programmation Python qui était une compétence requise au projet. Il s'agit d'un langage assez simple à comprendre et à apprendre. Il est également très adapté à l'analyse de données avec les nombreuses librairies à disposition.

L'utilisation d'un notebook Jupyter (ou sur Kaggle) permet d'avoir une vision plus interactive, plus directe. Au lieu d'avoir un IDE et une console d'un autre côté, il s'agit de cellules sur une page (notebook) que l'on lance les unes après les autres et sous lesquelles s'affichent le résultat demandé. Un notebook a également l'avantage d'avoir la possibilité d'écrire en Markdown dans les cellules. Cela permet une meilleure lisibilité, en ajoutant des explications, en structurant le code sous forme de plan avec des résultats, des graphiques directement lisibles.

b) Bibliothèque pandas

Pour la manipulation et l'analyse de données, une bibliothèque intéressante et très utilisée dans Python est pandas. Les données sont sous forme de dataframe. Un dataframe est un tableau de deux dimensions. Il y a donc des colonnes avec un nom et chaque colonne se voit attribuer un ensemble de valeurs : il s'agit d'une série. Les données sont exportables au format JSON ou CSV.

Les avantages d'utilisation d'une telle bibliothèque sont multiples. Il est très simple de travailler avec un dataframe même s'il y a des données manquantes à l'intérieur et de manipuler ces données. Il existe en effet un ensemble de fonctions définies dans la bibliothèque qui permettent de travailler sur les dataframes. L'utilisateur peut ainsi filtrer les données avec des conditions sur les lignes et/ou colonnes, supprimer des colonnes ou des lignes, ajouter une nouvelle ligne ou une nouvelle colonne, sélectionner des valeurs spécifiques, fusionner des données, etc.

	id	channelTitle	publishedAt	title	description	duration	viewCount	likeCount	dislikeCount	commentCount
703	E1OcZFAhBV4	Selph1ne	2021-12-11T12:45:01Z	{'publishedAt': '2021-12-11T12:45:01Z', 'chann...	Tu connais ce mec qui possède les autres penda...	PT4M4S	478	none	NaN	14
704	p53IRRQS0VY	Selph1ne	2021-12-08T16:30:06Z	{'publishedAt': '2021-12-08T16:30:06Z', 'chann...	Maple, personnage central de BOFURI, est une j...	PT2M57S	783	4	NaN	33
705	YfW4Tu-hjGI	Selph1ne	2021-12-04T12:30:07Z	{'publishedAt': '2021-12-04T12:30:07Z', 'chann...	Et oui, moins chère que pas chère c'est moé !\...	PT2M40S	485	none	NaN	9
706	QHNmKMYSw8c	Selph1ne	2021-12-01T15:00:49Z	{'publishedAt': '2021-12-01T15:00:49Z', 'chann...	Miko, personnage principale de Mieruko-chan, u...	PT3M47S	815	0	NaN	31
707	viGPi_jAonA	Selph1ne	2021-11-26T17:17:25Z	{'publishedAt': '2021-11-26T17:17:25Z', 'chann...	AUCUN SPOILER\nTu connais cette meuf	PT2M9S	766	none	NaN	31

Figure 5 - Visualisation d'une dataframe pandas

c) Bibliothèque Seaborn et Plotly

Seaborn est une bibliothèque basée sur matplotlib disponible sur Python qui permet de tracer des graphiques statistiques.

Plotly est également une librairie disponible avec Python permettant l'élaboration de graphiques. L'avantage est que ces graphiques sont interactifs et qu'il est assez simple de créer un dashboard avec dash permettant une belle visualisation des données. C'est une librairie assez simple d'utilisation pour avoir des premiers résultats satisfaisants mais dont les ressources peuvent permettre de faire des graphiques très informatifs et complexes.

2. Présentation de l'API Youtube

a) Historique

Youtube est le deuxième réseau social le plus utilisé aujourd'hui juste après Facebook en terme de nombre d'utilisateurs actifs par mois [3]. 122 millions d'utilisateurs regardent des milliards d'heures de vidéos chaque jour et 500h de contenus sont mis en ligne toutes les minutes. Avec cette quantité de données, les data analysts ont trouvé une ressource inestimable pour leurs travaux.

b) Connexion

L'API Youtube est une API accessible à partir d'un compte Google Cloud Developer [4]. Avec ce compte, il faut générer un token afin de pouvoir s'identifier au serveur à chaque nouvelle connexion pour effectuer des requêtes.

```python
# API information
api_service_name = "youtube"
api_version = "v3"

# Create an API client
youtube = googleapiclient.discovery.build(
    api_service_name, api_version, developerKey="enter_your_key_here")
```

Figure 6 - Connexion sur l'API Youtube via un script python

Il y a un quota de requêtes à respecter afin de ne pas surcharger le serveur. Chaque utilisateur a un quota de 10000 par jour et les requêtes ont des coûts différents, comme nous pouvons le voir dans la partie suivante.

c) Documentation des requêtes

Afin de se servir de cette API pour récupérer les données intéressantes, il a fallu s'intéresser à la documentation [5] afin de voir quelles étaient les requêtes que nous allions utiliser. Une requête intéressante et qui permet de tout récupérer d'un coup est la fonction *search*. Toutefois, elle est très coûteuse (100 coûts par requête) justement du fait de l'ampleur des informations qu'elle va chercher. Pour éviter de trop prendre sur les quotas, nous avons choisi d'utiliser les fonctions *channel().list*, *playlistItems().list*, *videos().list*, qui sont des requêtes de coût unitaire :

- *Channels().list* permet de récupérer les informations que l'utilisateur souhaite sur les chaînes correspondantes à l'un des identifiants en entrée. Ainsi, il est possible de rentrer une liste d'identifiants et la requête renvoie un dictionnaire avec les informations de chaque chaîne. Il est également possible de récupérer les statistiques : le nombre d'abonnés, le nombre de vidéos. La date de création, l'url pour l'image de profil, la description de la chaîne sont accessibles via cette requête. Très important pour la suite, il est également possible de récupérer l'identifiant de la playlist par défaut : il s'agit de celle contenant l'ensemble des vidéos publiées.
- *PlaylistItems().list* permet de récupérer les identifiants des vidéos dans la playlist dont l'utilisateur a fourni l'identifiant.
- *Videos().list* récupère les informations des vidéos dont l'utilisateur de l'API a fourni l'identifiant en entrée. Il s'agit encore d'informations statistiques (nombre de vues, nombre de likes / dislikes), nombre de commentaires), date de publication, est-ce qu'il s'agit d'un livre, les tags associés, le titre, la description, la catégorie et d'autres.

3. Présentation de l'API Twitch

a) Brève description de la plateforme Twitch

Twitch est une plate-forme américaine qui permet à des viewers de regarder des vidéos de leurs streamers favoris. Twitch est une des plateformes les plus pondérantes en termes de temps de visionnage des plateformes de contenus en direct. Les streamers partagent du contenu

gaming avec une possibilité de sauvegarder ces lives sous formes de vidéos stockées sur sa chaîne.

b) Connexion

L'API Twitch est une API REST accessible après avoir enregistré votre application auprès de Twitch. Après s'être enregistré, il faut générer un token afin de pouvoir s'identifier au serveur à chaque nouvelle connexion pour effectuer des requêtes.

Pour pouvoir récupérer le token, il faut tout d'abord renseigner les codes client et secret récupérable via la plateforme Twitch :

Codes de connexion

```
Client_ID = "uffof989s0jlx0rsqltxfd9twmhwbs"
Secret    = "byddoub52je363dea4ntqnueqs4qlu"
```

Figure 7 - Identifiant de connexion

Nous avons ensuite créé une fonction permettant d'effectuer une requête get selon une structure détaillé dans la documentation de l'API Twitch :

Récupération du token :

```
# Defines the function which allows to recover the token necessary to the communication with the A
PI
def GetToken(Client_ID):
    authURL = "https://id.twitch.tv/oauth2/token"
    AutParams = {"client_id": Client_ID,
                 "client_secret": Secret,
                 "grant_type": "client_credentials",
                 "scope": "channel:read:subscriptions"
                 }
    AutCall = requests.post(url=authURL, params=AutParams)
    access_token = AutCall.json()["access_token"]
    return access_token
```

```
access_token = GetToken(Client_ID)
```

Figure 8 - Script permettant la récupération d'un token

Il faut savoir qu'à partir d'un certain nombre de requêtes, il est nécessaire de réinitialiser le token. Ce quota de requêtes permet de respecter afin de ne pas surcharger le serveur.

c) Documentation des requêtes

Afin de se servir de cette API pour récupérer les données intéressantes, il a fallu s'intéresser à la documentation afin de voir quelles étaient les requêtes que nous allions utiliser. Pour éviter de trop prendre sur les quotas, nous avons choisi d'utiliser les fonctions suivantes :

- *Get Videos* qui permet d'obtenir des informations vidéo par un ou plusieurs ID vidéo, ID utilisateur ou ID de jeu. Pour la recherche par utilisateur ou par jeu, plusieurs filtres sont disponibles et peuvent être spécifiés comme paramètres de requête et sont détaillés dans la documentation.
- *Get Channels* récupère les informations liées à un channel particulier comme ID utilisateur Twitch du propriétaire de cette chaîne qui est nécessaire pour le Get Video, le nom du jeu en cours de lecture sur la chaîne.

B. Récupération des données

1. Objectifs

Les données que nous souhaitons récupérer sont de plusieurs natures. Nous souhaitons avoir d'un côté l'ensemble des chaînes Youtube et Twitch des VTubers francophones et d'un autre côté l'ensemble des vidéos associées à chacune de ces chaînes.

2. Récupération des identifiants Youtube

Le nombre de VTuber français est limité mais récupérer les identifiants manuellement serait trop fastidieux. Nous avons alors eu l'idée de faire un programme de *scraping*. Nous avions effectivement connaissance d'un site de fandom [6] qui recense un certain nombre de VTubers francophones comme nous pouvons le voir sur l'image ci-dessous.

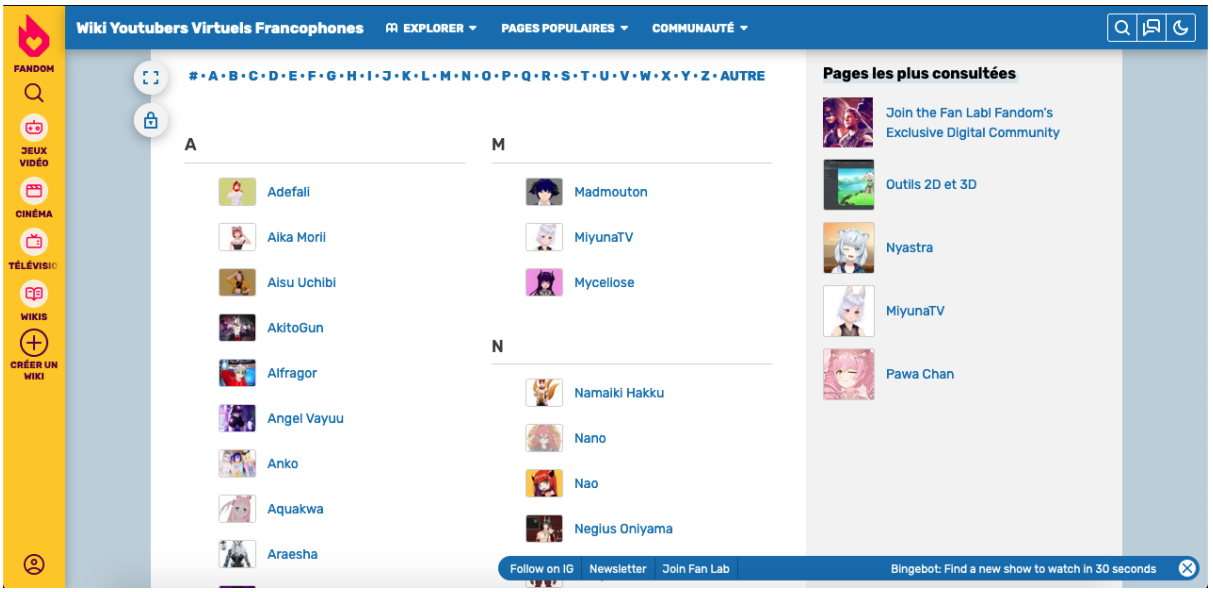

Figure 9 - Page d'accueil du site fandom des VTubers francophones avec un annuaire

La page d'accueil est un annuaire des VTubers et si l'utilisateur clique sur un nom, le site nous redirige vers la fiche de ce VTuber qui contient l'ensemble des informations. Parmi ces informations, ce qui nous intéresse est la partie médiatique qui nous donne l'ensemble des liens redirigeant vers les plateformes de réseaux sociaux utilisées par le VTuber :

Media	
Site officiel	https://vtuber-fr.wixsite.com/novel
Twitch	https://www.twitch.tv/selph1ne_nvl
Twitter	https://twitter.com/Selph1ne
Youtube	https://www.youtube.com/channel/UCMDzOuBszEjNub73uHgiVVw

Figure 10 - Données que nous souhaitons récupérer sur la page d'un VTuber particulier

En regardant le fichier html et en le parsant grâce à la bibliothèque Python BeautifulSoup, il est possible de sortir de manière automatique le lien Youtube sur le page, s'il existe (car il peut ne pas être renseigné, ou peut même ne pas exister). Ayant ce lien, il est possible de sélectionner l'identifiant qui correspond à la partie après le *https://www.youtube.com/channel/*. Ce faisant, nous avons incrémenté une liste contenant l'ensemble des identifiants Youtube, qui servira lors des requêtes comme expliqué dans la partie IV.B.1.

3. API Youtube/Twitch

Une fois les identifiants des VTubers obtenus, il est possible d'utiliser l'API Youtube pour faire les requêtes sur les identifiants et récupérer les informations souhaitées. Comme mentionné dans la partie sur la documentation de l'API Youtube (IV.A.2.c), il n'est pas possible de tout récupérer d'un coup. Il a fallu faire une première requête pour récupérer les chaînes correspondantes aux identifiants. Dans un deuxième temps, il a fallu extraire pour chaque chaîne l'identifiant de sa playlist par défaut (celle contenant l'ensemble des vidéos publiées). A partir de cette information, il est possible d'aller chercher les identifiants des vidéos qui la composent. Enfin, il est possible de faire les requêtes sur les vidéos. En fin de compte, nous avons les jeux de données suivants :

- La liste des identifiants des VTubers
- La dataframe des chaînes Youtube et Twitch
- La liste des identifiants des vidéos
- La dataframe de toutes les vidéos

C. Nettoyage des données

Il a fallu nettoyer les données obtenues afin d'avoir un ensemble homogène, exploitable. Le travail à effectuer ne concernait pas vraiment les dataframes obtenues après requêtes. Le nettoyage se trouvait surtout au niveau de la liste de jeux vidéo existants qui a été obtenue via l'API Twitch et sur la liste de VTubers obtenues par l'API Youtube. Pour la première, il y avait quelques incohérences, avec des jeux vidéo qui s'appelaient "A", "b", "O", "Music", "Live", etc. Ce ne sont pas des noms de jeux vidéo et étant donné leur fréquence d'apparition dans les descriptions des vidéos, cela biaisait les résultats quand nous avons voulu voir quels étaient les jeux vidéos les plus joués par VTuber.
Dans le deuxième cas, il y avait quelques chaînes qui ne correspondaient pas à ce que nous voulions. Ce n'étaient pas des chaînes francophones ou plus rarement, ce n'étaient pas des chaînes de VTubers.

V. Analyse de données

A. Stratégie d'analyse

Après avoir pris en main les outils qui nous seraient nécessaires, nous avons pu établir des graphiques que nous aimerions voir, des patterns à observer. Nous avons abordé l'analyse sous plusieurs angles. Le premier était de faire une première analyse globale, sur l'ensemble des VTubers : combien sont-ils ? Sont-ils actifs en France ? Sont-ils connus ?
Le deuxième angle était de faire une analyse spécifique à un VTuber : publie t-il à des jours fixes ? Quels sont les jeux joués dans ses vidéos ? **Est-il plus sur Youtube ou sur Twitch** ? Avec quels autres VTubers est-il lié ?

B. Analyse sur Youtube

1. Elaboration de graphiques sur l'ensemble des VTubers

En fin de compte, nous avons récupéré 121 identifiants de chaînes de VTubing sur Youtube. Pour faire une analyse générale sur les VTubers, nous avons décidé de ne sélectionner que les 20 plus connus (en se basant sur le critère du nombre d'abonnés).

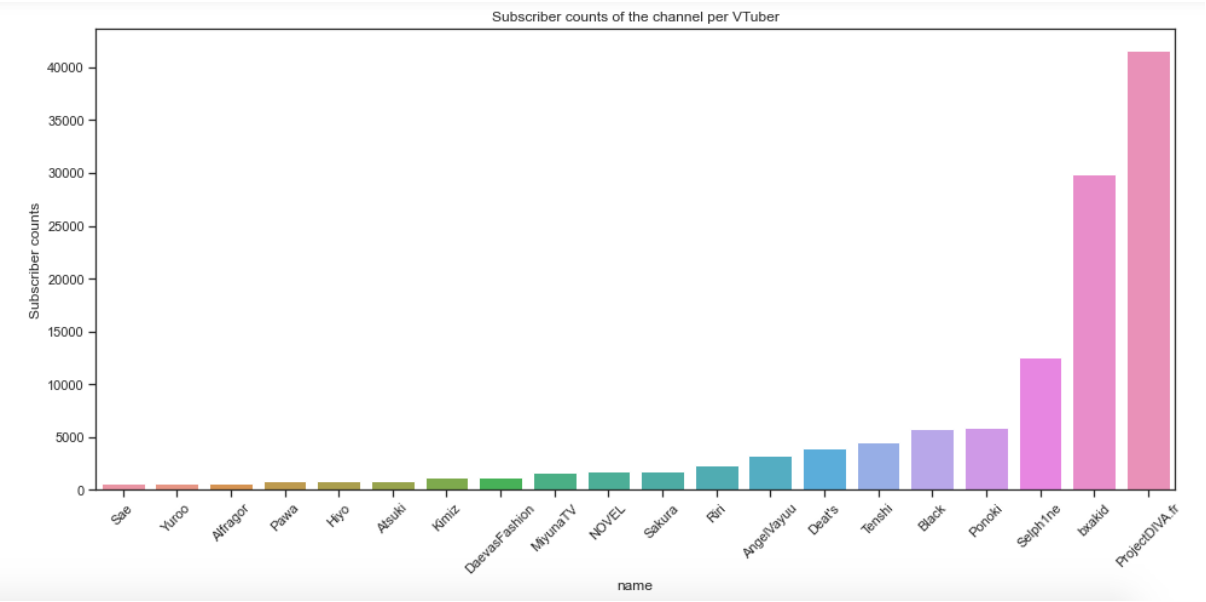

Figure 11 - Nombre d'abonnés par chaîne de VTuber sur Youtube

Nous pouvons alors voir sur le graphique de la figure 11 une répartition très inégale de popularité au sein de la communauté des VTubers. Un il y a une chaîne *ProjectDIVA.fr* qui est un groupe de VTubers qui sont déjà assez connus sur Youtube. Nous retrouvons aussi *bxakid* qui n'est de base pas un VTuber mais qui s'est lancé dans cette activité tardivement sur sa chaîne, ce qui explique ses 29.900 abonnés. Nous pouvons dire que *Selph1ne* est le VTuber le plus connu en France avec ses 12.500 abonnés.

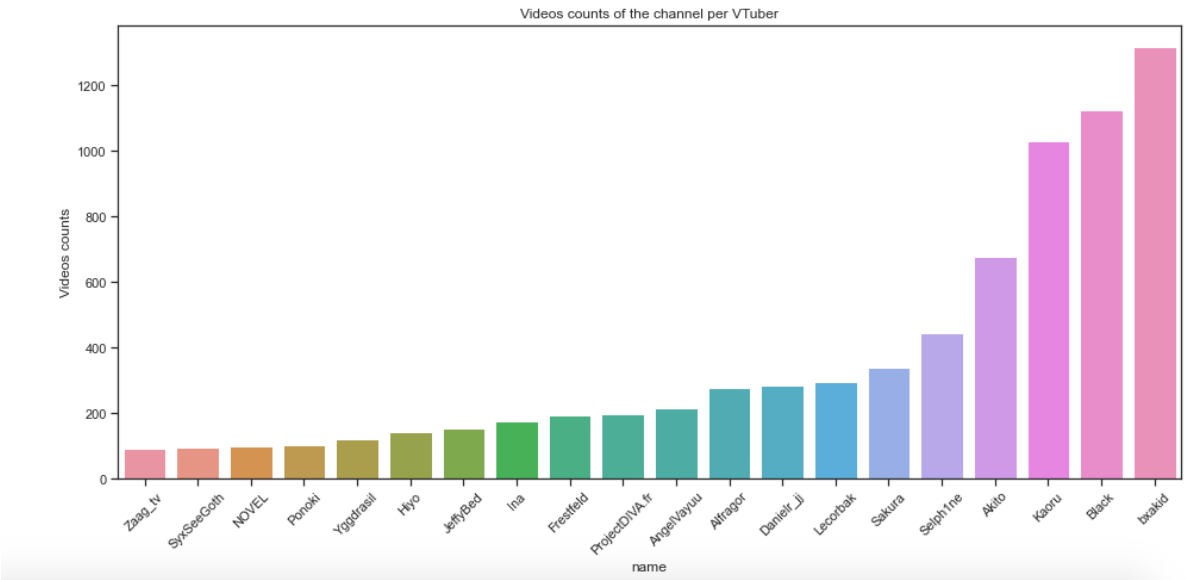

Figure 12 - Nombre de vidéos publiées par les VTubers francophones sur Youtube

Nous avons également souhaité faire un classement des VTubers selon leur nombre de vidéos. L'idée était initialement de voir si les plus populaires étaient aussi les plus forts créateurs de contenus. Nous retrouvons donc *bxakid* en tête de classement, *Selph1ne* n'arrive qu'en n°5 du classement et nous avons avant *Black*, *Kaoru* et *Akito*. *Kaoru* et *Akito* n'ont pas spécialement beaucoup d'abonnés (respectivement 247 et 349) mais publient beaucoup de contenus sur Youtube (respectivement 1030 et 676). Il est également intéressant de regarder les dates de création des chaînes. Par exemple, *Kaoru* existe depuis 2014 et Akito depuis 2007. Dans les deux cas, les chaînes ont été créées avant l'arrivée du phénomène VTuber en France. Après étude de leur chaîne respective, nous pouvons voir qu'initialement, leurs vidéos étaient des gameplay de jeux et que leur avatar est intervenu plus tard.
Nous pouvons donc constater que les VTubers qui publient le plus ne sont pas nécessairement les plus connus (même s'ils ont quand même une petite communauté). En revanche, les VTubers les plus connus publient également beaucoup.

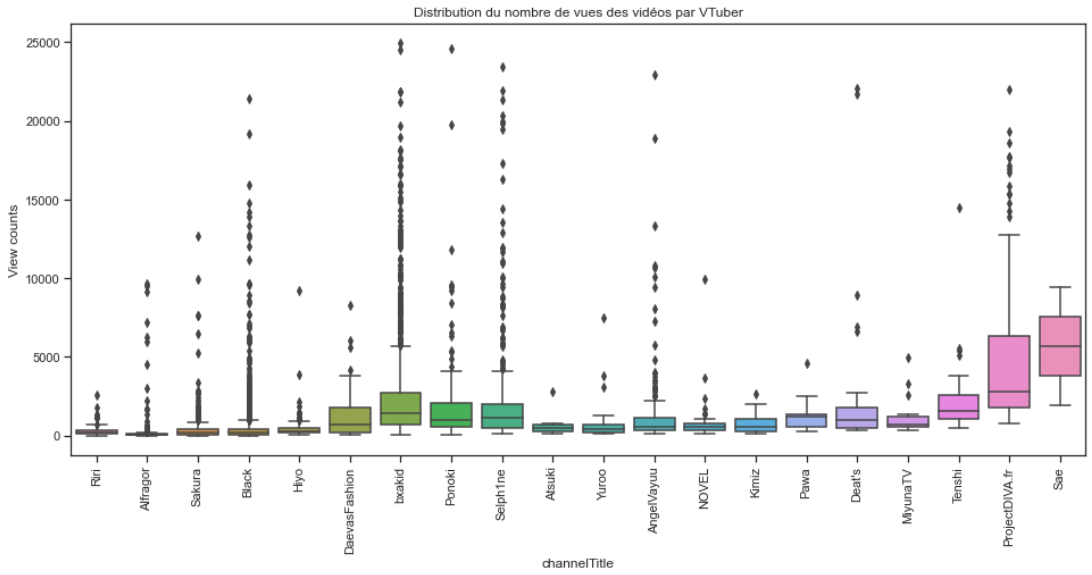

Figure 13 - Distribution du nombre de vues des vidéos par VTuber francophone sur Youtube

Nous avons aussi élaboré le graphique de la figure 13 montrant la distribution du nombre de vues des vidéos par VTuber. Notre motivation était de voir s'il y avait une distribution uniforme selon les chaînes Youtube, ou au contraire de voir s'il y avait quelques fois des *outliers*, des vidéos qui font le buzz chez certains VTubers. Nous remarquons que globalement, outre *ProjectDIVA.fr* et *Sae*, la majorité des vidéos de VTubers sont vues par un nombre assez homogène de personnes. Nous pouvons alors supposer que le public regardant ces vidéos est le même et qu'il s'agit d'un public de connaisseur qui va chercher des contenus similaires. La communauté francophone de VTubers n'est pas très grande, il peut être assez rapide de faire le tour.

Nous remarquons cependant que certaines chaînes rencontrent plus de succès sur leurs vidéos que d'autres. Ce sont les mêmes chaînes que nous avions identifiées comme étant les plus populaires auparavant : *bxakid*, *Selph1ne*, *ProjectDIVA.fr* avec des vidéos qui peuvent être vues jusqu'à 25000 fois. Il convient de notifier que lors de l'élaboration du graphique, nous avons filtré les vidéos pour ne garder que celles qui avaient été vues moins de 25000 fois afin de préserver la lisibilité du graphique.

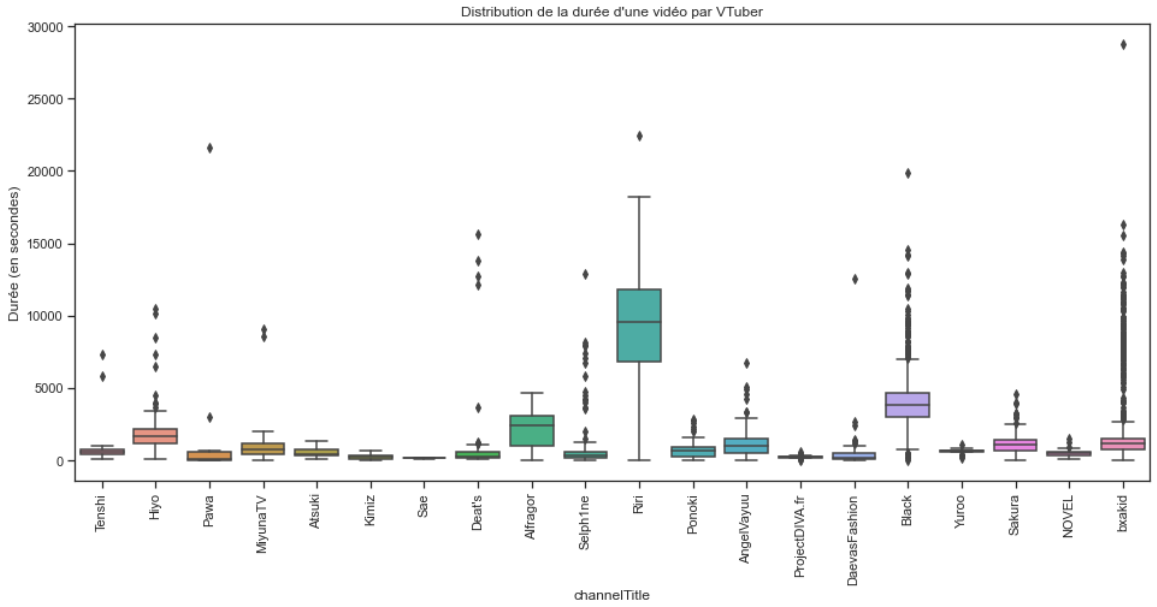

Figure 14 - Distribution de la durée des vidéos par VTuber francophone sur Youtube

Enfin, pour finir cette analyse globale sur la tendance générale au sein des VTubers, nous avons sorti le graphique de la figure 14, qui montre la distribution de la durée d'une vidéo par VTuber. Afin de mieux voir, nous avons également fait le graphique de la figure 15.

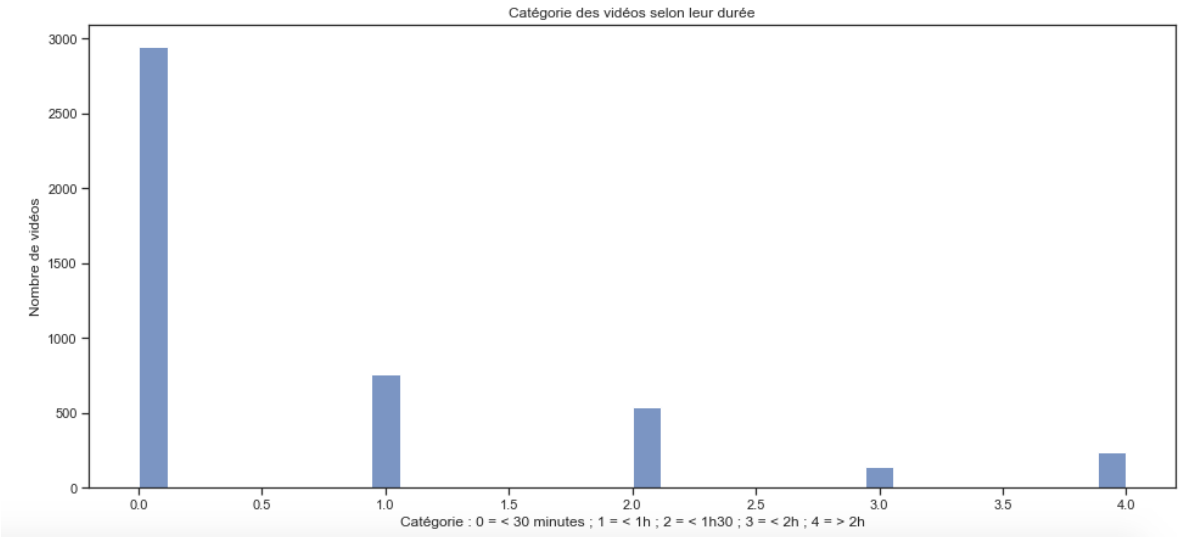

Figure 15 - Histogramme représentant le nombre de vidéos dans chacune des catégories de durée. **Catégorie 0** : la vidéo dure moins de 30 minutes. **Catégorie 1** : la vidéo dure entre 30 minutes et 1h. **Catégorie 2** : la vidéo dure entre 1h et 1h30. **Catégorie 3** : la vidéo dure entre 1h30 et 2h. Catégorie 4 : la vidéo dure plus de 2h.

Nous pouvons alors plus facilement voir que la grande majorité des vidéos durent moins de 30 minutes. Il y a cependant un nombre non négligeable dans les autres catégories. Les VTuber ont l'air de faire des streams sur leur chaîne Youtube, comme par exemple Aquakwa dont la durée

moyenne des vidéos est assez élevée (même s'il y a visiblement une grande disparité avec des vidéos très courtes de l'ordre de quelques secondes jusqu'à 5h35).

2. Elaboration de graphiques spécifiques à un VTuber

Nous avons pensé qu'il serait également intéressant de faire une analyse spécifique à chaque VTuber afin de voir s'ils avaient des comportements particuliers.

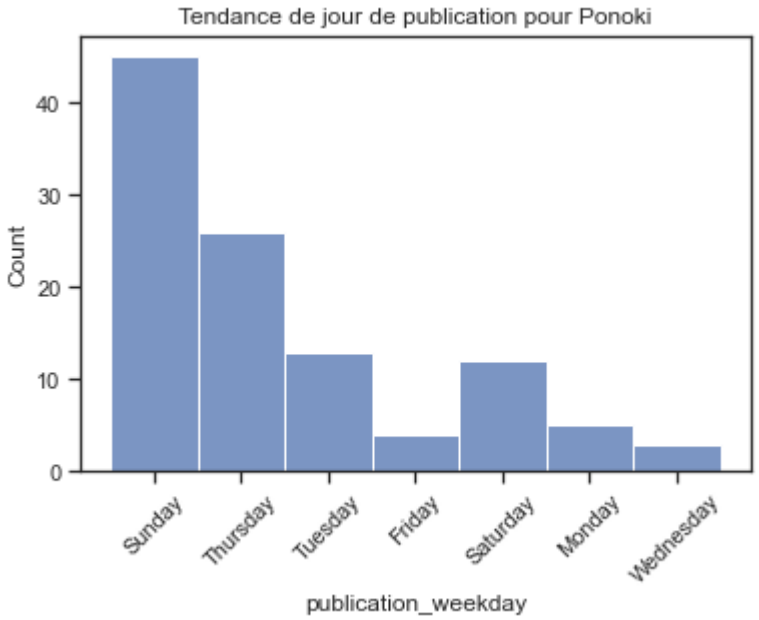

Figure 16 - Tendance des jours de publication pour le VTuber Ponoki sur Youtube

Sur le graphique ci-dessus, nous avons souhaité voir s'il y avait un pattern au niveau des jours de publication des vidéos sur Youtube. Sur les chaînes et les vidéos, quelques fois est indiqué en description que les vidéos sont publiées tels jours à tels horaires. C'est cette information que l'on veut voir sur le graphique. Nous avons pris l'exemple de la VTubeuse *Ponoki Chan* pour notre analyse. Nous pouvons ainsi voir que la majorité des vidéos ont été publiées le dimanche et une autre bonne partie le vendredi. Cette analyse n'est valable que si le VTubeur en question a un nombre important de vidéos.

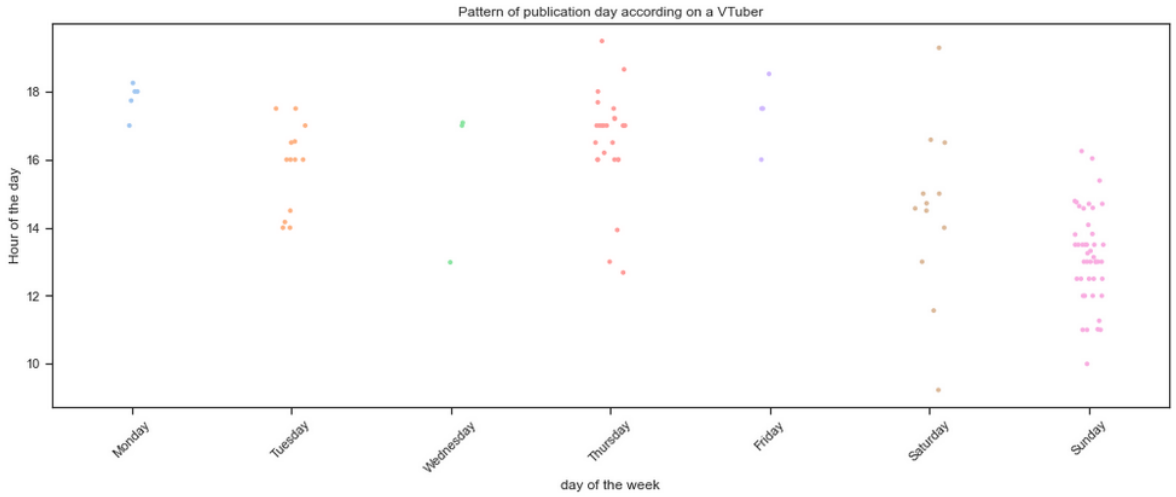

Figure 17 - Tendance des heures de publication par jour des vidéos des VTubers francophones sur Youtube

Nous avons aussi élaboré le graphique ci-dessus afin d'avoir encore plus de détails sur les habitudes de publication. Nous pouvons ainsi voir les horaires de publication des vidéos par jour. Chaque point correspond à une vidéo.

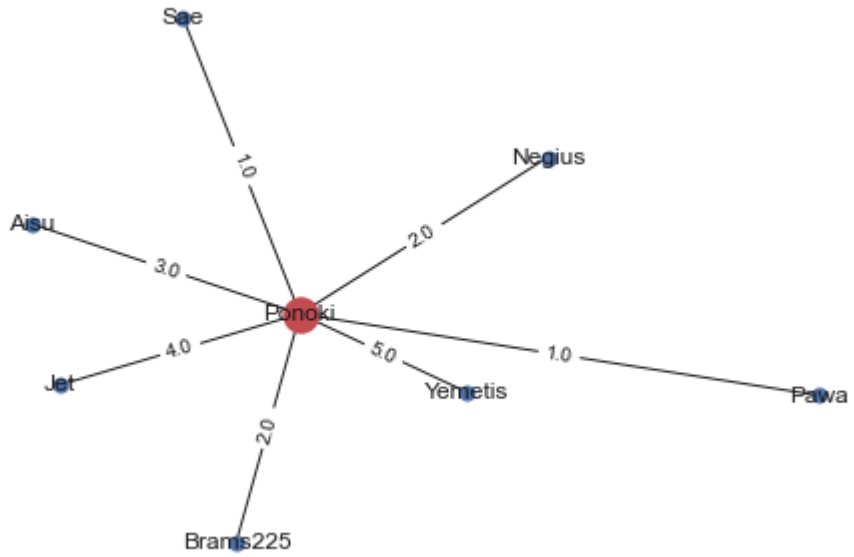

Figure 18 - Graphe social entre Ponoki et les VTubeurs francophones sur Youtube

Enfin, un graphique qui nous a semblé important à faire a été d'élaborer un graphe social. Le but ici était de voir les liens existants entre les VTubeurs de notre base de données comme des collaborations entre eux, etc. Grâce au graphe, nous pouvons également représenter la force des liens. Au centre, nous retrouvons le VTubeur dont les liens sont analysés.

C. Analyse sur Twitch

1. Elaboration de graphiques

Sur le même principe décrit dans les paragraphes précédents, sur une base de VTubers actifs recensés sur 10j, nous choisissons d'étudier ici les 20 VTubers les plus regardés.

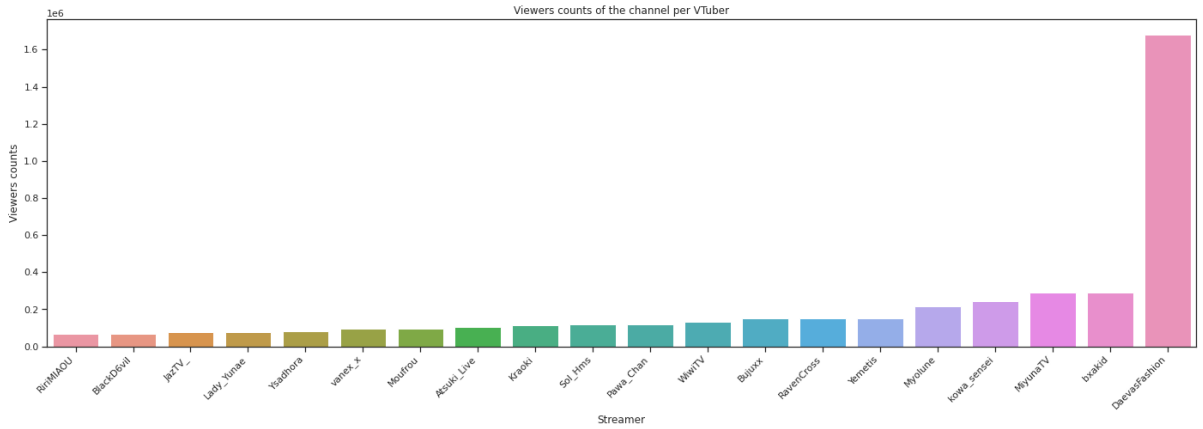

Figure 19 - Nombre de vues cumulées par chaîne de VTuber sur Twitch

La figure 19 montre le nombre de vues cumulées en fonction du streamer observé. On peut également voir sur le graphique une répartition très inégale de popularité au sein de la communauté des VTubers. Ici, DaevasFashion est la chaîne qui regroupe le plus de vues sur notre échantillon de streamer. Cette chaîne appartient à une VTubeuse streamant principalement du contenu de jeux multijoueurs.

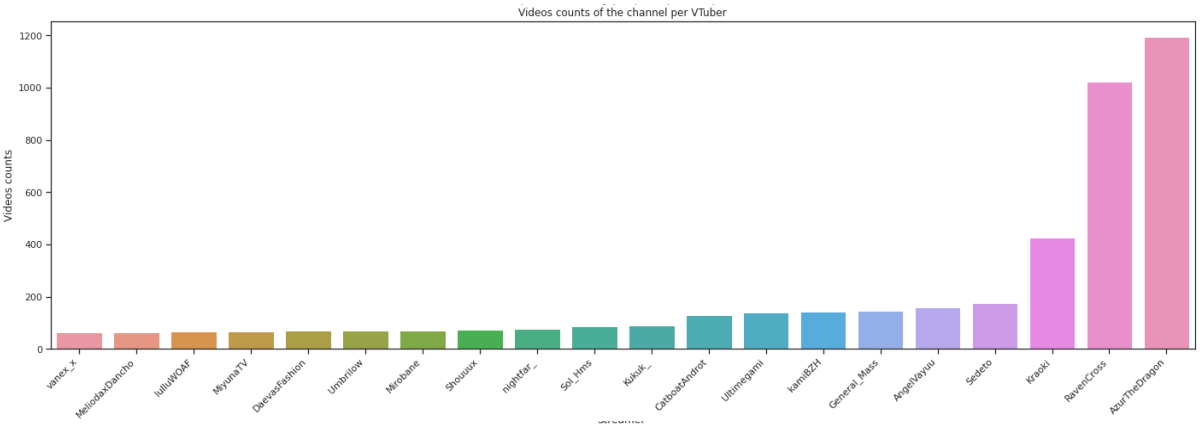

Figure 20 - Nombre de vidéos publiées par les VTubers francophones sur Twitch

La figure 20 met en évidence le nombre de vidéos mis en ligne en fonction des streamers. Dans la même idée, ici il est question d'observer si une corrélation entre streamer populaire et plus gros créateur de contenu existait. On peut observer qu'il ressort trois grands noms de streamer publiant de nombreuses vidéos. Il y a en première place AzurTheDragon, qui est un joueur

principalement de RPG, mais il diversifie également son contenu avec du storytelling et de chant. Ensuite, nous avons RavenCross qui a le même profil qu'AzurTheDragon. Finalement, nous avons Kraoki qui n'est pas un VTuber. En effet, cette personne a fait une vidéo avec un avatar mais le contenu principal de ce streamer ne correspond pas au profil d'un VTuber.
On arrive à la même conclusion que Youtube, on peut conclure sur le fait que les VTubers qui publient le plus ne sont pas nécessairement les plus connus.

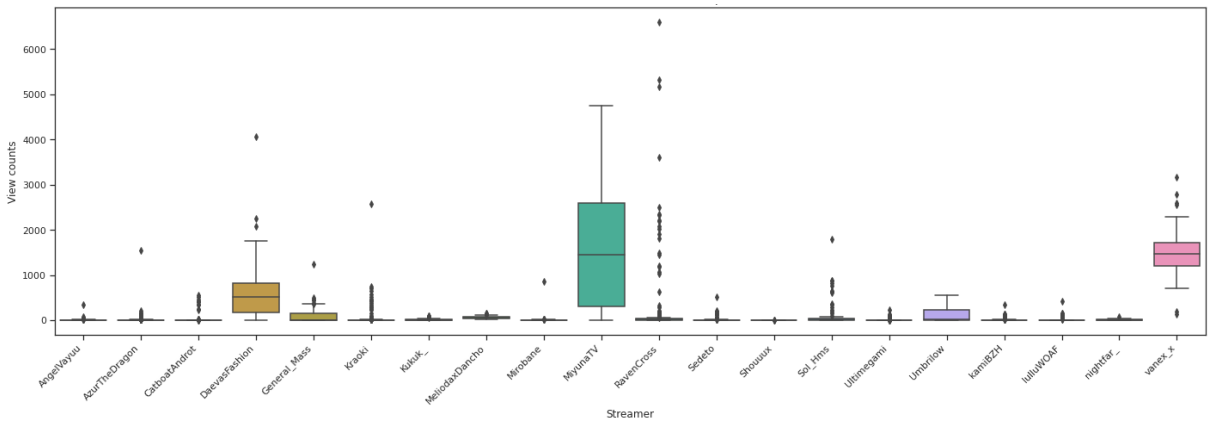

Figure 21 - Distribution du nombre de vues des vidéos par VTuber francophone sur Twitch

La figure 21 montre la distribution du nombre de vues des vidéos par VTuber. On peut voir ici qu'à part MiyunaTV et DaevasFashion, il y a que des distributions plus ou moins uniformes. MiyunaTV est un nom qui sort du lot que dans cette figure. Ce VTuber est assez populaire mais il semblerait que ce soit une chaîne assez récente qui possède peu de vidéos sauvegardées et donc peu de vues cumulées. Ce traitement nous a permis de ressortir ce genre de chaîne.

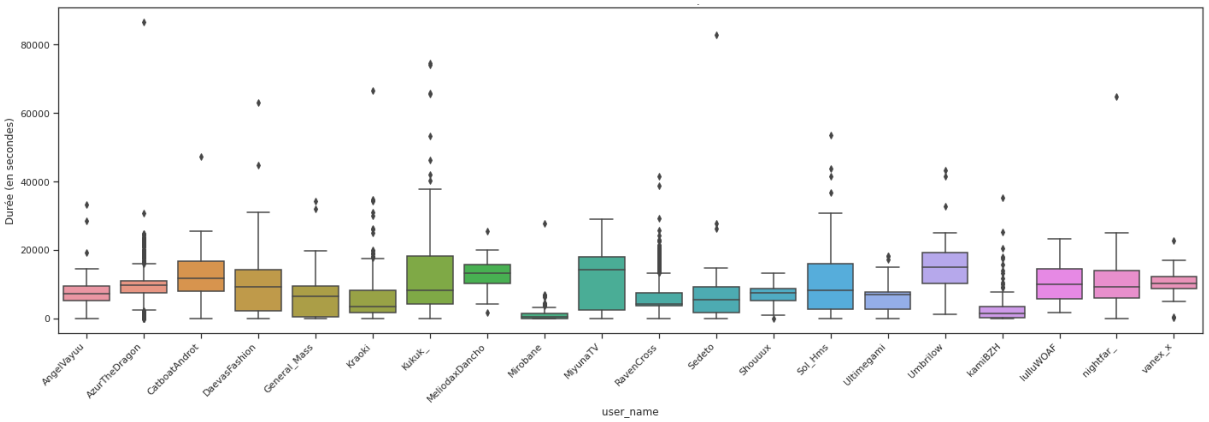

Figure 22 - Distribution de la durée des vidéos par VTuber francophone sur Twitch

Enfin, la figure 22 nous avons ce graphique, qui montre la distribution de la durée d'une vidéo par VTuber. On peut voir ici que la moyenne globale est assez uniforme et est aux alentours de 2h40. On peut voir des pics entre 20h et 24h de vidéo qui correspondent à des streamers qui réalisent des subathons.

2. Elaboration de graphiques spécifiques à un VTuber

Tout comme dans l'analyse faite via Youtube concernant l'analyse spécifique, nous avons reproduit le même type d'analyse sur les VTubers présent dans la base de données.

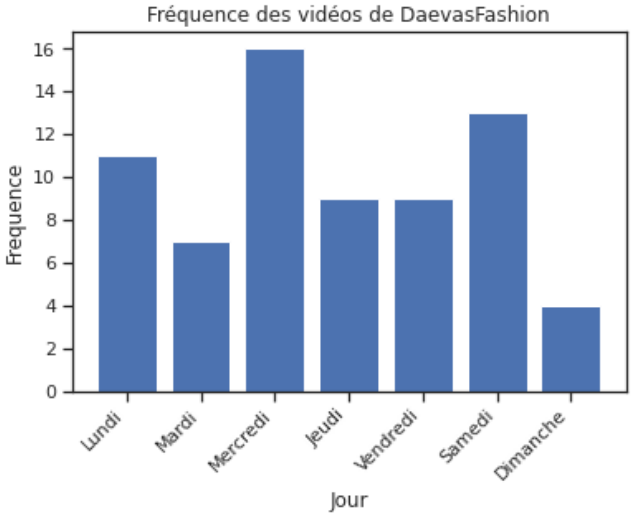

Figure 23 - Tendance des jours de publication pour le VTuber DaevasFashion sur Twitch

Sur la figure 23, il s'agit de la quantité de vidéo produite selon les jours de la semaine par DaevasFashion. Nous pouvons voir que ce streamer produit des vidéos globalement toute la semaine excepté le dimanche où il y a bien peu de vidéos.

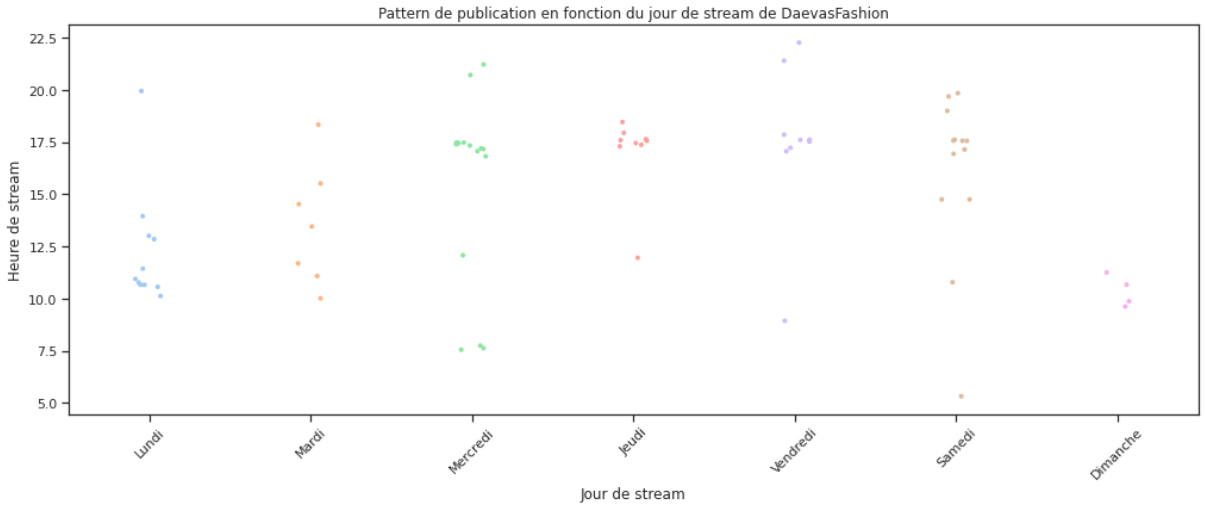

Figure 24 - Tendance des heures de publication par jour des vidéos de DaevasFashion sur Twitch

Finalement, sur la figure 24, nous avons réalisé le graphique afin d'avoir un peu plus de précisions sur les habitudes de publication. Ainsi, nous pouvons observer les horaires de publication des vidéos par jour avec pour chaque point la correspondance à une vidéo postée.

D. Mise en commun et élaboration d'un dashboard

1. But de la mise en commun

Comme mentionné auparavant, nous avons travaillé séparément sur les API Youtube et Twitch afin de récupérer les données et les analyser suivant une même stratégie. Une fois cette étape faite, nous pouvions nous concentrer sur l'objectif initial qui était de mettre en commun nos analyses. Ce peut être très intéressant de voir pour un VTuber qui est sur les deux plateformes s'il a un comportement complémentaire, si l'utilisation de Youtube est différente de Twitch. On peut également voir si un VTuber est plus orienté Twitch que Youtube. Enfin, de manière globale, on peut réussir à voir si ce phénomène de VTuber en France est plus populaire sur Youtube ou Twitch.

2. Elaboration du dashboard et visualisation

Afin de visualiser les données et l'analyse qu'on en a faite, nous avons décidé d'élaborer un dashboard au lieu de rester sur un notebook. Cela rend la visualisation plus attractive et plus interactive. Le dashboard était initialement concentré sur les analyses Youtube. Certaines données et analyses Twitch ont été intégrées mais un travail supplémentaire pourrait être fait pour continuer cette démarche. Le dashboard a été fait en utilisant la bibliothèque dash de plotly.

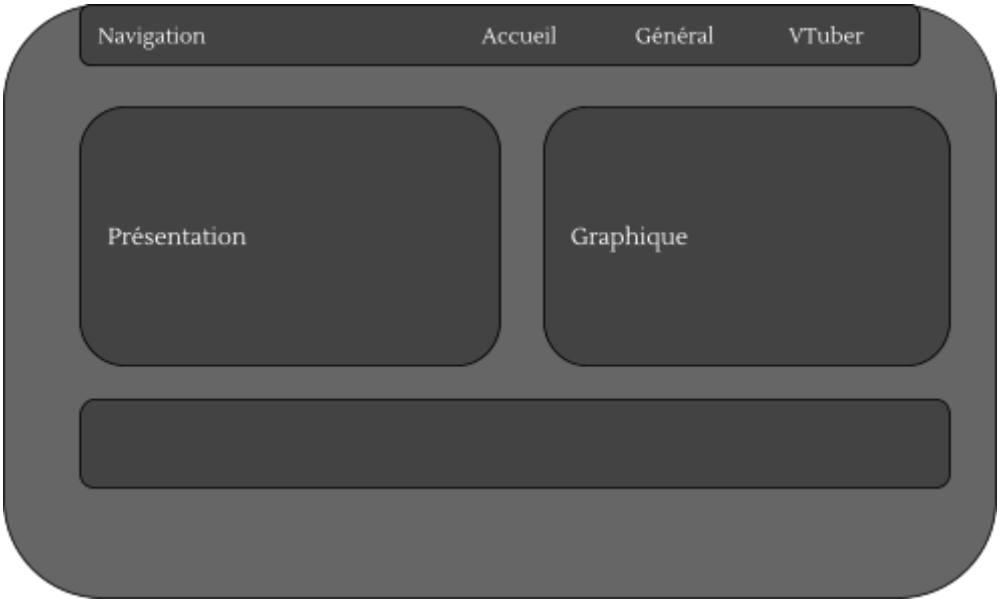

Fig 25 : maquette du dashboard

Nous avons pensé le dashboard comme montré sur la figure 25. Une page de présentation (figure 26) introduit le sujet pour expliquer ce qu'est le VTubing, montre un graphique d'évolution du

nombre de VTubeur au cours du temps, montrant l'émergence de ce nouveau phénomène qui prend de l'ampleur. Cette page contient aussi des références.
Pour les analyses, à l'image de notre stratégie, nous avons fait deux onglets différents afin d'accéder à une analyse globale (figure 28) et à une analyse spécifique à chaque VTuber (figure 29).
Pour plus de clarté dans le code, nous avons écrit un script par page. Ces pages sont regroupées dans un module qui est importé dans le fichier *index.py* qui fait le lien dans tout le dashboard.

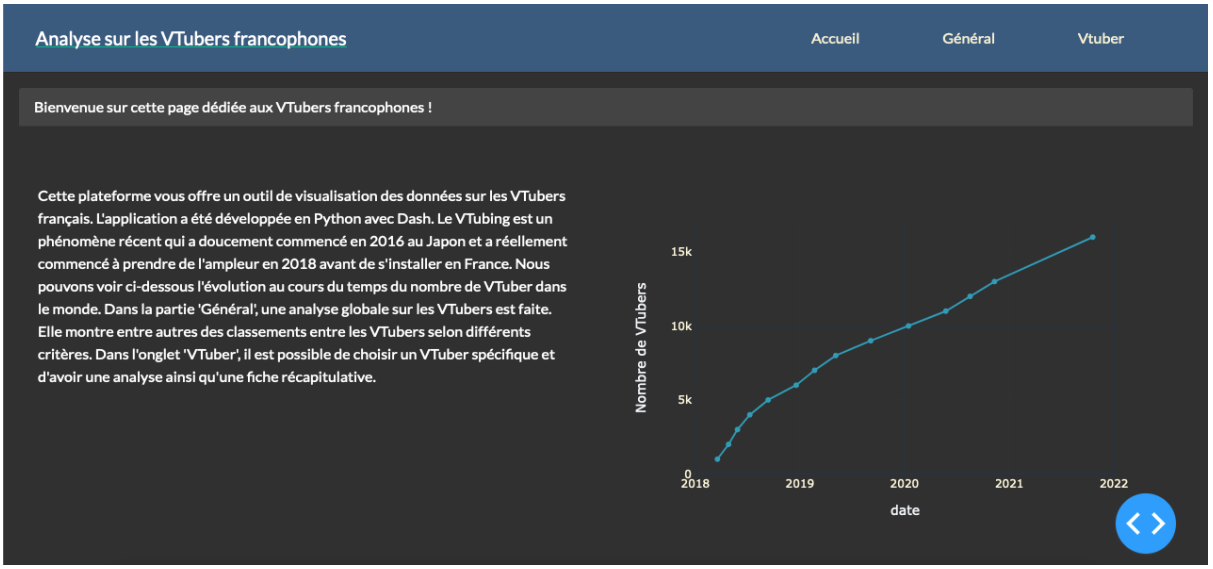

Figure 26 : Page d'accueil du dashboard

Quand on charge le dashboard en local depuis une ligne de commande, on arrive directement sur la page d'accueil. Pour naviguer, il suffit de cliquer sur les onglets comme présenté sur la figure 27.

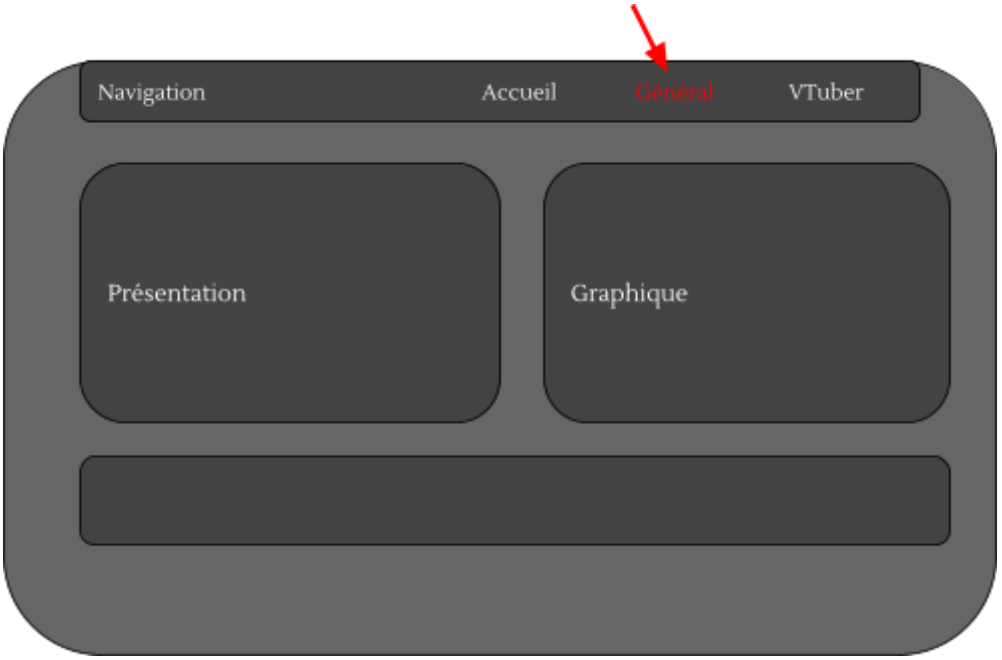

Figure 27 : indication d'utilisation - navigation

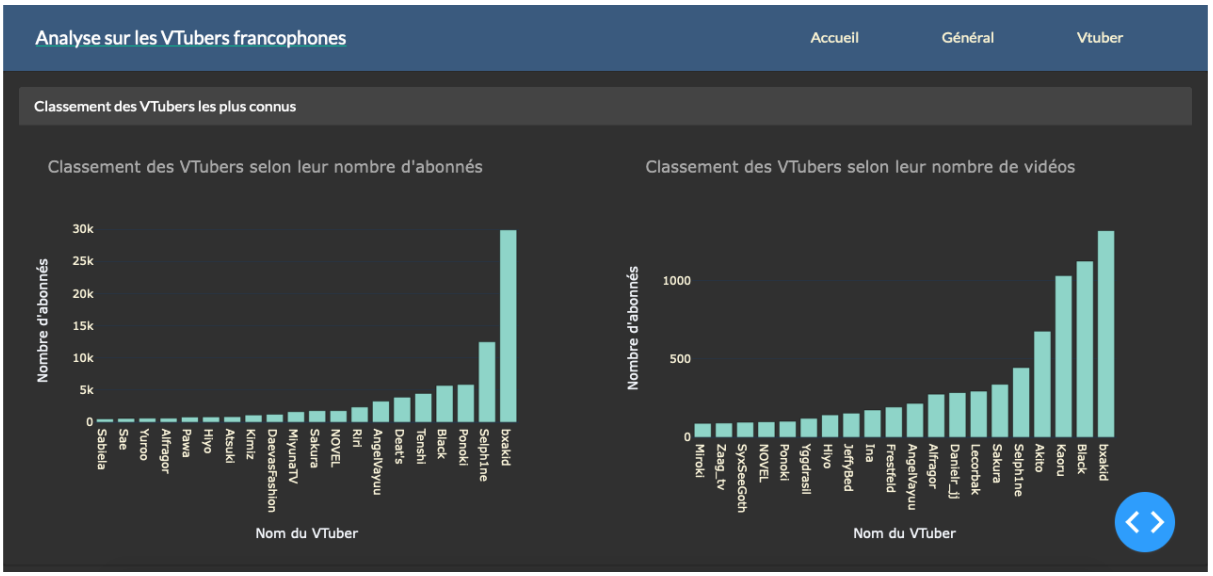

Figure 28 : Page d'analyse globale

On peut également cliquer sur l'onglet *Vtuber* afin d'accéder à l'analyse spécifique du VTuber de notre choix. Par défaut, l'analyse présentée est celle du VTuber *Selph1ne*. Le temps de chargement est plus ou moins long selon le nombre de vidéos à analyser. Par exemple, pour l'exemple du VTubeur *Selph1ne* que l'on peut voir figure 29, l'affichage met 43 secondes. Cela est dû à une recherche sur toutes les vidéos de mots-clés pour connaître les jeux vidéos apparaissant le plus sur la chaîne. Il y a donc 444 vidéos et pour chacune, on recherche si au moins 10000 mots appartiennent à la description de la vidéo : cela représente beaucoup de traitement. Pour réduire le temps, le calcul pourrait être fait en amont et être sauvegardé dans une base de données. Pour *Ponoki*, le temps d'affichage pour analyser 102 vidéos est de 11 secondes.

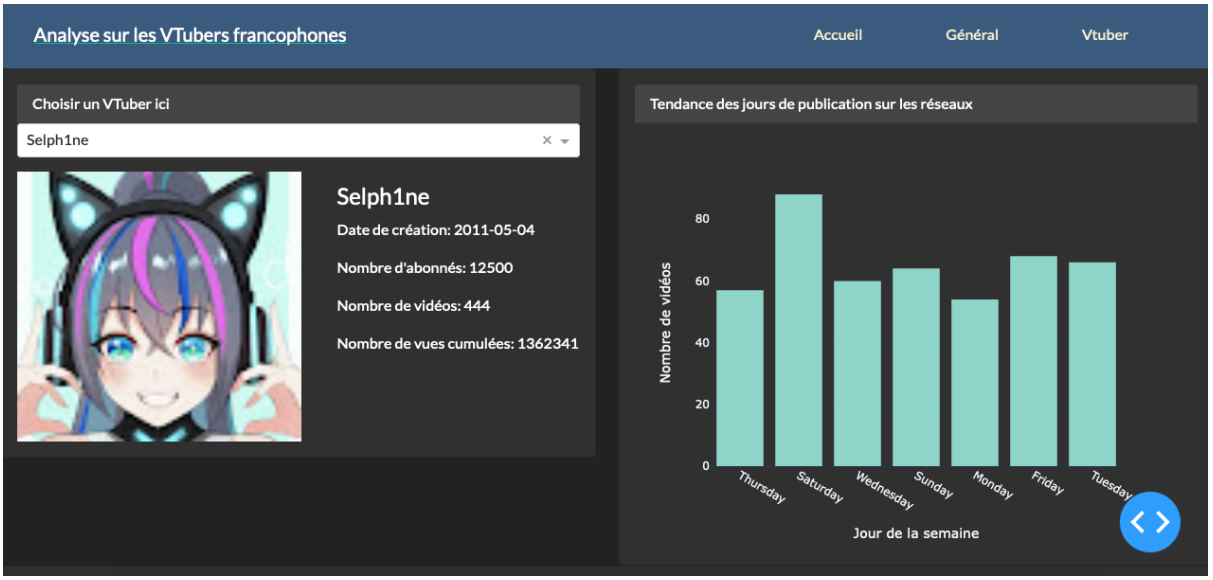

Figure 29 : Page pour l'analyse d'un VTuber

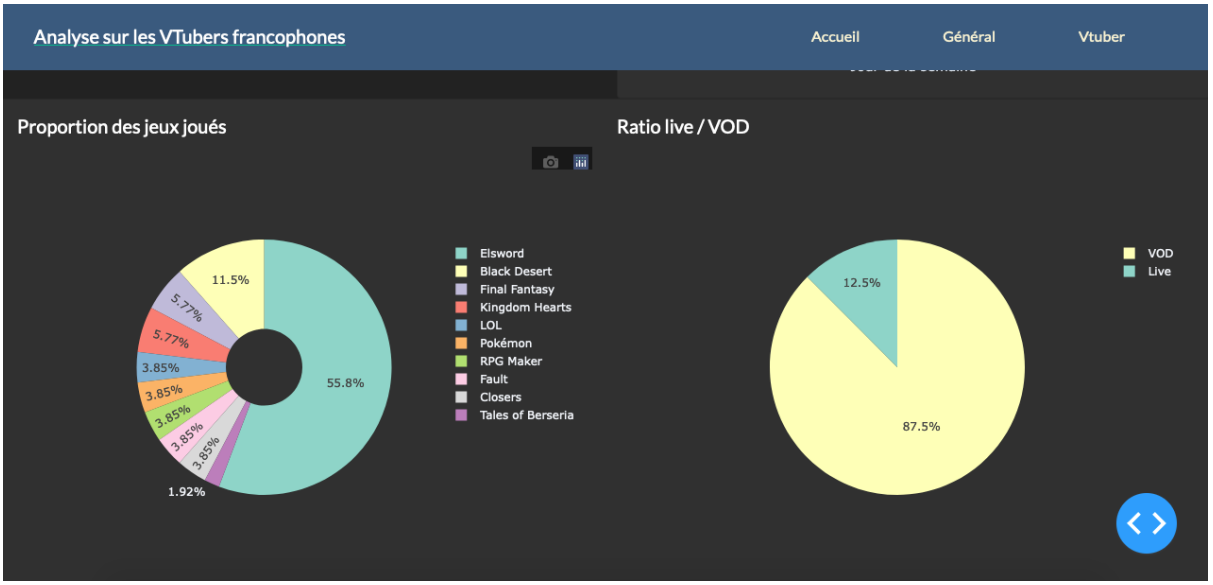

Figure 30 : Page pour l'analyse d'un VTuber

3. Analyse comparative

Nous avons décidé de comparer plusieurs choses entre les deux plateformes :

- Les informations de la chaîne :
  - Date de création de la chaîne
  - Nombre d'abonnés
  - Nombre de vidéos
  - Nombre de vues cumulées sur les vidéos
- Tendance des jours de publication

Nous pouvons voir une comparaison pour le VTubeur Aquakwa.

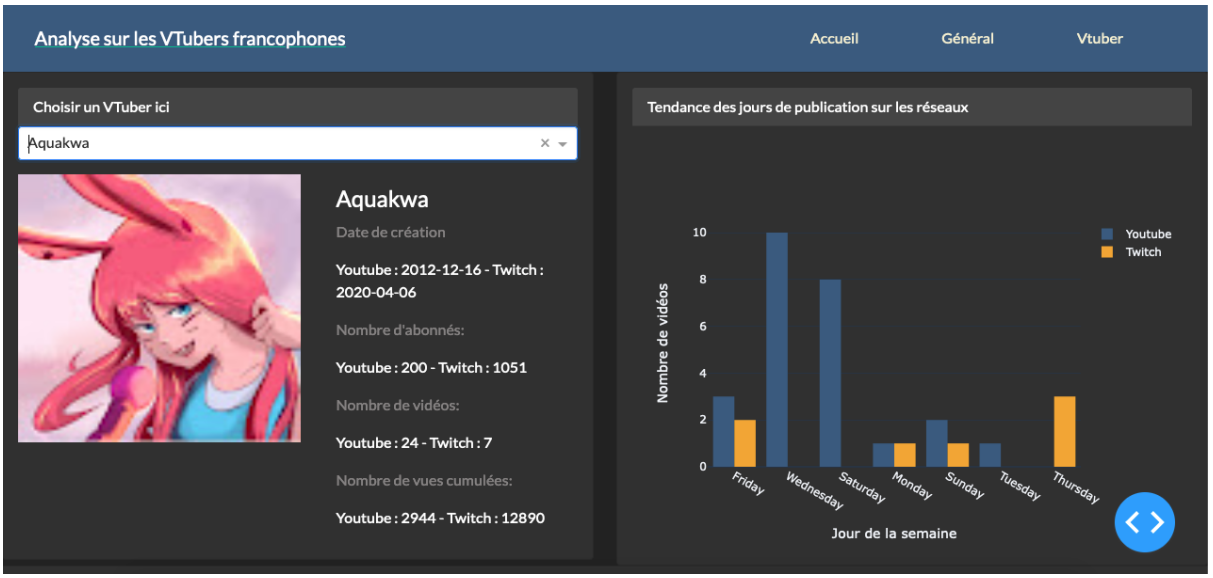

Figure 31 : Comparaison des données Youtube et Twitch pour Aquakwa

VI. Difficultés rencontrées

A. Apprentissage / familiarisation des outils

1. Bibliothèques de programmation

Avant de rentrer dans le vif du sujet, il a fallu se familiariser avec les outils nécessaires à l'analyse de données en Python. D'une part, Constance avait déjà des connaissances en python et en pandas, de l'autre côté, Khaled était novice. Nous avons donc effectué un tutoriel sur Kaggle afin d'apprendre les bases, ou de consolider les acquis. Cela a pris un peu de temps au début parce que nous voulions bien faire les choses, et que nous avons pris le temps de lire la partie théorique mais également de faire toutes les parties pratiques proposées dans ce tutoriel.
Une partie assez compliquée d'apprentissage de bibliothèque a été quand nous avons décidé d'élaborer un dashboard pour la visualisation des données. Pour ce faire, nous avons décidé d'utiliser dash de Plotly qui est un outil très adapté quand il s'agit de jouer avec des données.

2. API et leur documentation

Il était également nécessaire de comprendre comment fonctionnent les API. Nous nous étions répartis le travail en 2 : Constance sur Youtube, Khaled sur Twitch. Nous nous sommes alors heurtés à la documentation. La prise en main n'était pas facile, parce qu'il y avait beaucoup d'informations et qu'il n'était pas facile de voir ce qu'il nous intéressait. Du côté de l'API Youtube, nous avons pu trouver de l'aide du côté d'un article écrit sur Medium qui présentait l'API et les requêtes au travers d'exemple. Cela a permis de mieux comprendre comment cela fonctionnait.

B. Nettoyage des données

Le nettoyage de données représente une partie non négligeable du temps alloué au projet. Comme nous l'avons mentionné avant (IV.C), le plus nettoyage était sur la liste de jeux vidéo et sur la liste de VTubers récupérée sur l'API. Le travail a été fait manuellement.
En ce qui concerne la liste de VTubers, nous avons pris le dataframe et nous avons parcouru ligne par ligne pour aller chercher l'identifiant et regarder sur Youtube s'il s'agissait bien d'une chaîne de VTuber francophone.
Pour la liste de jeux vidéo, il a également fallu parcourir ligne par ligne et regarder si le nom du jeu était cohérent, et dans le doute, vérifier sur internet s'il existait bien.

VII. Discussion

Dans cette partie, il est question de regrouper les possibles évolutions du travail fourni lors de ce projet. Nous avons répertorié 2 grand axes d'évolutions :

- Dans la même idée que socialblade.com, pouvoir mettre le dashboard online permettant de sortir des statistiques sur les VTubers

- Possibilité d'aller plus loin dans l'analyse, notamment sur les points suivants:
    - Liens entre Vtubers
    - Analyse des chats
    - Analyse des followers

Il est aussi à noter que nous travaillons sur une base de données de noms de VTuber d'une taille assez petite (400 VTubers). Il serait intéressant de voir s' il y a une possibilité d'augmenter la taille. Cette étape n'a pas pu être réalisé lors de notre étude car notre base de données des noms de VTuber est basé sur le recensement de VTubers en ligne avec le tag VTuber et la langue fr, cependant cette recherche doit être effectué sur de longue période afin d'être la plus complète possible.

Autrement, d'un point de vue scientifique, nous pouvons qualifier notre démarche de répétable. En effet, en remodelant certains filtre, la démarche est assez générale et permet l'étude des VTubers, autant que l'étude de streamer de joueur de poker par exemple.

VIII. Conclusion et ouverture

Notre étude propose d'étudier le phénomène des VTubers sur Youtube et Twitch grâce à des analyses de données récupérées via les API des plateformes correspondantes. Cette étude marque son originalité car jusqu'à présent ce phénomène avait été envisagé d'un point de vue sociologique sans réels chiffres à l'appui. Grâce à notre analyse de données, nous pouvons avancer des faits et émettre des hypothèses plausibles appuyant nos propos. De plus, notre étude se concentre plus particulièrement sur les VTubers francophones, qui est un milieu moins connu que les VTubers japonais ou encore anglais.

Nous avons procédé en plusieurs étapes. Il a fallu dans un premier temps récupérer les données grâce aux API, les nettoyer et les traiter afin qu'elles soient exploitables par la suite. Nous avons ensuite défini une stratégie d'analyse permettant de mettre en évidence des phénomènes récurrents. Enfin, ces données ont été mises en forme dans un dashboard pour une meilleure visibilité d'un côté client par exemple, et permettant également une facilité de comparaison entre les deux plateformes Youtube et Twitch.

Cette étude peut être utilisée auprès de clients qui auraient des demandes. Un client pourrait par exemple souhaitait avoir un groupe de 5 Vtubers qui streament à eux cinq sur l'ensemble de la semaine 7j/ 7, 24h /24. Les graphiques produits montrant les tendances de publication pourraient répondre à cette demande.

Enfin, ce projet nous a beaucoup apporté. Tout d'abord, nous avons pu travailler sur un sujet qui nous intéressait beaucoup et qui était attractif de par les technologies impliquées. Nous avions au préalable des compétences en python mais cela nous a permis de les développer plus encore au service des données. Nous avons pu prendre en main les librairies python d'analyse de données (pandas) et de graphiques (seaborn, plotly) ainsi que pour l'élaboration d'une application web (dash avec plotly). Nous avons également eu un aperçu des APIs permettant de faire des requêtes. Outre ces compétences techniques, nous avons pu découvrir un milieu qui nous était étranger mais qui est très intéressant. Ce projet a également permis d'exercer nos capacités de réflexion et d'analyses. Il n'y avait, en effet, pas de ligne directrice donnée par nos encadrants qui ont souhaité nous laisser autonomes.

Enfin notre travail est accessible sur github au lien suivant : https://github.com/sahjiro/analysis-vtuber.

IX. Bibliographie

[1] Kim Morrissy, “Une société de classement de données répertorie plus de 16000 Youtubeurs Virtuels”, Animewsnetwork, https://www.animenewsnetwork.com/fr/interest/2021-10-20/une-societe-de-classement-de-donnees-repertorie-pus-de-16-000-youtubeurs-virtuels/.178647, publication : 20 octobre 2021, consultation : 22 octobre 2021.

[2] Jason Urgo, “Analytics Made Easy”, *Social Blade*, https://socialblade.com, publication : 2008, consultation : 25 janvier 2021

[3] Statista Research Department, “Most popular social networks worldwide as of January 2022, ranked by number of monthly active users”, https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/, publication : 8 mars 2022, consultation : 25 mars 2022.

[4] Pedro Hernandez, “YouTube Data API v3 in Python: Tutorial with examples“, *Medium*, https://medium.com/mcd-unison/youtube-data-api-v3-in-python-tutorial-with-examples-e829a25d2ebd#78e4, publication : 8 septembre 2021, consultation : 16 novembre 2021

[5] “API Reference”, Google Developers, https://developers.google.com/youtube/v3/docs, dernière date de publication : 2 juillet 2021

[6] “Youtubeurs virtuels francophones”, https://vtubers-fr.fandom.com/fr/wiki/Cat%C3%A9gorie:V-tubers_fran%C3%A7ais, date de consultation : 6 décembre 2021

[7] “API Reference”, Twitch Developers, https://dev.twitch.tv/docs, dernière date de publication : 15 mars 2022, consultation : 20 novembre 2021

[8] Twitch Tracker, “Twitch Subs Count Statistics”, https://twitchtracker.com/subscribers, dernière date de publication : mars 2022, consultation : 20 mars 2022