

Note d'analyse du GPE

Cette note permettra de réaliser une synthèse des différentes technologies qui seront utilisées pour le développement de ce projet, et de comprendre la démarche que nous avons pensée afin de la mettre en place.

Twitch, c'est une véritable success-story, lancée en 2011. Depuis, le service de streaming n'a cessé de gagner du terrain. Dans l'univers des jeux vidéo, la plateforme s'est fait un nom et draine des millions de visiteurs chaque mois – il y en aurait pas moins de 15 millions chaque jour à travers le monde – bien aidée par la popularité de certains streamers comme Ninja qui cumule plus de 14 millions de followers. Par ailleurs, sa chaîne a été observée pendant plus de 226 millions d'heures en 2018.

Grâce à Twitch, des millions de visiteurs partout dans le monde regardent leurs streamers jouer. Ces derniers vivent parfois des moments très intenses suite à la réalisation des exploits et avec un max de nombre de leurs communauté qui regardent. Ces moments sans doute sont à capturer et à enregistrer pour l'histoire.

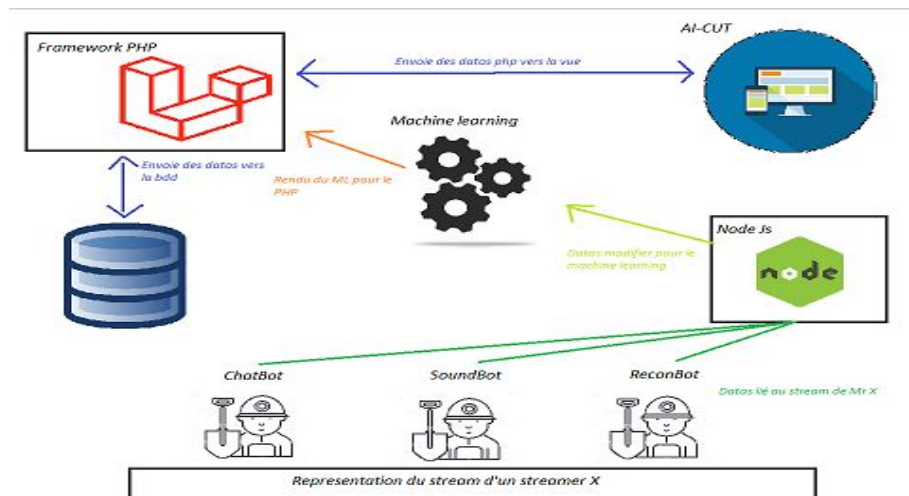
Nommé encore provisoirement AI-CUT, AI-CUT est un ensemble d'applications permettant de faciliter la vie d'un *streamer* en apportant un ensemble de solutions. Nous avons remarqué une augmentation du nombre de gens qui diffusent du contenu en live, pourtant des technologies manquent encore afin de les aider dans ce nouveau métier digital.

AI-CUT permettra à n'importe quel *streamer*, de pouvoir transformer "les moments forts" d'un live, à partir de critères prédéfinies, en clips créés automatiquement, afin que le *streamer* puisse en faire une vidéo par la suite, ou simplement les archiver et/ou les organiser. Le streamer bénéficie donc d'un gain de temps sur l'échantillonnage de ses clips et pourra prétendre à une meilleure visibilité sur la toile.

AI-CUT permettra la visualisation de données liées au *streamer* afin qu'il puisse toujours plus améliorer ses contenus en fonction de sa audience.

AI-CUT permettra une personnalisation de la diffusion du contenu. Cela permettra au *streamer* de diffuser très rapidement de nombreux contenus sur différentes plateformes de son choix.

Quelles sont donc les technologies que nous avons décidé de choisir afin de mener à bien ce projet ?



Node JS & Bots de récoltes

Lors de ce développement pour le projet du GPE nous allons avoir besoin d'un serveur node.js qui aura comme fonctionnalité de récupérer les données relative à un streamer grâce à des bots et d'alimenter le script de machine learning.

Pourquoi Node.js ?

Node.js est une plateforme logicielle libre en JavaScript intégrant un serveur HTTP. Son fonctionnement est basé sur une boucle événementielle lui permettant de supporter de fortes montées en charge.

L'utilisation de Node.js en tant que serveur web permet de traiter un gros volume de requêtes simultanément de manière efficace. Cette performance élevée s'explique par une conception asynchrone (modèle non bloquant) permettant d'éviter les attentes

Quel sont les bots qui vont travailler et donc alimenté le script de machine learning ?

- Un chatbot : Ce bot va permettre de pouvoir récolter des information lié au chat d'un streamer. Il devra gérer plusieurs tâches tel que : de l'administration du chat, récolter le nombre de viewers actif, le nombre de messages envoyés en X secondes, les messages qui se ressemblent et qui se suivent, le nombre d'abonnés et de non-abonnés.

- Un bot de reconnaissance faciale : Ce bot permettra de récupérer les expressions d'un streamer, par exemple si le streamer est assez expressif (amusement , peur , énervement, ...) alors le bot devra envoyer l'information à la machine learning pour peut être créer un moment clef
- Un bot analyseur de son : Ce bot quant à lui permettra de reconnaître le son du stream. Il pourra intégrer des phrases personnalisées du *streamer* comme "Bot lance le moment clef". Il pourra aussi définir un moment clef si le *streamer* crie ou explose de rire durant son stream.

Machine learning & script Python

Clairement, Python n'est pas vraiment au dessus des autres langages pour l'intelligence artificielle. Mais il s'y prête bien et sa syntaxe concise et facile permet d'y progresser très certainement plus aisément que dans d'autres langages.

Il dispose aussi de quelques bibliothèques spécialisées en IA qui lui permettent de s'initier à cette discipline.

En outre, concernant le machine learning, la problématique la plus difficile est souvent de disposer de bons jeux de données - et cela va parfois bien au delà du besoin de savoir implémenter ou utiliser correctement un algorithme d'apprentissage automatique. Sur ce plan, Python est particulièrement bien outillé avec des bibliothèques comme Numpy ou Pandas, par exemple.

Enfin, concernant le domaine de l'apprentissage automatique Python se distingue tout particulièrement en offrant une pléthore de bibliothèques de très grande qualité, couvrant tous les types d'apprentissages disponibles sur le marché ; le tout, accompagné d'une grande et dynamique communauté.

Par exemple, le projet Scikit-Learn qui est un des plus réputés pour cette discipline compte près de 800 contributeurs référencés sur GitHub.

Introduction à Scikit Learn

Scikits-Learn est une bibliothèque d'apprentissage automatique couvrant l'ensemble de la discipline:

- Les types d'apprentissage : supervisé, non supervisé, par renforcement, par transfert
- Les algorithmes :
 - Linear Regression (régression linéaire),
 - Logistic Regression (régression logistique),
 - Decision Tree (arbre de décision),
 - SVM (machines à vecteur de support),
 - Naive Bayes (classification naïve bayésienne),

Groupe de : Boulei_n, Broway_w, Amiens_k, Mamane_s, Slaman_n

- KNN (Plus proches voisins),
- Dimensionality Reduction Algorithms,
- Gradient Boost & Adaboost,
- Réseaux de neurones
- Elle dispose d'une excellente documentation fournissant de nombreux exemples
- Elle dispose d'une API uniforme entre tous les algorithmes, ce qui fait qu'il est facile de basculer de l'un à l'autre
- Elle est très bien intégrée avec les Librairies Pandas et Seaborn
- C'est un projet open source
- Son code est rapide, certaines parties sont implémentées en Cython

Le Framework PHP

Dans le cadre de notre projet, nous devons mettre en place une interface qui devra être capable de réaliser différentes choses:

- Récupération des datas traitées par le machine learning et le node.js
- Connexion et envoi des datas vers la BDD
- Traitement des données finales (Création du clip, Répartition et organisation des dossier local, Administration etc)

En basant sur les compétences techniques de l'équipe, nous avons hésité principalement entre deux outils, Symfony et Laravel.

Symfony et Laravel fournissent des frameworks et des fonctionnalités PHP multiplateformes qui accélèrent le développement d'une application PHP. L'échafaudage de l'application permet de définir sa structure et les plugins étendent la capacité à ajouter rapidement des fonctionnalités.

Au premier abord, les deux options se ressemblent. Cependant, nous avons déterminé quelques points qui nous ont permis de prendre une décision. En réalité, ce qui différencie les deux dépend des besoins spécifiques de l'application et du développeur.

Voici les points:

- Popularité. En effet, le nombre de ressources pour Laravel est plus important.
- Facilité d'apprentissage. Même si cela reste subjectif, nous avons trouvé plus de ressources pour le framework Laravel

Groupe de : Boulei_n, Broway_w, Amiens_k, Mamane_s, Slaman_n

- Livraison plus rapide avec moins de ressources. En effet, pour le GPE, nous devons être capable d'avoir une livrable très rapidement.

D'un point de vue technique, les deux frameworks pouvaient répondre à nos besoins.

Nous avons donc choisi le framework Laravel pour y parvenir.

Laravel est l'un des frameworks d'applications web open source très utilisé qui conçoit des applications web personnalisées rapidement et facilement.

Les développeurs préfèrent généralement Laravel à d'autres frameworks en raison des performances, des fonctionnalités et de l'évolutivité qu'il offre. Il suit Model View Controller (MVC), ce qui le rend plus utile que PHP.

Il tente d'alléger le développement en facilitant les tâches courantes utilisées dans la majorité des projets Web, telles que l'authentification, le routage, les sessions et la mise en cache. Il a une architecture unique, où il est possible pour les développeurs de créer leur propre infrastructure spécialement conçue pour leur application.

Laravel est utilisé non seulement pour les grands projets, mais aussi pour les petits projets.

Sources

Node.Js

<https://openclassrooms.com/fr/courses/1056721-des-applications-ultra-rapides-avec-node-js/1056866-node-js-mais-a-quoi-ca-sert>

<https://www.smooth-code.com/articles/6-raisons-de-choisir-node-js>

<https://blog.arcoptimizer.com/comment-construire-votre-premier-bot-discord-avec-node-js>

<https://blog.betomorrow.com/bot-slack-nodejs-d89c07e7313d>

Framework PHP

<https://laravel.com/>

<https://symfony.com/>

<https://medium.com/techcompose/why-laravel-is-consider-as-one-of-the-best-php-framework-in-2018-4f40def9c69c>

<https://www.valuecoders.com/blog/technology-and-apps/symfony-vs-laravel-php-framework-choose/>

<https://www.educba.com/laravel-vs-symfony/>

Machine learning

<https://www.lebigdata.fr/python-langage-definition>

<https://makina-corpus.com/blog/metier/2017/initiation-au-machine-learning-avec-python-pratique>

<https://scikit-learn.org/stable/>

<http://www.python-simple.com/python-pandas/panda-intro.php>