Année 2019-2020

RAPPORT DE PROJET – PTS 14

Prédiction du Succès d'un Film

Projet réalisé par

Riad BENRADI
Théo ZANGATO
Youssef ZEMALI

Projet encadré par

Philippe LIVOLSI

REMERCIEMENTS

Tout d'abord, nous tenons à remercier les personnes ci-dessous pour leur soutien et leur aide dans la réalisation de ce projet ingénieur :

- M. Philippe LIVOLSI, responsable de notre projet, pour ses conseils, ses orientations ainsi que la confiance qu'il nous a accordés tout au long de ce projet.
- L'ESILV et l'ensemble du corps enseignant pour leur coopération, leur accompagnement ainsi que leur professionnalisme durant toute la durée de ce projet.
- Nos camarades et tout particulièrement M. Steven Worick, qui nous ont apporté un soutien inconditionnel et qui nous ont apporté un regard extérieur afin que l'on mène à bien ce projet.

RESUME

Ce rapport traite de la conception de l'application développée dans le cadre du projet PTS de l'ESILV ayant pour but la prédiction du succès d'un film. Mettre la technologie et l'apprentissage supervisé semble aujourd'hui nécessaire à une industrie comme celle du cinéma, qui, gouvernée par l'incertitude, peine à trouver une rentabilité suffisante. Le système imaginé pour ce projet ingénieur classifie un film qui n'est pas encore sorti en salles en fonction de ses paramètres propres ou features afin de déterminer si ce dernier sera un flop, un film moyen (average) ou un hit.

Mots-clés : Machine Learning, prédiction, classification, apprentissage supervisé, Python, Deep Learning, cinéma

SUMMARY

This report exposes the conception of an application that was developed within the PTS project at ESILV. This application serves the need of predicting the outcome of a movie regarding the box-office. Nowadays, using Machine Learning and new technologies seems necessary for an industry such as the movie industry, which struggles to reach a break-even point due to the uncertainty of the success of a film. The software application designed during this engineering scale project classifies a new unknow film according to its features in order determine if this film is a flop, an average movie or a hit.

Key words: Machine Learning, prediction, classification, supervised learning, Python, Deep Learning, cinema

SOMMAIRE

I. INTRODUCTION

Dans le cadre de notre deuxième année de cycle ingénieur en informatique en alternance à l'ESILV il nous a été proposé la réalisation d'un projet de 7 mois afin de mettre en applications les enseignements et les connaissances assimilés au cours de notre cursus au travers d'un cahier des charges.

Nous avions un panel de 12 projets, et c'est parmi ces 12 projets que l'équipe composée alors de Théo ZANGATO et Youssef ZEMALI, qui sera rejoint dans un deuxième temps par Riad BENRADI, a choisi un projet en rapport avec la prédiction et plus précisément l'apprentissage automatique.

L'équipe 14, encadrée par M. Philippe LIVOLSI a donc choisi le 12ème sujet disponible : Prédiction du Succès d'un Film.

II. BESOINS ET OBJECTIFS DU PROJET

1. CONTEXTE

Le concept de Machine Learning a émergé dans la seconde moitié du XXème siècle. C'est un domaine de l'intelligence artificielle qui fait référence à l'implémentation d'algorithmes capables d'accumuler de la connaissance à partir d'expériences, sans être explicitement programmés pour chaque cas possible. Toutefois l'application d'une telle technologie dans le domaine du cinéma est récente, en effet les entreprises précurseurs dans le domaine comme Cinelytic ou ScriptBook opère toutes les deux depuis 2015.

Fort d'une industrie pesant près de 100$ milliard, l'industrie du cinéma, qui a bénéficié d'une hausse de 9% en 2018 semble d'apparence, avec ses 190 000 salles à travers le monde, bien se porter. Alors pourquoi ces nouveaux prestataires de services sont autant plébiscités ?

Pour comprendre cela implique d'appréhender le monde du cinéma et notamment le concept du *« Nobody knows »*. Ce concept exposé par Richard Caves, économiste et professeur à l'université d'Harvard, fait référence à l'incertitude qui pèse sur l'ensemble des produits culturels. Et c'est précisément à propos du cinéma que ce concept est né : On ne connaît pas la réussite d'un film avant qu'il ne soit sorti.
Le problème de rentabilité des films, malgré une industrie que semble en parfaite santé se pose de plus en plus à mesure que les budgets consacrés aux réalisations explosent. Le budget moyen d'un blockbuster à atteint aujourd'hui les 200$ millions, à titre d'exemple le budget d'*Avengers : Infinity War* s'élève à 420$ millions.
Afin de mieux comprendre pourquoi cette rentabilité occupe de plus en plus outre-Atlantique on notera qu'en 2017 le film américain de plus rentable a généré 360$ millions tandis que le 20ème n'a réalisé que 52$ millions, qu'en est-il alors des 337 autres films sortis cette même année ? En France le constat est pareil, si ce n'est pire. Malgré des budgets bien inférieurs à ceux des blockbusters américains, environ 4€ millions, en 2018, seulement 2% d'entre eux étaient rentables.
On comprend donc que la majorité des films ne sont pas rentables et que seule une poignée d'entre eux génèrent l'ensemble des profits du secteur, dominé par le *nobody knows*. Dès lors, il convient pour les acteurs financiers de cette industrie de trouver des solutions afin de prévoir le succès et donc la rentabilité.

2. MOTIVATIONS

Lors du choix du cahier des charges de notre projet nous avions une multitude de sujets différents à disposition, du développement d'applications en passant par la domotique. L'une de nos principales motivations était pour nous de pouvoir s'investir dans un projet qui allait nous permettre de parfaire nos compétences ainsi que d'acquérir de nouvelles connaissances en programmation et plus particulièrement en DataScience. Dès lors, ce projet ingénieur fut la parfaite occasion de s'essayer à la réalisation d'un projet concret concordant avec notre projet professionnel. Aussi, le fait d'être en complète autonomie sur un projet pour lui donner la forme que l'on souhaite en s'imposant aussi soi-même des contraintes et des critères de réussite est une source de motivation supplémentaire.

3. ENJEUX

Outre les motivations précédemment mentionnées, c'est aussi le défi technique qui nous a encouragé à choisir ce projet. L'apprentissage automatique et la classification étant des domaines relativement récents à l'échelle humaine et d'autant plus récents pour nous, étudiants, nous savions que ce projet allait nécessiter un travail de recherche et de documentation en plus d'une rigueur dans l'implémentation des différents algorithmes. Nous voulions tester nos compétences en situation et ce projet en fut le moyen adéquat.

D'autre part, bien que ce soit un projet scolaire, celui-ci fait référence à un problème bien réel qu'est la rentabilité des films et de l'industrie du cinéma plus généralement. La dimension économique qui entourait le projet nous a également motivés : Comment pourrions-nous assurer le succès d'un film au box-office français afin de prévoir sa rentabilité financière ?

Finalement, c'est aussi le travail de groupe sur un projet de longue durée qui fut un des enjeux de ce projet. Malgré un cahier des charges prédéfini, il est nécessaire de se mettre d'accord et de communiquer efficacement tout au long du projet afin de mener à bien ce dernier.

4. OBJECTIFS ET CONTRAINTES

a. Objectifs techniques

Le projet étant limité en temps et en ressources nous nous en sommes tenu à la stricte réalisation du cahier des charges, base du projet. Nous avons donc implémenté deux algorithmes pour ce faire. En parallèle nous avons rempli nos bases de données avec des données suffisantes collectées sur : les acteurs, actrices, réalisateurs et les films, afin que l'analyse puisse être pertinente.

Dans un second temps nous avons décidé d'implémenter deux nouveaux algorithmes pour mettre en perspective les résultats obtenus avec les deux premiers algorithmes et confronter les taux de réussite entre les différentes techniques employées.

Finalement nous avons créé une interface graphique afin de faciliter l'utilisation de notre application.

b. Contraintes

Différentes contraintes nous étaient imposées par le cahier des charges. Premièrement, celui-ci stipulait qu'afin de réaliser notre analyse, nous devions construire un jeu de données. Aucune information supplémentaire ne nous était donnée si ce n'est que c'était de notre ressort de définir les features à prendre en compte.
Il était aussi indiqué que l'implémentation d'un algorithme bayésien était obligatoire, et que le recours à des bibliothèques dédiées pouvait succéder à condition d'avoir implémenté en premier lieu l'algorithme nous-même.
Finalement le cahier des charges nous laissait le soin de définir ce qui caractérisait un film de flop, hit ou average.

Les délais furent aussi une contrainte à ne pas négliger. Outre la date de rendu final du projet, des échéances intermédiaires étaient fixées tous les mois avec une soutenance de mi-projet au mois de février. Afin de terminer ce projet, il était nécessaire de gérer correctement le temps et les différentes échéances, c'est dans cette optique que nous avons mis en place des outils comme le diagramme de GANTT par exemple.

III. GESTION DE PROJET

1. L'EQUIPE

L'équipe 14 est composée de trois étudiants aux parcours et cursus différents :

- Riad BENRADI, issu de classes préparatoires intégrée à l'ESILV et apprenti Développeur Full Stack à la BNP Paribas.
- Théo ZANGATO, issu de classes préparatoires intégrée à l'ESIEA et apprenti Data Scientist au Ministère des Armées.
- Youssef ZEMALI, chef du projet, issu des classes préparatoires aux grandes écoles du Lycée Albert Schweitzer et apprenti Ingénieur IT chez Lyxor Asset Management.

Fort de nos expériences tant personnelles que professionnelles différentes, l'équipe s'est formé autour de centres d'intérêt similaire ainsi qu'une volonté unique d'utiliser ce projet pour développer nos domaines de compétences respectifs. Nous nous sommes donc réunis autour d'un projet de prédiction et d'apprentissage automatique.

2. PLANNIFICATION ET OUTILS DE GESTION

Afin de mener à bien le projet et de veiller au respect des contraintes de délais et de performance l'équipe s'est appuyée, dès le début de celui-ci, sur des outils et techniques de gestion de projet. Ces techniques et outils sont d'autant plus indispensables que les différents membres et futurs ingénieurs seront amenés à prendre part, et gérer des projets d'envergure à l'avenir. L'équipe a donc profité de ce « projet ingénieur » pour expérimenter et mettre se familiariser avec ces techniques et outils.

a. Cahier des charges

Le cahier des charges, qui décrit l'ensemble des besoins et les explique aux différents acteurs. Le cahier des charges était fourni dès le début de ce projet et nous a permis de cadrer les objectifs à réaliser, les livrables à délivrer ainsi que les axes de développement envisagés. Il nous a servis tout au long du projet pour vérifier la concordance de notre projet.

b. Moyens de communication

Les tâches qui incombent chaque acteur du projet étant réalisées séparément, il a fallu mettre en place des voies de communication afin de permettre aux membres d'échanger, de communiquer sur l'avancement du projet et de leurs tâches respectives, mais aussi de travailler en collaboration. Nous avons donc créé un groupe Teams pour communiquer et un drive pour échanger. Nous avons aussi fait le choix de développer sur Collabotary, un IDE en ligne de Google, qui permet de stocker les programmes sur le drive et de coder en simultané.

c. Diagramme de GANTT

Le diagramme de Gantt, nous a permis de représenter visuellement l'état d'avancement des différentes tâches qui constituent le projet. Chaque tâche, matérialisée par une barre horizontale, dont la position et la longueur représentent la date de début, la durée et la date de fin offre une vision de l'avancement du projet ainsi que les taches qui dépendent d'autres taches antérieures afin de prévoir au mieux d'éventuels problèmes.

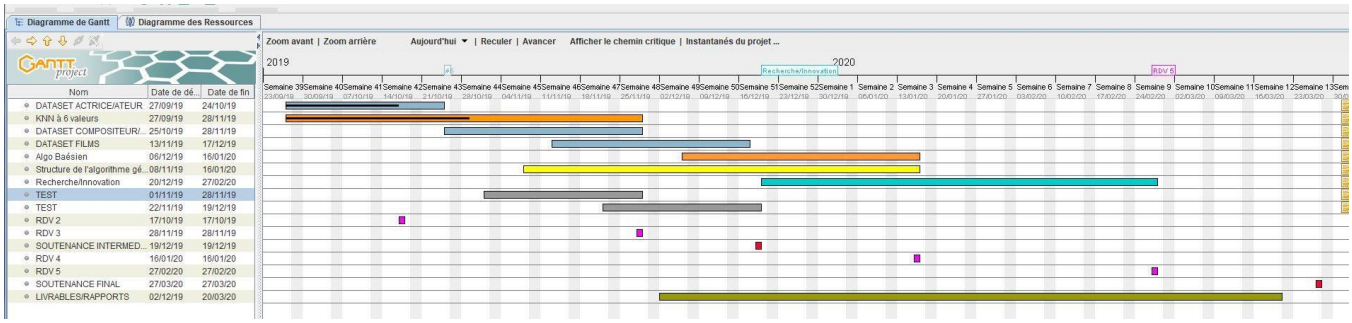

Figure 1 : GANTT

3. REPARTITION DES TACHES ET DES TECHNOLOGIES

a. Répartition des tâches

Le diagramme ci-dessous illustre la répartition en unité de temps des différentes tâches majeures du projet. On apprécie facilement les taches les plus chronophages.

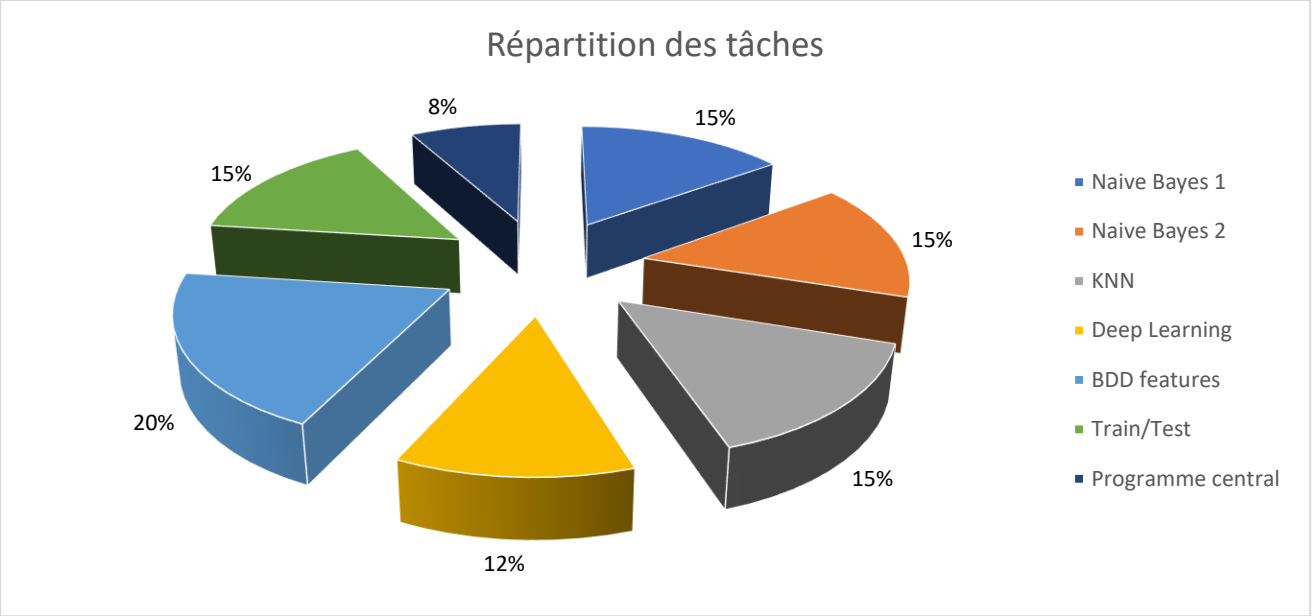

Figure 2 : Répartition des Tâches

b. Répartition des technologies

Le diagramme ci-dessous illustre la répartition des différentes technologies utilisées, la place qu'elles occupent au sein du projet.

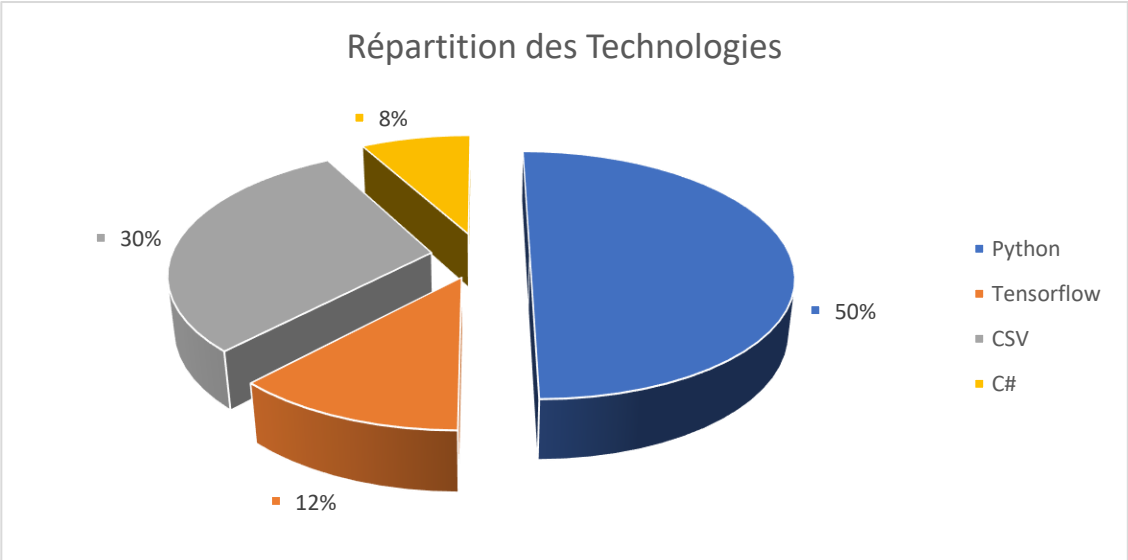

Figure 3 : Répartition des Technologies

IV. DEVELOPPEMENT TECHNIQUE

1. STRATEGIE

a. Stratégie prévisionnelle

Une fois le cahier des charges en notre possession il a fallu mettre en place une stratégie prévisionnelle et fixer les objectifs intermédiaires pour échelonner les tâches afin d'arriver à réaliser le projet dans les temps. La stratégie initialement mise en place par l'équipe, a évoluée car elle s'est basée sur des estimations qui se sont heurtées aux différents problèmes, contraintes et évènements extérieurs qui ont pu survenir.

Comme dans tout projet, l'interface et l'expérience utilisateur sont des éléments non négligeables d'un projet. Néanmoins, le devoir de résultats prime sur l'ergonomie et le design. Dès lors, l'équipe s'est avant tout focalisée sur la partie algorithmique du projet avant toute chose.
La base de l'apprentissage automatique est de s'appuyer sur une base de données pour entrainer le modèle à reconnaitre les différentes classes. Nous avons donc dans un premier temps élaborée nos bases de données. En parallèle, nous avons commencé à développer le premier algorithme bayésien sur un jeu de donnée de test.

Nous avions développé l'algorithme KNN en C#, nous avons décidé de l'adapter afin de pouvoir l'utiliser pour confirmer ou non les résultats obtenus avec le premier algorithme. On a ici le fil conducteur du projet, respecter le cahier des charges tout en s'appuyant sur des connaissances déjà acquises lors de notre cursus.

Lors de nos recherches sur les algorithmes naïf bayésiens nous avons trouvé différents moyens de converger vers le même objectif de classification. Le cahier des charges stipule que l'on peut s'appuyer sur des bibliothèques existantes un fois que l'on a développé un algorithme. Toutefois l'équipe n'a pas retenu cette solution et a donc utilisée les résultats de ses recherches pour développer un deuxième algorithme bayésien.

Durant la période de conception du projet, les membres du projet ont aussi acquit de nouvelles connaissances notamment en Deep Learning. Nous avons implémenté un nouvel algorithme se basant sur un réseau de neurones pour construire une nouvelle analyse avant de tout regrouper au sein d'un même programme.

Dans sa version finale, le projet comporte donc 4 algorithmes distincts, qui réalisent la même classification du succès d'un film.

b. Architecture Logicielle

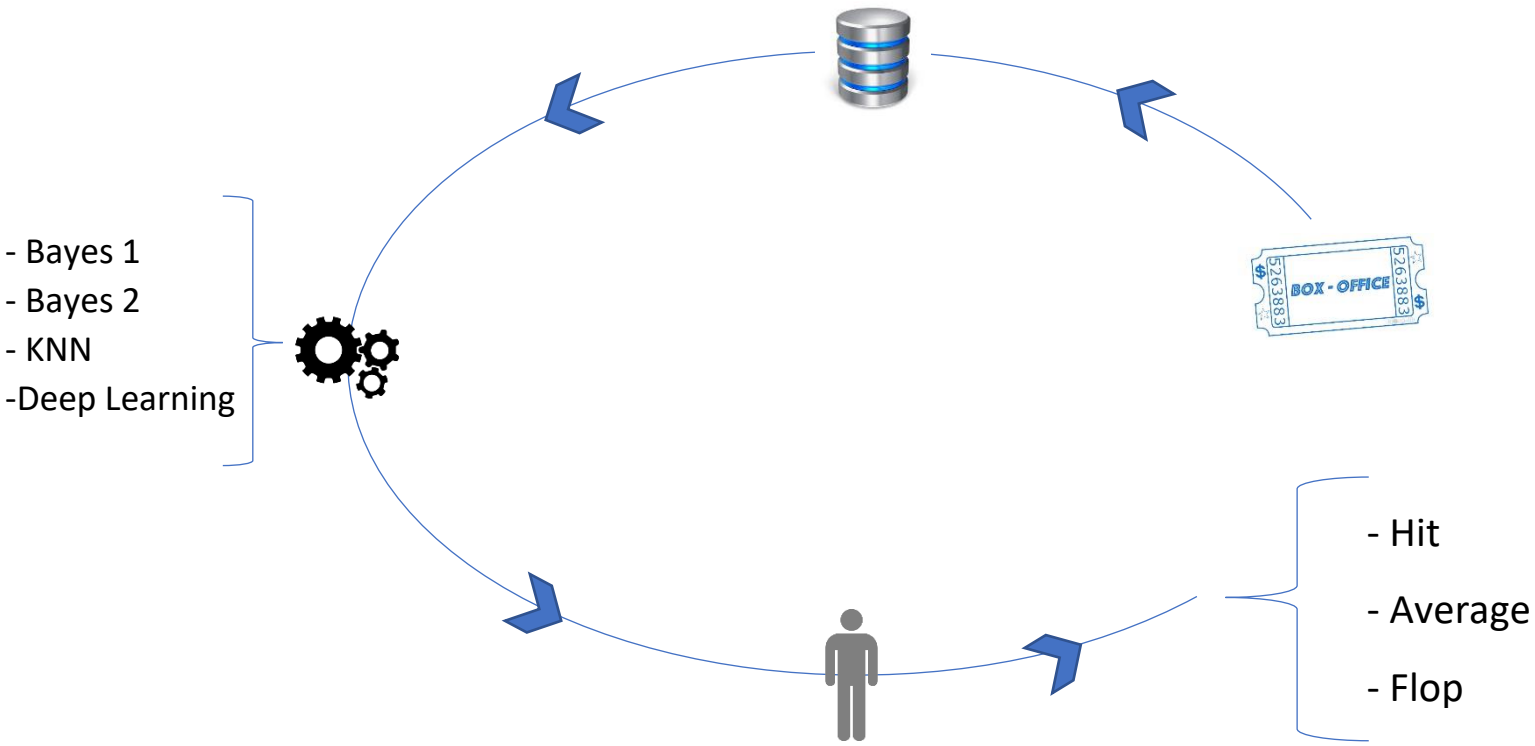

Figure 4 : Architecture logicielle

Le schéma ci-dessus représente l'architecture logicielle de notre projet. Peu importe la méthode de classification choisie la structure reste pareil. Les chiffres du box-office viennent nourrir la base de données, ensuite l'algorithme se sert de cette dernière pour apprendre 3 les classes, l'utilisateur saisit alors une nouvelle entité, un film, représenté par ses features le caractérisant, pour finir l'algorithme le compare à ce qu'il connait et le classifie.

2. SOLUTIONS LOGICIELLES

a. CSV

Afin de stocker les informations extraites sur internet qui composent les différentes bases de données nécessaires à l'apprentissage nous avons fait le choix d'utiliser des fichiers *.CSV*. Les fichiers *CSV* sont des fichiers texte formatés représentant des données tabulaires sous forme de valeurs séparées par un charactère défini. Ce choix a été fait pour faciliter l'accès aux différentes bases de données par nos algorithmes.

b. C#

C# est un langage de programmation orienté objet et dérivé du *C++* lui-même dérivé du C. Il n'a pas été très présent dans le projet car nous avons fait le choix, par soucis de simplicité de tout migrer en *Python*. Néanmoins l'algorithme KNN a été développé en *C#*.

c. Python

Python est un langage de programmation interprété dynamique. C'est le langage principalement utilisé dans le milieu de la DataScience, c'est la raison pour laquelle, en plus de sa souplesse relative, on a choisi de développer l'intégralité du projet en *Python* afin de monter en compétence.

d. TensorFlow

TensorFlow est un outil open source d'apprentissage automatique développé par Google. Il se base sur le langage Python cité ci-dessus et utilisé pour le projet. C'est l'un des outils les plus répandu en intelligence artificielle et en apprentissage car il permet grâce à des bibliothèques comme *Keras* de mettre en place des réseaux de neurones nécessaires pour l'implémentation du Deep Learning.

3. IMPLEMENTATION

L'ensemble des algorithmes qui va être exposés dans cette partie du rapport fonctionnent tous autour des mêmes paramètres que nous avons choisis. En effet, nous avons dû déterminer quels sont les paramètres qui influencent la réussite d'un film. Après diverses recherches nous avons choisi de nous baser sur le casting du film, comme l'entreprise Cinelytic *(cf. p14 : II,1)*. Nous avons donc élaboré notre jeu d'essai autour du casting du film. Dans un premier temps nous voulions choisir comme paramètres : [acteur,actrice,réalisateur,compositeur,producteur,genre], toutefois la pauvreté des ressources pour certains de ces indicateurs nous a amené à revoir notre modèle. Finalement nous avons choisi comme liste de features d'utiliser les deux principaux acteurs, les deux principales actrices ainsi que le réalisateur.

a. Base de Données

Notre base de données se compose de 5 fichiers. Trois d'entre eux contiennent les informations nécessaires aux différents algorithmes concernant les acteurs, les actrices et les réalisateurs. Ils contiennent tous les trois 120 enregistrements qui contiennent eux-même le nombre d'entrées de leurs cinq derniers films. A cela s'ajoute une note ainsi que le nombre de films réalisés en carrière. Tous les algorithmes n'utilisent pas les deux derniers paramètres mais tous se basent sur le nombre d'entrées des cinq derniers films.
La base de données comporte aussi un test set de 16 films dont il faut prédire le succès et un training set de 120 films.
La principale difficulté que l'on a rencontrée durant ce projet a été de construire la base de données. En effet nous n'avions, alors, pas les connaissances nécessaires afin de réaliser un scrapping pour remplir la base de données. De plus les bases de données disponibles en ligne ne satisfaisaient pas nos exigences, nous avons donc les construire à la main, ce qui fut très chronophage.

b. Naïve Bayes Classifier

Le classifieur naïf bayésien est un classifieur linéaire permettant de classer selon un modèle probabiliste basé sur le théorème de Bayes. L'appellation « naïf » fait référence au postulat qui consiste à supposer une totale indépendance des features entre eux. Le modèle se base sur de l'apprentissage supervisé pour reconnaitre par la suite les différentes classes.

Ainsi grâce au théorème de Bayes nous pouvons écrire :

$$p(C|F_1,\dots,F_n) = \frac{p(C)\,p(F_1,\dots,F_n|C)}{p(F_1,\dots,F_n)}.$$

I. Premier modèle

Le premier modèle développé est fidèle au théorème de Bayes ci-dessus. Les étapes d'implémentations sont définies comme telles :

- On calcule les probabilités d'appartenance à chaque classe cible de chaque feature pour chaque enregistrement donné du dataset. Cela correspond à multiplier la probabilité conditionnelle pour chaque enregistrement.
- On regroupe ensuite chaque probabilité conditionnelle avec la probabilité antérieure correspondant pour chaque classe.
- On normalise ensuite le résultat pour se rapprocher au mieux de la réalité.

La normalisation a été calculée comme étant le rapport de la somme du nombre de films réalisés par chacun des protagonistes du film, sur le nombre total de films réalisés par toutes les personnes figurant dans le dataset. En effet certains acteurs figurent dans des films qui font beaucoup d'entrées mais ne joue que dans très peu de films, il est donc nécessaire de pondérer cette information. Ce choix de pondération a été influencé par le fait que nous ne possédions pas un datasets exhaustif des acteurs, films et réalisateurs, rendant compliqué une pondération se rapprochant le plus de la réalité. Cette étape de pondération fut compliquée à mettre en œuvre contenu notre dataset, nous avons testé plusieurs approches comme une pondération par somme des ratios de film par année pour chaque feature par exemple.

Le schéma ci-dessous, illustre les étapes d'implémentation.

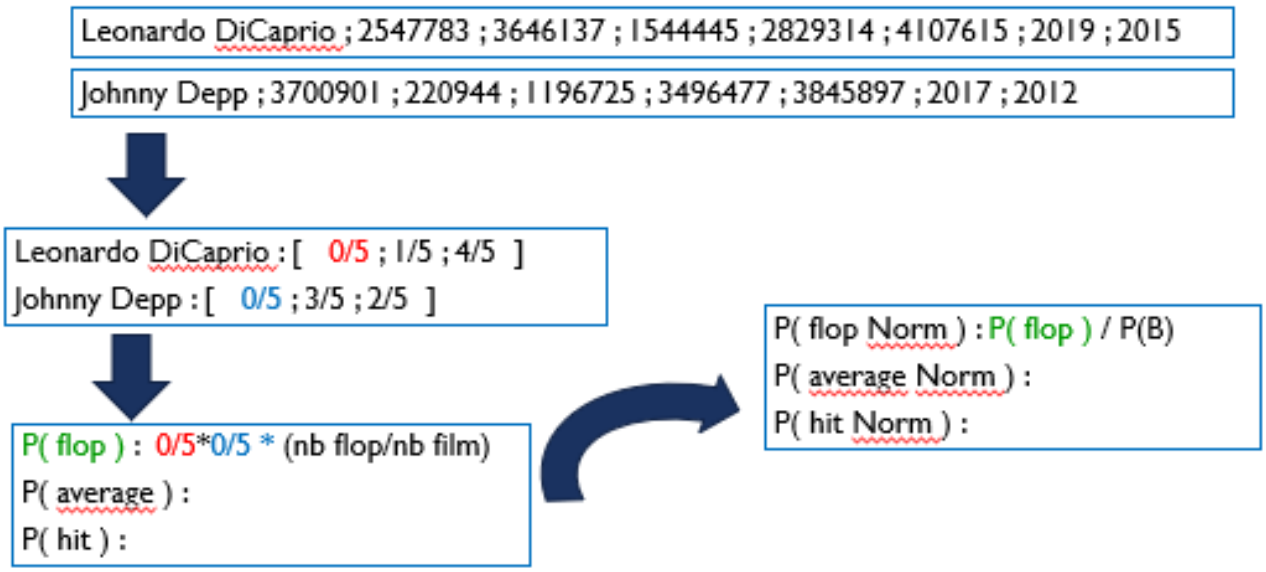

Figure 6 : Exemple d'implémentation du classifier de Bayes

II. Deuxième modèle

Concernant le deuxième modèle, une autre approche est utilisée. En effet, pour réaliser un classifier naïf bayésien il est courant d'utiliser le maximum de vraisemblance qui est un estimateur statistique utilisé pour inférer les paramètres de la loi de probabilité d'un échantillon donné en recherchant les valeurs des paramètres maximisant la fonction de vraisemblance. Dès lors, il est possible de s'affranchir des probabilités bayésiennes. Nous n'utilisons plus que l'écart-type et la moyenne des différentes features.

Pour y parvenir, il nous a d'abord fallu séparer notre dataset par classe c'est-à-dire Hit, Average et Flop. Nous avons ensuite pour chaque classe, calculer la moyenne et l'écart-type de chaque paramètre d'entrée. Ainsi, pour la classe Hit, nous obtenons 5 moyennes et écart-types différents qui correspondent aux 5 paramètres d'entrée (note acteur1, note acteur2, note actrice1, note actrice2, note réalisateur). Il s'agit du même principe pour les 2 autres classes. Nous pouvons alors calculer la probabilité pour chaque film d'appartenir aux différentes classes en utilisant la densité de probabilité de la loi normale qui est donnée par cette fonction :

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Il nous suffit alors de récupérer la probabilité la plus élevée afin de savoir à quelle classe appartient le film.

c. KNN

L'algorithme des K plus proches voisins, ou *KNN* (K-Nearest-Neighbours) fait référence en apprentissage supervisé à une méthode de classification qui permet d'estimer la classe associée à une nouvelle entrée en la comparant à ses K plus proches voisins selon une distance préalablement fixée. La classe retenue est alors la plus représentée parmi les voisins de la nouvelle entrée. Cet algorithme diffère de ceux précédemment énoncés car il ne nécessite pas de base d'apprentissage à proprement parler. En effet, il utilise et travail directement sur le test set sans exiger de s'entrainer sur un jeu d'essai.

Le schéma ci-dessous présente une représentation graphique de l'algorithme des K plus proches voisins.

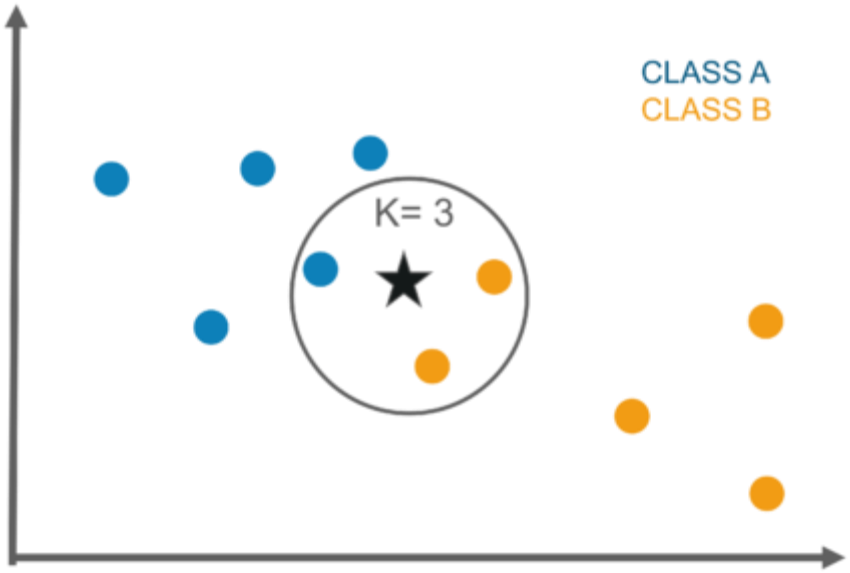

Figure 7 : Représentation graphique KNN

d. Deep Learning

Le Deep Learning en intelligence artificielle est un type de Machine Learning ou la machine apprend d'elle-même en corrigeant ses erreurs. Le Deep Learning s'inspire du cerveau humain en s'appuyant sur des réseaux de neurones. Le réseau correspond aux différentes couches de neurones qui reçoivent chacune une information en provenance des couches antérieures. Chaque neurone interprète mathématiquement une information avant de la pousser à la couche postérieure.

Notre réseau s'établit comme tel :

- Une couche intérieure de 50 neurones avec une fonction d'activation du type RELU définie par : $f(x) = x^+ = \max(0, x)$.
- Une couche de sortie composée de 3 neurones, un neurone pour chaque classe de sortie avec une fonction d'activation du type softmax définie par :

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \text{ pour tout } j \in \{1, \dots, K\}.$$

Le schéma ci-dessous est une représentation de notre réseau de neurones.

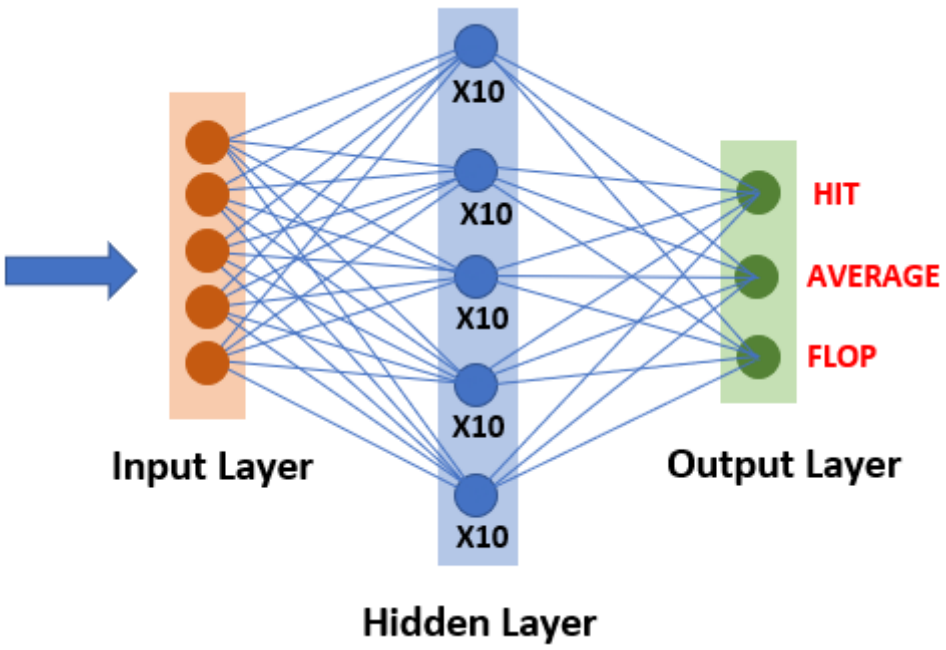

Figure 8 : Représentation graphique du Réseau de Neurones

e. Résultats

Voici le tableau récapitulatif des résultats de nos différents algorithmes. On remarque aisément que l'ensemble des algorithmes reconnaissent facilement nos différentes classes à l'exception de notre réseau de neurones. Concernant les algorithmes qui fonctionnent bien, cela est forcément dû au fait que le nombre de feature choisi n'est pas très élevé, une analyse avec plus de features pourrait nous permettre de préciser ces résultats.
Concernant l'algorithme de Deep Learning, on ne possède pas à ce jour de méthode permettant de choisir la structure parfaite qui maximise la reconnaissance, pour cela, il est nécessaire de tester de nombreuses structures différentes afin de choisir celle qui obtient de meilleurs résultats.
A l'avenir, tester toutes ces possibilités nous permettrait d'augmenter le taux de réussite de cet algorithme.

Algo\Classes	HIT	AVERAGE	FLOP
KNN	5/5	4/5	5/5
Naïve Bayes 1	4/5	5/5	5/5
Naïve Bayes 2	4/5	5/5	5/5
Deep Learning	5/5	1/5	2/5

Figure 9 : Tableau Récapitulatif des Résultats

V. BILAN DU PROJET

1. APPORTS INDIVIDUELS ET COLLECTIFS

Riad Benradi

Malgré le fait que je sois arrivé en retard sur le projet, mes coéquipiers m'ont directement et sans aucun mal intégré au projet. Le sujet qui nous était proposé utilisait des notions que nous avions déjà étudiées au semestre dernier, c'était donc plus facile à aborder d'où le choix de cette thématique.
Remplir le dataset, malgré le fait que ce fut une tache fastidieuse, était finalement l'occasion de se renseigner un peu plus sur le monde cinématographique et ses classiques. Pour conclure, le projet fut mené à bien surtout par la base instaurée par mes coéquipiers lorsqu'ils n'étaient que deux, ce qui a permis le bon déroulement du projet tout le long de cette année.

Théo Zangato

Ayant commencé à découvrir les nombreux outils et technologies relatifs à la DataScience l'année dernière et cette année, aussi bien au travers d'enseignements qu'en entreprise, je souhaitais m'investir dans un projet concordant avec mon projet professionnel. Ce projet était la parfaite occasion pour moi de découvrir ces outils mais aussi de développer un peu plus mes compétences en programmations. J'ai pu apprécier la nécessité de mettre en place une gestion de projet dès la genèse du projet et de la maintenir tout au long de ce dernier pour pouvoir le mener à bien.

Youssef Zemali

Etant un passionné de DataScience, le choix de ce projet m'est apparu comme une évidence. En effet, j'ai pu au cours de ce projet utilisé les compétences que j'ai acquises durant cette année en Machine Learning, Neural Network et en Python. Ce fut l'occasion d'utiliser les différentes techniques apprises en cours mais aussi de réaliser un projet concret avec diverses contraintes et d'y faire face avec une équipe soudée. Effectivement, nous avons pu mettre en place une réelle organisation d'équipe qui a porté ses fruits tout au long du projet et qui nous à amener à la réussite du projet.

2. Conclusion générale

Cette année, nous avons pris part à un projet scolaire, en plus des différents projets que auxquels nous contribuons dans nos entreprises respectives. Ce projet est un vrai projet ingénieur, avec l'établissement qui a joué le rôle de client en nous fournissant un cahier des charges, des délais à respecter, des réunions de projets mensuelles et des livrables à fournir.

Ce projet informatique nous a permis de travailler sur de nouveaux outils de DataScience et d'apprentissage automatique via une réelle étude de cas sur le cinéma. Nous avions à cœur de faire le maximum et de ne pas s'en tenir uniquement au cahier des charges initialement prévu. En effet, les différentes technologies utilisées nous ont permis de vérifier, ou non, nos résultats, et ont donc participé à un éveil collectif sur la nécessité d'interprétation et de validation des résultats obtenus.

La gestion d'un projet avec des membres ayant des parcours et des cursus différents, qui ne se connaissaient pas forcément au début du projet, nous a permis de nous projeter un peu plus dans nos futurs projets ingénieurs.

Finalement, nous sommes tous satisfait à titre personnel et collectif du projet que nous avons réalisé aujourd'hui.

VI. PERSPECTIVES

Ce projet, malgré qu'il soit un projet scolaire, peut être considéré comme un POC (Proof Of Concept), une ébauche d'un réel projet plus ambitieux à destination d'une entreprise et de l'industrie du cinéma comme c'est déjà le cas aujourd'hui.

Nous avons démontré qu'il est possible avec peu de moyens et de connaissances d'aboutir à des résultats exploitables, il conviendra dès lors d'apporter des modifications afin de garantir et renforcer ces résultats.

On pensera par exemple à :

- S'appuyer sur une vraie base de données en scrapant le web afin d'augmenter considérablement les datasets et les features prit en compte
- Réaliser une étude des features les plus importants (*PCA, Decision Tree Classifier*)
- Renforcer l'expérience utilisateur et ouvrir une application web pour permettre l'accès généralisé à la plateforme.

VI. BIBLIOGRAPHIE

https://fr.wikipedia.org/wiki/Th%C3%A9or%C3%A8me_de_Bayes

https://fr.wikipedia.org/wiki/Probabilit%C3%A9_conditionnelle

https://fr.wikipedia.org/wiki/Classification_na%C3%AFve_bay%C3%A9sienne

https://stackoverflow.com/

https://www.programiz.com/

https://docs.python.org/

https://www.stechies.com/

https://www.bfmtv.com/tech/a-hollywood-l-intelligence-artificielle-utilisee-pour-predire-le-succes-des-films-1701850.html

https://en.wikipedia.org/wiki/Richard_E._Caves

https://www.linkedin.com/company/scriptbook/

cinelytic.com/about/

https://www.quantmetry.com/une-petite-histoire-du-machine-learning/

https://mrmint.fr/introduction-k-nearest-neighbors

https://www.futura-sciences.com/tech/definitions/intelligence-artificielle-deep-learning-17262/

https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/ReduceLROnPlateau