Université Paris-Dauphine | Tunis
Institut Pasteur de Tunis

Rapport de stage d'été (Juillet – Aout 2019)

Développement d'approches « Deep-Learning » pour la découverte des médicaments.

Réalisé par : **Merhbene Oumaima (M1 BIG DATA)**

Sous la direction de **: Emna Harigua (PhD. Bioinformatician)**

Année universitaire 2018/2019

Remerciements

La réalisation et le bon déroulement de ce stage a été possible grâce au concours de plusieurs personnes à qui je voudrais témoigner toute ma reconnaissance.

- ✓ Mes sincères remerciements et mon profond respect à **Mme Emna Harigua** qui a eu l'amabilité de m'accueillir comme stagiaire au sein du laboratoire bioinformatique. Le suivi qu'elle a apporté à mon stage, ses encouragements, sa confiance et surtout l'autonomie qu'elle m'a offert pendant ce stage.
- ✓ Je tiens à exprimer mes plus vifs remerciements à **Mme Khaouther BOUSSEMA** pour son soutien, sa compétence, sa rigueur scientifique, sa clairvoyance, ses encouragements et surtout ses conseils qui m'ont beaucoup appris.
- ✓ Je tiens à saisir cette occasion et adresser mes profonds remerciements et mes profondes reconnaissances à **Mr Hichem RAMMEH** pour son encadrement et ses conseils précieux durant toute l'année universitaire.
- ✓ Je remercie également très vivement **Mme Claudine DHUIN** pour son encadrement et son soutien constant durant toute l'année universitaire.
- ✓ J'exprime toute ma reconnaissance et gratitude à **l'Université de Paris Dauphine Tunis** pour l'opportunité qu'elle m'a accordée et surtout sur la qualité de son enseignement.
- ✓ Enfin, je tiens à remercier ma famille pour le soutien et la patience qu'elle m'a témoignée et toutes les personnes qui m'ont conseillé et relu lors de la rédaction de ce rapport de stage : mes amis et mes camarades de promotion.

Merci à toutes ces personnes qui ont fait de mon stage une réussite.

Sommaire

I. Introduction :

Pour créer un nouveau médicament, les chercheurs doivent tester des dizaines de milliers de composés et déterminer comment ils interagissent. Une fois une substance jugée efficace contre une maladie, elle doit ensuite continuer de fonctionner au fil des trois phases d'essais cliniques et finalement être approuvée par les organismes de réglementation. On estime qu'en moyenne, la mise sur le marché d'un nouveau médicament nécessite 1000 personnes, 12 à 15 ans, et jusqu'à 1,6 milliard de dollars.

Une nouvelle étude parue dans la revue bioinformatique dévoile une intelligence artificielle (IA) qui prédit les risques d'interactions médicamenteuses grâce au Deep Learning. L'IA conçu pour aider à découvrir de nouveaux médicaments et raccourcir considérablement le temps et l'argent nécessaires pour le faire.

A l'Institut Pasteur, au laboratoire de recherche bioinformatique mon stage consiste à développer des algorithmes en se basant sur des « open source » pour prédire des nouvelles associations médicament -maladie.

Je suis passionnée de réaliser mon stage dans ce domaine et appliquer le Deep Learning en bioinformatique pour obtenir des informations aussi précieuses qui servira les chercheurs en biologie.

Cette mission m'a attiré car c'est un bon point de départ pour mettre en pratique non seulement mes connaissances en IA mais aussi en statistique et probabilité.

II. Motivation et contexte :

A-Motivation :

Mon stage est un premier pas d'une recherche proposer par Mme Emna Harigua qui consiste à améliorer l'approche de DeepDR (Deep Learning-based drug repositioning) en changeant l'Input.

L'approche DeepDR :

DeepDR (Source ouverte publiée le 22 mai 2019) est une étude qui a développé une approche de Deep Learning en intégrant 10 Networks hétérogènes et non linéaires pour la réutilisation de médicaments (*drug repurposing*) . En fait la réutilisation des médicaments est le ré-emploi des médicaments déjà approuvés dans le traitement d'une première pathologie ou des médicaments en cours de développement, mais dont l'innocuité a déjà été démontrée lors des essais cliniques de phase I, afin de traiter une maladie totalement différente de celle pour laquelle ils avaient été développés à l'origine.

Plus précisément, DeepDR apprend les caractéristiques de haut niveau des médicaments à partir de 10 réseaux hétérogènes. Nous avons constaté que DeepDR révélait des performances élevées. Il est important de noter que les associations médicament-maladie prédites à la DeepDR ont été validées par la base de données ClinicalTrials.gov (AUROC = 0,826) par exemple la validation de plusieurs nouveaux médicaments pour la maladie d'Alzheimer.

B-Contexte :

Durant mon stage, dans un premier lieu j'ai bien étudié le projet DeepDR puis j'ai reproduit ces pipelines. J'ai ensuite généré un nouveau set de données fournit par le labo et la transformer en une 11^{ième} network en utilisant la librairie python RDkit. Finalement j'ai refait le calcul avec DeepDR.

III. DeepDR pas à pas :

A. Introduction :

Le concept de DeepDR est de fusionner diverses informations provenant de différents types de réseaux et déduire de nouvelles applications pour les médicaments existants qui n'ont pas été approuvés à l'origine.

Tout cela en construisant un modèle de prévision dans lequel le résultat est défini comme si le médicament est un traitement connu de la maladie, et les prédicateurs sont les profils d'expression de chaque médicament. Les médicaments qui ne sont pas connus à l'origine pour traiter la maladie mais ont des probabilités prédites élevées sont considérées comme de bons candidats pour le repositionnement.

Les données de DeepDR sont rassemblées à partir de deux bases de données couramment utilisées DrugBank (Wishart, et al., 2018) et repoDB (Brown and Patel, 2017). En totale, ils ont construit 9 Networks : (1) clinically reported drug-drug interactions, (2) drug-target interactions, (3) drug-side-effect associations, (4) chemical similarities, (5) therapeutic similarities dérivé de « the Anatomical Therapeutic Chemical Classification System », (6) drugs' target sequence similarities, (7) Gene Ontology (GO) biological process, (8) GO cellular component, (9) GO molecular function. Pour la validation, une 10ème Network est construite en assemblant les plus récent associations médicament-maladie de la base de données ClinicalTrials.gov .

B. Reconstruction des réseaux hétérogènes :

C'est la partie de prétraitement de ces 10 Nets. En effet on applique à chacune quelques fonctions mathématiques pour capturer les informations structurelles du réseau et pour caractériser le contexte topologique de chaque Net afin de les utiliser comme une entré dans le prochain modèle d'IA.

- **La fonction Compute_similarities :**

  Elle consiste à calculer la matrice de similarité du Net toute en calculant la distance de Jaccard entre les lignes.

- **La fonction ScaleSimMat :**

  Elle a comme objective de normaliser la matrice.

- **La fonction RandomSurf :**

Nous adoptons une approche (Cao,2016 Random walk-based network). En supposant que le sommet actuel soit le -ième sommet, une matrice de transition A capture les probabilités de transition i entre différents sommets. Il prend en compte les modèles de connectivité topologique locaux et globaux au sein du réseau pour exploiter pleinement les relations directes ou indirectes entre les nœuds. Ainsi, à chaque fois, la procédure de marche aléatoire continuera avec une probabilité de ω et retournera au sommet d'origine et recommencera la procédure avec une probabilité de $1 - \omega$. Cette étape peut être reconnue matrice diagonale comme suit :

$$p_k = \omega . p_{k-1} . A + (1 - \omega) . p_0$$

Où p_k est un vecteur ligne, dont j-ème entrée indique la probabilité d'atteindre le j-ième sommet après k étapes de transitions, et p_0 est le vecteur 1-hot initial avec la valeur de la j-ième entrée étant 1 et toutes les autres entrées i étant 0. En sommant chaque marche aléatoire de p_k et en répétant le processus pour chaque nœud du réseau, nous pouvons obtenir une matrice probabiliste de cooccurrence.

On obtient alors les matrices PCO.

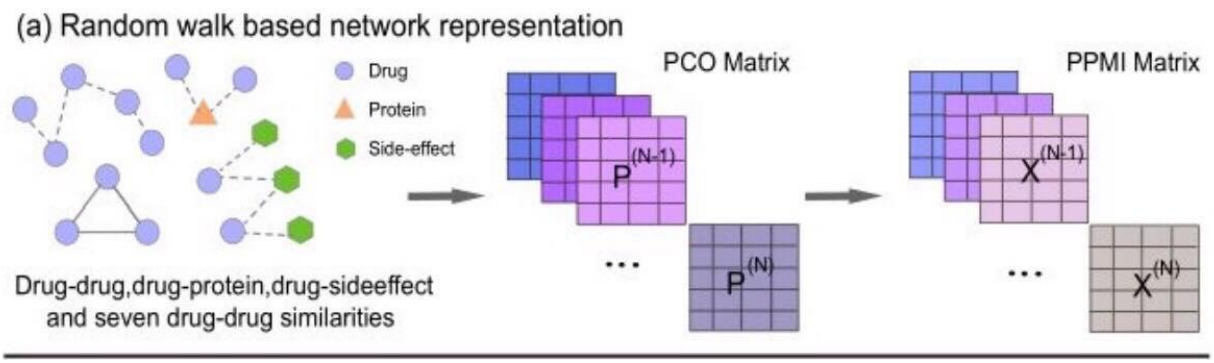
- **La fonction GetPPMIMatrix :**

Ensuite, on calcule les matrices PPMI (Positive Pointwise Mutual Information). La matrice PPMI peut être considérée comme une méthode de factorisation matricielle qui factorise une matrice de cooccurrence pour obtenir des représentations de réseau. La matrice PPMI peut être construite comme suit:

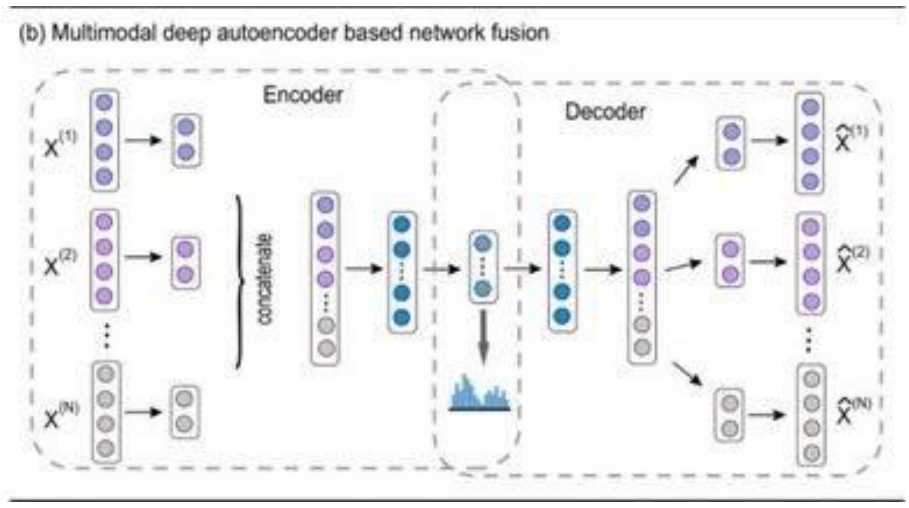
$$PPMI = \max\left(log \frac{M(i,j) . \sum_i^{Nr} \sum_j^{Nc} M(i,j)}{\sum_i^{Nr} M(i,j) . \sum_j^{Nc} M(i,j)}\right.$$

Où M est la matrice de cooccurrence d'origine, Nr le nombre de lignes, Nc le nombre de colonnes. Les valeurs PPMI négatives sont remplacées par 0.

Ces étapes basées sur la marche aléatoire atténuent la dispersion « sparsity »de certains réseaux, ce qui constitue une étape de prétraitement préalable à l'intégration plus poussée décrite dans les étapes suivantes.



C. Fusion des Networks via multimodal deep autoencoder (MDA) :



Cette partie consiste à fusionner les matrices PPMI en une matrice compacte de faible dimension qui représente tout les Networks à travers un multimodal deep autoencoder (MDA). La nouvelle matrice est en fait extraite de la couche interne de l'MDA.

Un autoencoder est un type particulier de réseau de neurones composé d'une partie codage et d'une partie décodage. Nous avons formulé la partie codage et décodage de la MDA comme suit :

- **Codeur :**

Dans la première couche cachée de la MDA, nous avons d'abord calculé l'incorporation non linéaire à faible dimension $H_{\text{encode}}^{(j)} \in R^{dj.n}$, pour chaque réseau $j \in \{1..N\}$.

$$H_{\text{encode}}^{(j)} = \sigma(wX^{(j)} + B^{(j)}{}_{\text{encode}}$$

Où $w \in R^{dj.n}$ et $B^{(j)}{}_{\text{encode}} \in R^{dj.n}$ sont des matrices de poids et de biais. σ est la fonction d'activation sigmoïde. Nous avons ensuite concaténé l'incorporation non linéaire de tous les réseaux et calculé une représentation de caractéristiques communes en leur appliquant plusieurs fonctions non linéaires. Il peut y avoir des couches après avoir obtenu la représentation commune de L.

$$H_{c,1} = (W_1 [H^{(1)},\ldots,H^{(N)}] + B_1)$$
$$H_{c,+1} = (W_l H_{c,l} + B_l)$$

Où $[H^{(1)},\ldots,H^{(N)}]$ sont les matrices d'activation enchaînées concaténées des couches précédentes, $l \epsilon \{1..L\}$ est le numéro de couche pour les plongements intégrés successifs.

- **Décodeur :**

On a d'abord calculé la couche commune reconstruite $\boldsymbol{H_{c,2L}}$ avec le même nombre de couches de décodage que les couches de codage. Ensuite, on peut calculer des représentations individuelles pour chaque réseau $\boldsymbol{H^{(j)}_{\text{décodeur}}}$:

$$\hat{X}^{(j)} = \sigma(W^{(j)}_{\text{décodeur},2} H^{(j)}_{\text{décodeur}} + B^{(j)}_{\text{décodeur},1})$$

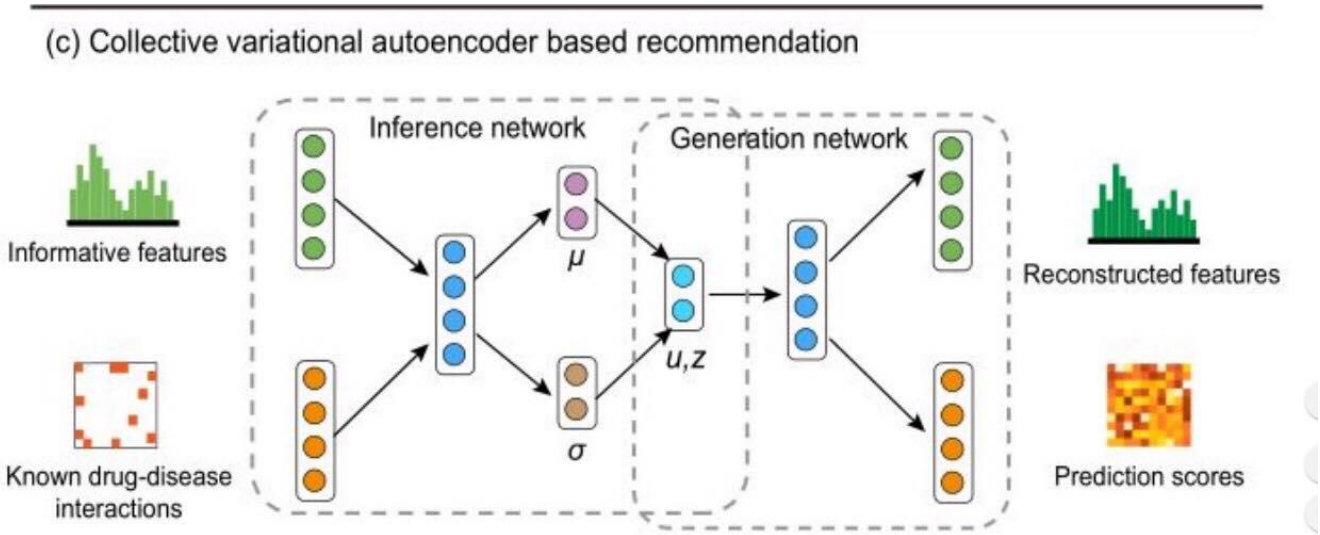
Le but de la MDA est de minimiser la perte de reconstruction entre chaque matrice PPMI originale et reconstruite, définie comme suit:

$$argmin_\theta \sum_{j=1}^{N} loss(X^{(j)}, \hat{X}^{(j)})$$

Où est la fonction d'entropie croisée binaire par exemple et $\theta = \{W^{(j)}_{codeur}, B^{(j)}_{codeur}, W^{(j)}_{décodeur}, B^{(j)}_{décodeur}, W_l, B_l\}$ pour $l \in \{1,..,2L\}$

est l'ensemble de tous les paramètres dans les parties de codage et de décodage de la MDA à apprendre dans le processus d'apprentissage. Nous choisissons l'algorithme standard de rétro-propagation pour optimiser la fonction de perte.

D. Collective Variational Autoencoder(cVAE) :



Les caractéristiques extraites de MDA servent d'informations secondaires sur les médicaments. Nous avons ensuite utilisé le modèle : Collective Variational Autoencoder (cVA) (Chen et De Rijke, 2018) pour déduire de nouvelles associations médicament-maladie. cVAE code et décode les associations médicament-maladie et les informations complémentaires via les mêmes réseaux d'inférence et de génération.

- **Réseau de génération :**

Même si les associations médicament-maladie et les caractéristiques des médicaments sont deux types d'informations différents, cVAE suppose que la sortie du réseau de génération suit des distributions différentes en fonction du type d'entrée qu'il a reçu. DeepDR a définit les associations médicament-maladie comme étant $\boldsymbol{Y}$ et les caractéristiques du médicament comme étant $\boldsymbol{X}$. Suivant la pratique courante de la VAE, on a d'abord supposé les variables latentes et suivi d'une distribution gaussienne :

$$\mu \sim N(0, I) \quad , \quad z \sim N(0, I)$$

Où $I \in R^{k \times k}$ est une matrice d'identité et constitue la dimension de la représentation de drogue latente. Motivés par le cadre d'apprentissage positif non étiqueté (PU), où

les entrées observées et non observées sont pénalisées différemment dans l'objectif, on a introduit un paramètre permettant d'équilibrer les échantillons positifs et les échantillons α négatifs. Tandis que X et Y sont introduits dans le même réseau, nous aimerions les distinguer via différentes distributions. Pour les associations médicament-maladie, la classification de la maladie j sur tous les médicaments suit une distribution de Bernoulli :

$$y_j \mid u_j \sim Bernoulli(\sigma\left(f_\theta(u_j)\right))$$

Ceci définit la fonction de perte lors de l'alimentation d'associations médicament-maladie, c'est-à-dire le log-vraisemblance logistique pour la maladie:

$$logp_\theta\left(y_j \mid u_j\right) = \sum_{i=1}^{n} \alpha . y_{ji}\ log\sigma(f_{ji}) + (1 - y_{ji})\log\left(1 - \sigma(f_{ji})\right)$$

Où f_{ji} est le i-ème élément du vecteur $f_\theta(u_j)$ et $f_\theta(u_j)$ est normalisé par une fonction sigmoïde.

- **Réseau d'inférence :**

La log-vraisemblance de cVAE est intraitable en raison des transformations non linéaires du réseau de génération. Nous avons donc eu recours à l'inférence variationnelle pour approximer la distribution. L'inférence variationnelle se rapproche du véritable postérieur intraitable avec une distribution variationnelle plus simple. Nous avons suivi l'hypothèse du champ moyen $q\ (U, Z)$. En définissant une distribution $q\ (U, Z)$ gaussienne entièrement factorisée:

$$q\ (U, Z) = \prod_{j=1}^{m} q(u_j) \sum_{j=1}^{d} q(z_j)$$

Avec $q(u_j) \sim N(\mu_j, diag(\sigma^2{}_j))$, $q(z_j) \sim N(\mu_m, diag(\sigma^2{}_{m+j}))$

De plus, nous avons remplacé les paramètres de variation individuels par des fonctions dépendantes des données par un réseau d'inférence paramétré par Ø i.e ..,$f_\emptyset$ où μ_j et σ_j sont générés comme suit:

$$\mu_j = \mu\left(f_\emptyset(y_j)\right) \ , \ \sigma_j = \sigma\left(f_\sigma(y_j)\right) \quad , \forall j = 1..m$$
$$\mu_{m+j} = \mu\left(f_\emptyset(x_j)\right) \ , \ \sigma_{m+j} = \sigma\left(f_\sigma(x_j)\right) \quad , \forall j = 1..m$$

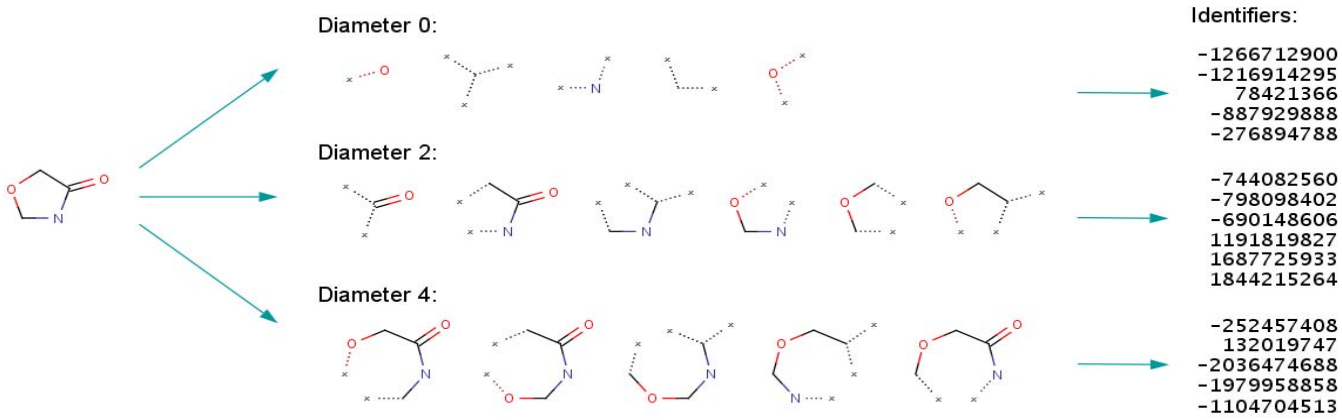
IV. 11ième Network :

Deux fichiers sont fournis par le laboratoire bioinformatique : un fichier qui contient les noms de « cidals »(une substance qui tue les bactéries) et leurs codes et l'autre contient le nom des médicaments et leurs codes.

A l'aide de librairie python rdkit et à partir des codes des molécules ,on a calculé pour chacune les « morgan fingerprints » : Les empreintes moléculaires digitales qui sont un moyen de coder la structure d'une molécule.

La comparaison d'empreintes digitales nous permet de déterminer la similarité entre eux.

On a alors transformé des données chimiques en deux vecteurs pour continuer le travail de DeepDR.



Ensuite, pour calculer la similarité, on a utilisé le coefficient de Tanimoto. Une matrice de similarité est donc construite.

Finalement, on a repris la routine de prétraitement pour obtenir la 11ième matrice PPMI et on a refait le calcul avec DeepDR.

V. *Résumé :*

J'ai commencé mon stage par la reproduction de deepDR qui se résume en 3 étapes :

1- Convertir la structure topologique de chaque network en une représentation matricielle en appliquant d'abord la méthode de marche aléatoire avec retour en arrière (Random walk with restart RWR) et puis la construction des matrices PPMI qui capture l'information structurel du network.
2- Fusionner les matrices PPMI en une matrice compacte de faible dimension commune pour toute les Networks à l'aide de l'MDA d'une manière non supervisée.
3- Alimenter la matrice de la deuxième étape dans le VAE pour le prétraitement puis en raffine le VAE en alimentant la Network d'association médicament-maladie. C'est ce qu'on appelle cVAE.

Ensuite j'ai construit une 11ème Network et finalement j'ai reproduit les pipelines de deepDR avec ces 11 Nets.

VI. *Résultat :*

L'obtention de deux matrices Z-score : une avec l'Input de 10 Nets et l'autre en ajoutant la 11ième.

Z-score est une matrice qui contient les probabilités qu'un médicament existant s'associe à une telle maladie.

La comparaison de ces deux matrices et l'étude du résultat est en cours d'analyse.

VII. Conclusion :

En guise de conclusion, ce stage est une occasion de s'approfondir dans une des applications les plus puissantes de l'IA dans la bioinformatique. Cet accord a accéléré le traitement et l'analyse de grandes quantités de données. En effet, la grande puissance de prédire et d'analyser permet de gagner du temps et d'être efficace dans les recherches. Les professionnels de la santé pourront ainsi améliorer la qualité de vie grâces à la mise en place rapide de nouveaux traitement médicaux efficaces.

Ce stage n'est qu'une initiation au domaine du big data, qui m'a permis d'étudier cette performance impressionnante de l'IA. Cette victoire de la BigData m'a sollicité à pénétrer dans ce monde.