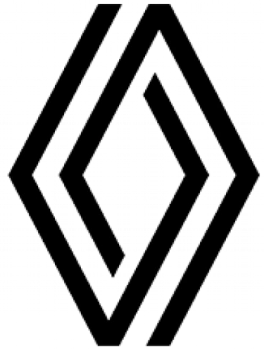


Applications de l'IA générative dans les control towers Renault



RENAULT

Nicolas MENOURET
Murilo MOURA
Ping'an YANG
Hamza ZERHOUNI

ENPC 2023-2024

Ce projet a été réalisé dans le cadre du module « Projet Data Science » 2023–2024 de l'ENPC

Équipe d'élèves



Murilo
MOURA - GI



Nicolas
MENOURET - IMI



Pingan
YANG - GI



Hamza
ZERHOUNI - IMI

Encadrants



Christian SERRANO
- Renault



Mohamed Saâd EL
HARRAB - ENPC

Notre démarche:

1. Compréhension du fonctionnement des control towers de Renault lors de réunions avec les équipes inbound et outbound;
2. Approfondissement sur les méthodes d'IA génératives plus modernes et recherche sur les applications dans l'industrie;
3. Définition des cas d'utilisation où l'IA générative pourrait apporter une valeur ajoutée aux control towers;
4. Sélection des cas d'utilisation plus prometteurs et rédaction de la feuille de route pour les mettre en œuvre.

Les Control Towers Renault



Image générée par Dall-E 3

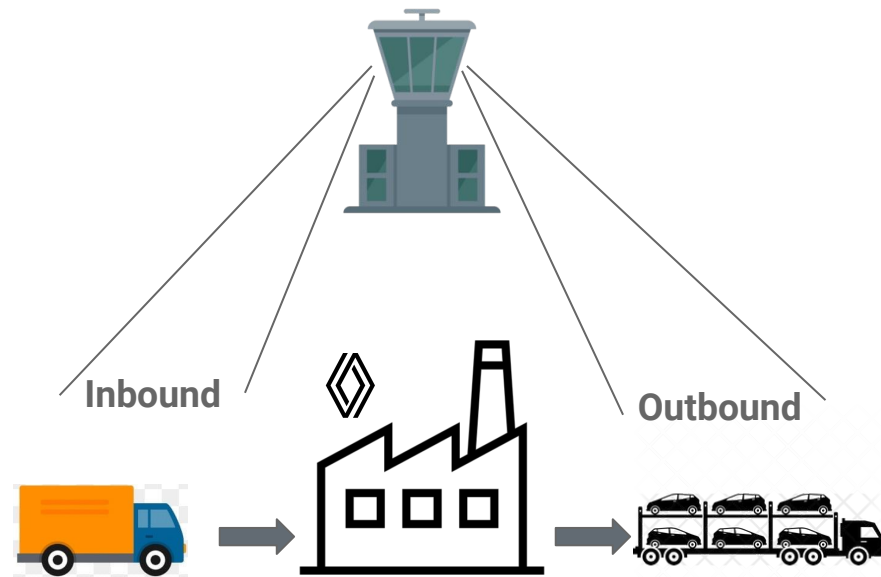
Les Control Towers Renault constituent un outil favorisant la visibilité des flux et la prise de décision

Objectifs principaux :

- Avoir une visibilité en temps réel des flux de transport du groupe Renault
- Anticiper les problèmes liés aux retards de livraison et proposer des solutions
- Contribuer à la satisfaction client

Données disponibles :

- Besoin en pièces des usines et niveaux de stock (inbound)
- Commandes de concessionnaires (outbound)
- Position géographique et prévision d'arrivée des camions
- Prévisions météorologiques
- Situation du trafic



Question : où l'IA Générative pourrait apporter une valeur ajoutée dans le fonctionnement des control towers?

L'Intelligence Artificielle Générative

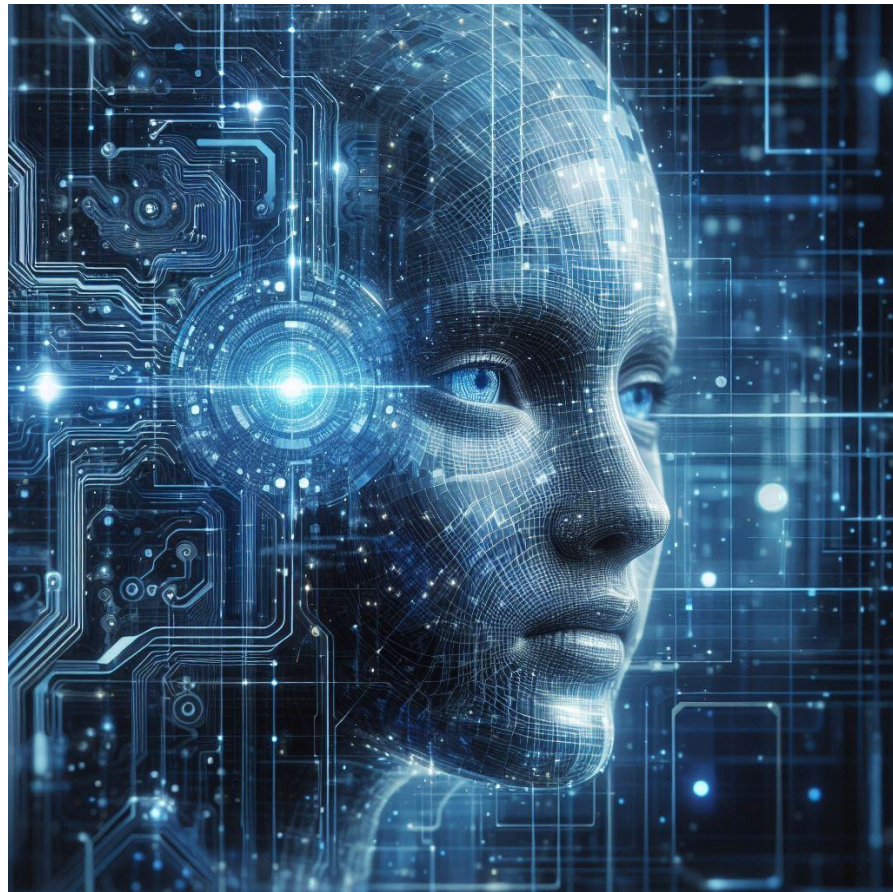


Image générée par Dall-E 3

Le principe de l'IA Générative

Machine Learning Traditionnel :

- Est utilisé la plupart du temps pour des tâches de classification/régression.
- Apprend à extraire des informations significatives (patterns) des données disponibles.
- Nécessite la plupart du temps des données labellisées pour l'entraînement.
- Peut fonctionner sur un volume restreint de données.

IA Générative :

- Est utilisée pour la création de nouvelles données.
- Apprend la distribution sous jacente des données.
- Ne nécessite pas une labellisation des données pour l'entraînement.
- Nécessite un grand (très grand) volume de données pour fonctionner.

L'IA générative est surtout utilisée pour la création des procédés **text-to-text** ou **text-to-image** qui permettent la génération d'images ou de textes à partir d'un texte entré par l'utilisateur (ChatGPT, Google Bard, Dall-E, Midjourney, Stable Diffusion...).

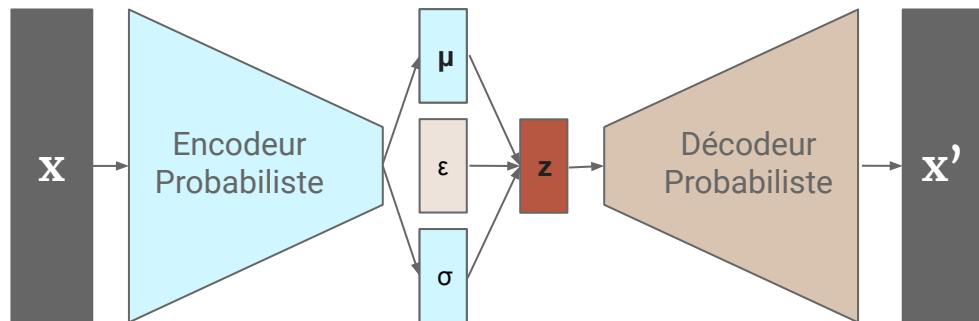
Des exemples de modèles d'IA généraliste.

Variational Auto Encoders

Les VAEs sont des modèles d'IA génératives se basant sur **trois parties**:

- **Un espace latent** qui représente la version compressée et probabiliste des données d'entrée.
- **Un encodeur** qui va apprendre à déterminer les paramètres (moyenne et variance) d'une distribution de probabilité dans l'espace latent.
- **Un décodeur** qui va apprendre reconstruire la donnée d'entrée à partir d'un point de l'espace latent.

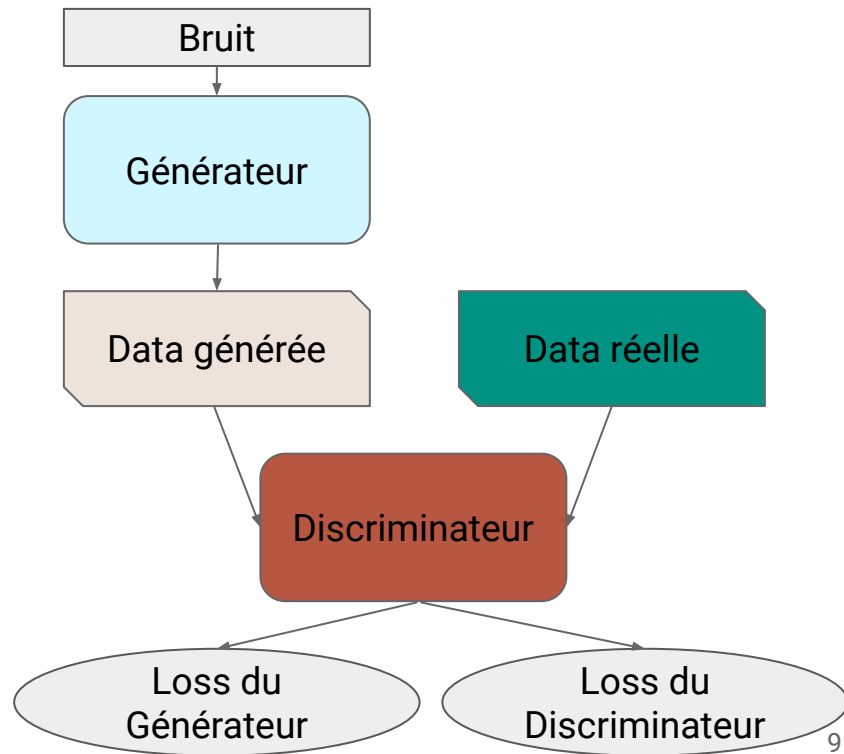
Cette approche probabiliste permet la génération de nouvelles données en échantillonnant dans l'espace latent.



Generative Adversarial Networks

Les GANs sont des méthodes d'IA générative se basant sur l'**entraînement simultanée de deux modèles** (un générateur et un discriminateur) en **compétition**:

- **Le Générateur** apprend à simuler des données impossibles à distinguer des données réelles pour le discriminateur.
- **Le Discriminateur** apprend à différencier les données réelles des données engendrées par le générateur.



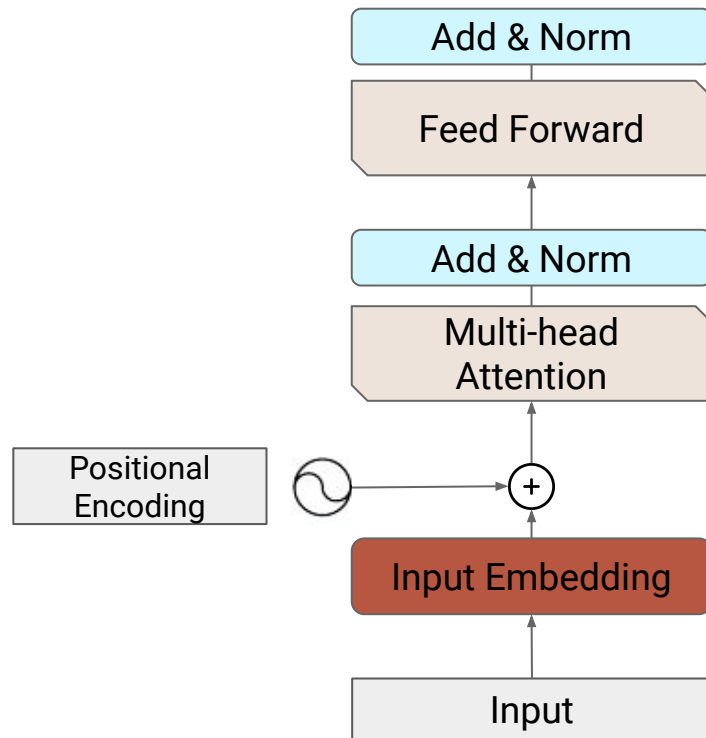
Transformers

Le Transformer est le premier modèle de transduction reposant entièrement sur le mécanisme d'auto-attention pour calculer les représentations des inputs utiliser de RNN ou de convolution.

- **BERT:** Modèle utilisant l'architecture Masked Language Model (MLM), il se distingue par son approche bidirectionnelle, prenant en compte l'ensemble du contexte d'une séquence de mots simultanément. Pendant l'entraînement, BERT masque 15% des mots de chaque séquence pour prédire ces mots masqués en se basant sur le contexte fourni par les mots non masqués

- **GPT:** Modèle auto-supervisé utilisant plusieurs fois les couches du Transformer et lisant le texte de gauche à droite pour prédire le mot suivant.

10

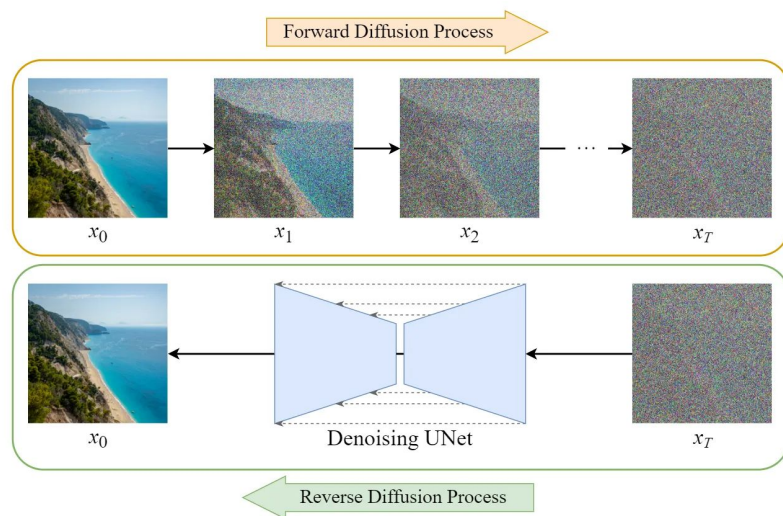


Stable Diffusion

Les méthodes par Stable Diffusion se compose de **deux parties**:

- Un processus de **diffusion Forward** qui consiste à bruiteur la donnée d'entrée étape par étape
- Un processus de **diffusion inverse** qui consiste à reconstruire la donnée d'entrée en débruitant le résultat obtenus à l'étape précédente.

Cette méthode est parmi les meilleurs pour la génération d'image.



Medium ([Stable Diffusion Clearly Explained](#))

Trois exemples d'utilisations pour les Control Towers Renault

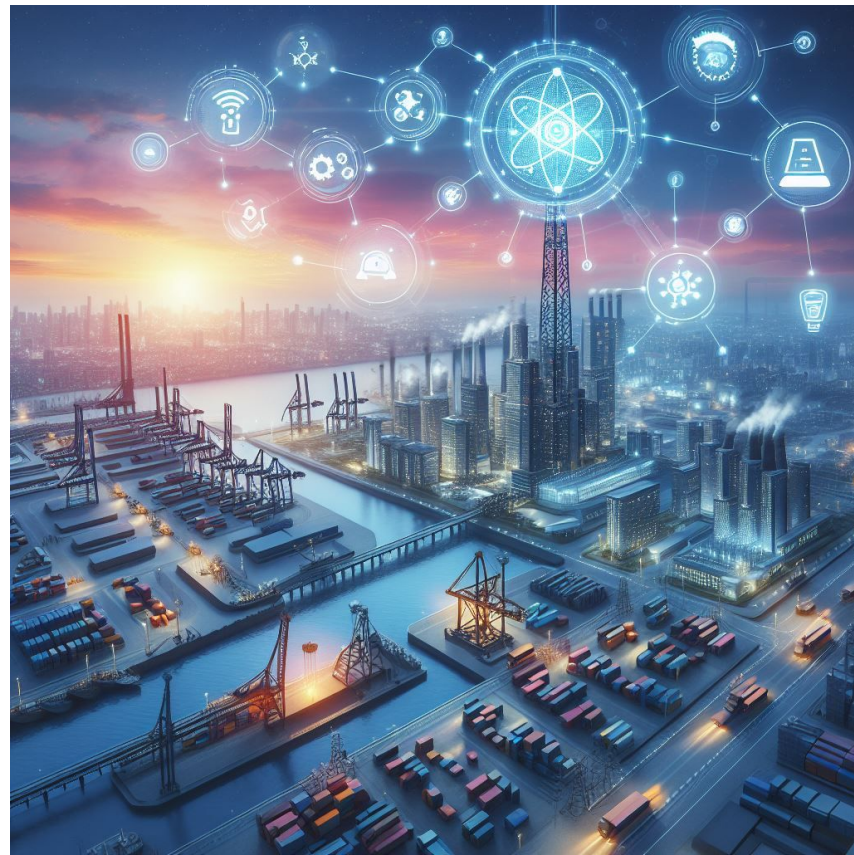


Image générée par Dall-E 3

Cas d'utilisation n°1: Explications en langage naturel des scénarios Inbound par IA générative.

Principe

Inputs

Données relatives à la situation:

- Coût et ETA calculé par l'algorithme
- La situation de météo et trafic
- Le type de pièce
- Position géographique de l'usine



Output

Texte apportant une analyse et une interprétation des scénarios et qui explique le choix d'un scénario pour cette situation

Exemple: "Lors de situation similaire rencontrée, avec de la neige et un accident sur une autoroute, il est préférable de prioriser la diminution des coûts, car ..."

Objectif: Prévoir une zone de texte dans l'onglet des scénarios proposés pour afficher l'analyse de l'IA générative.

Données à récolter/utiliser

- Données historiques de choix de scénario:
 - Coût effectif de scénario
 - ETA de scénario
 - Choix de scénario
 - Type de pièce
 - Source de la pièce: entrepôt fournisseur, usine Renault, autre camion
 - Position géographique de l'usine
 - Donnée sur la situation lors de la prise de décision (trafic, météo, accidents...)
- Données à récolter pour les futures prises de décision: hypothèses sur les éventuels scénarios et leurs informations respectives.

Implémentation

1. Générer une base de données texte pour décrire la situation et le choix de scénario

Exemple: “Lors de situation similaire rencontrée, avec de la neige et un accident sur une autoroute, le scénario choisi est de prioriser la diminution des coûts.”

Modèle: modèle de langage pré-entraîné comme GPT-3 capable de générer du texte de manière cohérente.

Input: requête sous forme de texte à envoyer au modèle

Génération des descriptions: soumettre les requêtes et utiliser les réponses comme descriptions des données

2. Prédire le niveau de succès des scénarios choisis pour les différentes situations à venir

Input: score sur le succès des scénarios futures et informations relatives aux scénarios

Modèle: Machine Learning (Random Forest / XGBoost) ou Deep Learning (NN) pour la prédiction du succès des scénarios

3. Améliorer la description des situations futures en indiquant le scénario avec le plus de succès

Exemple: “Pour l’incident qui a eu lieu à ... pour livrer les pièces ... de l’usine ... à l’usine ..., la solution fournisseur a été choisi parce qu’elle permet d’avoir un coût de .. avec un retard de Cette solution a été la bonne solution en permettant de livrer les pièces nécessaires dans la même journée.”

Avantages

Utilisation d'un Modèle de Langage Pré-Entraîné (GPT-3) : L'utilisation d'un modèle de langage pré-entraîné comme GPT-3 pour générer des descriptions de situations et de choix de scénario offre l'avantage de produire du texte cohérent et contextuellement pertinent. Cela peut améliorer la qualité des données générées.

Modèles de Prédiction (Random Forest / XGBoost, NN) : Le recours à des modèles de prédiction tels que Random Forest, XGBoost, ou des réseaux de neurones (NN) pour évaluer le niveau de succès des scénarios futurs est une approche solide. Ces modèles peuvent exploiter efficacement les données historiques pour fournir des prédictions utiles.

Approche Itérative : Le plan propose une approche itérative en générant d'abord des données textuelles, puis en les utilisant pour former des modèles de prédiction. Cela permet une amélioration continue en ajustant les modèles en fonction des résultats obtenus.

Inconvénients

Dépendance aux Données Historiques : La qualité des prédictions dépend fortement de la qualité des données historiques. Des anomalies ou des changements significatifs dans les conditions futures peuvent affecter la validité des modèles.

Complexité de l'Interprétation des Modèles : Les modèles de prédiction, en particulier les réseaux neuronaux, peuvent être complexes à interpréter. Il est important de comprendre comment les modèles prennent leurs décisions pour garantir une utilisation et une adaptation appropriées.

Besoin de Données Abondantes : Les modèles de prédiction, en particulier les modèles de deep learning, nécessitent souvent des ensembles de données volumineux pour obtenir des performances optimales.

Synthèse

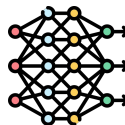
1. Collecte et Préparation des Données Historiques :

- Collecter des données historiques
- Récolter les données des futures scénarios considérés et donner leur donner un score pour évaluer leur succès



2. Implémentation des Modèles de Prédiction :

- Utiliser un modèle de langage pré-entraîné, tel que GPT-3, pour générer des descriptions de situations et de choix de scénario à partir des données historiques.
- Entraîner des modèles de prédiction, pour évaluer le niveau de succès des scénarios futurs.



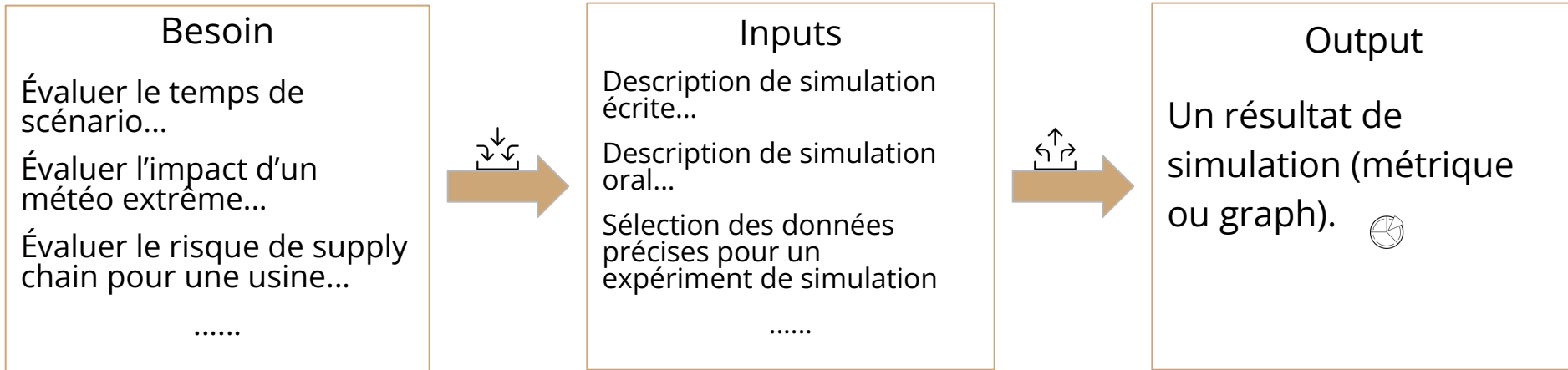
3. Intégration des Résultats dans le Processus de Prise de Décision :

- Intégrer les résultats des modèles de prédiction dans le processus de prise de décision, en fournissant des recommandations basées sur l'analyse des scénarios futurs.
- Mettre en place un suivi continu pour ajuster les modèles en fonction des nouvelles données et des évolutions du contexte opérationnel.



Cas d'utilisation n°2: Simulations par langage naturel grâce à l'IA générative.

Principe



Intégration dans les process Renault:

Mettre en place un système de simulation où on peut lancer facilement la simulation pour n'importe quel genre de besoin.

Données à récolter/utiliser

Pour générer de une simulation sur à partir de n'importe quel besoin, il faut deux types de base de données:

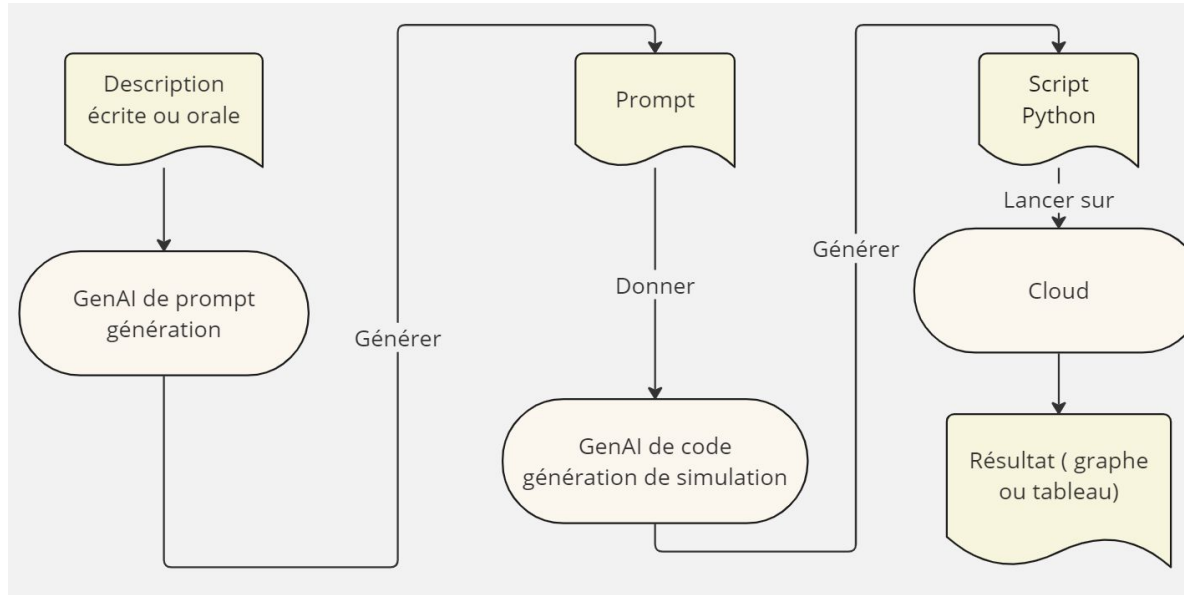
Une base de données pour **fine tuning** une GenAI:

- Des prompts (descriptions) bien structurés
- Des codes de simulation relatifs

Une base de données relative à la simulation:

- Des **données** et leurs **distributions statistiques** concernée comme la vitesse de camion, le lieu de départ, la destination, etc.
- Des **données obtenus de la control tower Renault** comme les zones rouges, la situation d'urgence des pièces, etc.
- Des **données récupérées par API** comme des données de trafic sur les routes, des données météorologiques ou des accidents sur les routes, etc.

Implémentation



Deux modèle de GenAI à utiliser pour garantir la performance:

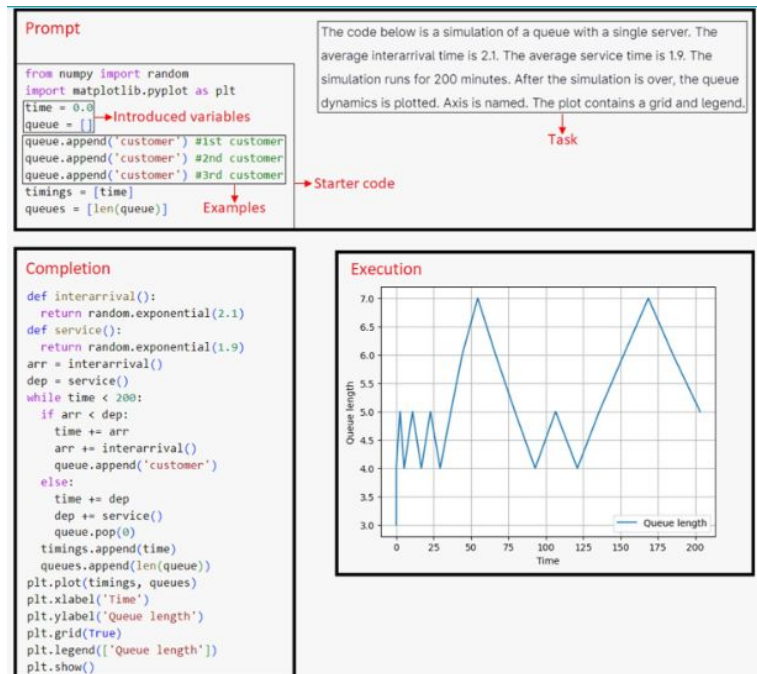
- Un GenAI **existant** pour mieux comprendre le besoin de simulation et faciliter l'utilisation de tous
- Un GenAI à **fine tuning** pour générer le code pour la simulation

Avantages

Un papier du MIT sorti en 2023 démontre que **GPT3 est capable de générer du code valide** pour simuler le comportement des files d'attente et la gestion du stock. [Lien](#)

Cette approche permettrait notamment:

- L'utilisation de données qui sont **déjà disponibles** dans la control tower
- Haut degré de **personnalisation** du prompt
- **Agilité** dans la mise en oeuvre d'une simulation
- Habilitier un utilisateur opérationnel à lancer des simulations même si il n'a pas de connaissances en programmation



Exemple présent dans le papier du MIT

Inconvénients

- Cette solution requiert un **environnement d'exécution qui est extérieur** à la control tower et n'existe pas pour l'instant
- Le code généré par la gen IA **peut avoir des bugs** pas forcément identifiables lors de l'exécution
- L'intégration d'un modèle de langage avec les données Renault peut avoir des **challenges de confidentialité** et conformité RGPD
- La solution est limitée par la **qualité des modèles pré codés**, qui aussi peut être trop **simpliste**

Synthèse

1. Réaliser des sessions avec les équipes opérationnelles pour identifier ses besoins en matières de simulation et ensuite développer les modèles pré codés en interne.



2. Il existe des outils, comme **Dataiku**, qui facilitent l'intégration des données internes avec des modèles de langage. (Renault a déjà un abonnement)



data
iku

3. La flexibilité et agilité de l'approche font de ce use case une perspective prometteuse, mais il est nécessaire de rester vigilant à la sécurité des données et à la qualité du modèle pré codé qui tourne derrière.

Illustration – ChatGPT3.5 + Python

Prompt

Modèle pré-codé pour le routage
des véhicules en utilisant l'API
Google Maps en Python



M

You

write a simulation for a trip between Paris and Marseille. Plot the value of the average duration of the trip in function of the departure time throughout the day. Consider the worst case scenario.

Illustration – ChatGPT3.5 + Python

Output

Remplacement des paramètres du modèle pré codé

```
origin_query = "Renault Velizy"  
destination_query = "École des Ponts"  
sim_mode = "best_guess"
```

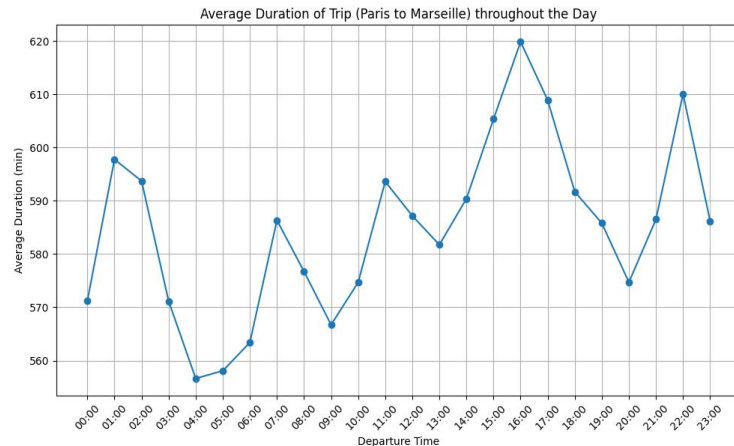


```
# Origin and destination queries  
origin_query = "Paris, France"  
destination_query = "Marseille, France"  
# Simulation mode for worst-case scenario  
sim_mode = "pessimistic"
```

Écriture de la simulation demandée avec un graphe pour afficher le résultat

```
# Departure times throughout the day  
departure_times = [datetime(2024, 1, 1, hour=h) for h in range(24)]  
  
# List to store average durations  
average_durations = []  
  
# Perform simulations for each departure time  
for departure_time in departure_times:  
    directions_result = gmaps.directions(origin_point,  
                                         destination_point,  
                                         mode="driving",  
                                         language="en",  
                                         traffic_model=sim_mode,  
                                         departure_time=departure_time)  
  
    # Extract duration in minutes  
    duration = directions_result[0]['legs'][0]['duration_in_traffic']  
    average_durations.append(duration)  
  
# Plotting the results  
plt.plot(departure_times, average_durations, marker='o')  
plt.xlabel('Departure Time')  
plt.ylabel('Average Duration (minutes)')  
plt.title('Average Trip Duration from Paris to Marseille (Worst Case Scenario)')  
plt.xticks(rotation=45)  
plt.tight_layout()  
plt.show()
```

Exécution du code généré



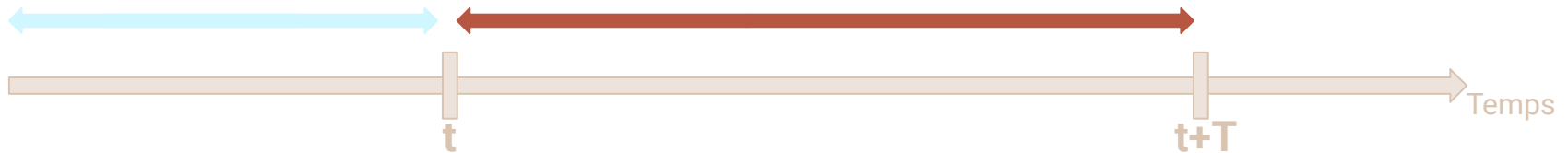
Cas d'utilisation n°3: Méthodes de simulation par IA génération et robustesse des scénarios Inbound.

Principe

Control Tower InBound
actuelle

Information passée
donc **connue**.

Information future donc **inconnue** sur la
période T . Pendant ce laps de temps, **la
situation peut évoluer** et donc affecter la
performance du scénario choisi.



1. Accident au temps t .
2. Estimation des scénarios en se basant sur la situation au temps t .
3. Choix du scénario.

Pièce livrée: Problème résolu au
temps $t+T$.

Principe

Control Tower InBound avec
simulations par IA générative

Simulations de différentes évolutions de la
situation par IA générative sur la période T.

Information passée
donc connue.



1. Accident au temps t .
2. Estimation des scénarios en se basant sur la situation au temps t .
3. **Simulations de différentes situations futures et évaluation robustesse du scénario.**
4. Choix du scénario.

Pièce livrée: Problème résolu au
temps $t+T$.

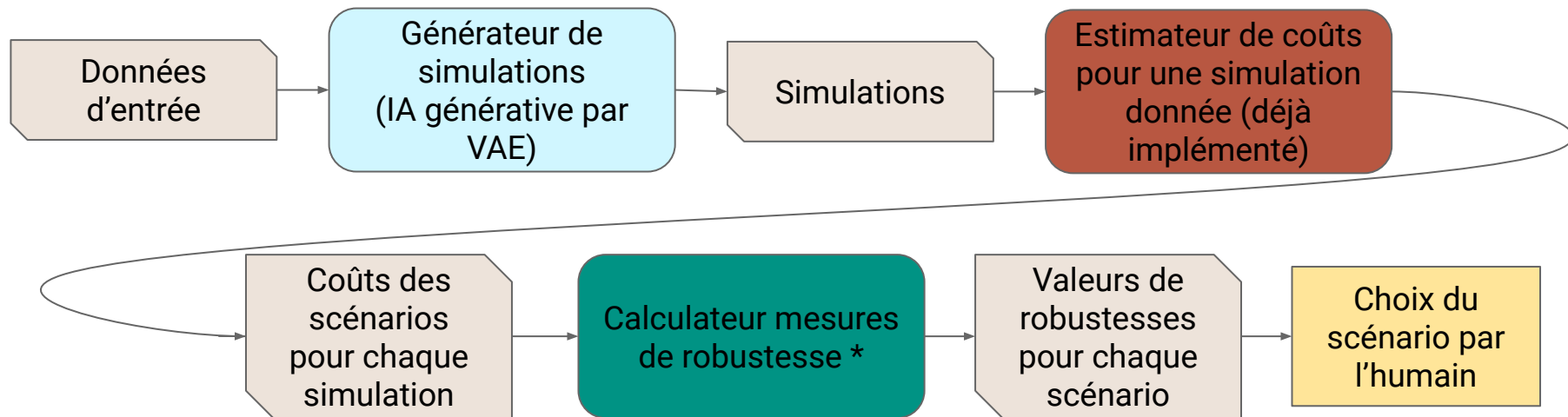
Données à récolter/utiliser

Pour générer de telles simulations, on a besoin d'accéder aux informations nécessaires **sous la forme de Time Series**. Ainsi, à l'instant t , on peut se baser sur les informations d'une période T_p antérieure à l'instant t pour simuler l'évolution de la situation sur la période T à venir.

Les informations requises sont de plusieurs types:

- Les **informations issues d'un historique de la control tower Renault** comme les zones rouges en fonction du temps.
- Des **informations accessible par API** (ou directement depuis la base de données Renault) comme des données de trafic sur les routes en fonction du temps, des données météorologiques ou encore des taux d'accidents pour chaque route.
- Des **informations relatives à la situation concernée** (point de départ, point d'arrivée, type de problème...) qui seront utiles pour "cibler la simulation", on ne va pas simuler des situations pour la région Bretagne si le problème se situe à Montpellier.

Implémentation



* Une fois les situations simulées, on peut **évaluer la robustesse des scénarios** établis au temps t par:

- L'écart moyen des coûts simulés à l'instant $t+T$ avec le coûts prédit à l'instant t . (**précision de la prédiction**)
- Le pourcentage de scénarios où le coût simulé à l'instant $t+T$ est supérieur au coût prédit à l'instant t (**risque de dépassement**).
- Valeur maximale/minimale des coûts simulés à l'instant $t+T$ (**pire des cas/meilleur des cas**).

Avantages

Les méthodes de simulation par IA génératives présente l'avantage majeur d'être **à la pointe de la technologie pour l'analyse, la simulation et la prédiction de Time Series**. Plusieurs articles scientifiques sont disponibles sur le sujet:

- [Diffusion Models for Time Series Applications: A Survey](#)
- [TimeVAE: A Variational Auto Encoder For Multivariate Time Series Generation](#)
- [Time-series Generative Adversarial Networks](#)

Des exemples d'implémentations sont même disponibles pour certaines méthodes:

- [TimeVAE implementation \(Github\)](#)

Le secteur de l'IA générative est en pleine croissance, il y a donc de nombreuses nouveautés régulièrement.

Enfin, ce cas d'utilisation dans sa globalité permet une **meilleure compréhension de la situation** et des scénarios proposés ce qui aboutit une **meilleure prise de décision**.

Inconvénients

L'implémentation d'une méthode de simulation par IA générative présente **plusieurs inconvénients** pour une entreprise comme Renault:

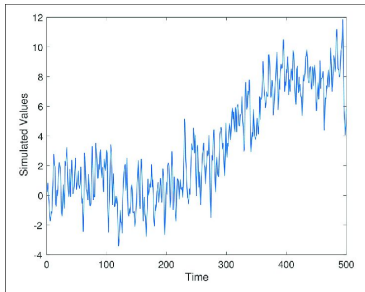
- Ces modèles sont **complexes aussi bien à comprendre qu'à implémenter correctement**. Ils demandent donc un vrai investissement en temps pour les data scientists de l'entreprise.
- Ces modèles de simulation par IA générative **requièrent une immense quantité de données** pour être entraîné, ce qui peut être un point critique pour notre cas d'utilisation.
- Ces modèles sont **très gourmands en ressources de calculs**, ils ont donc un coût financier élevé.

Ce dernier point pose un vrai **problème de rentabilité** à implémenter ces méthodes en interne pour l'entreprise.

Synthèse

Trois étapes pour la suite:

1. Évaluer en interne **les coûts de développement** d'une telle méthode et **déterminer d'autres utilisations** possibles de ces méthodes de simulation par IA générative.



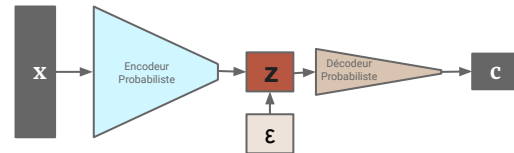
2. Se renseigner sur la possibilité d'**intégrer des outils déjà existants** comme [YData](#) dans les process Renault.



3. Vers un **modèle plus simple**?

On peut entraîner un **modèle inspiré d'un VAE** qui retourne un **"coût réel estimé"** à partir des données d'entrée précédemment utilisée et d'un **historique des coûts réels** de chaque scénarios.

En guise de simulations, on encode notre situation au temps t , puis on la **"bruite"** plusieurs fois pour retourner plusieurs coûts estimés.



Conclusion



Image générée par Dall-E 3

IA générative chez Renault: 3 cas d'utilisation

L'IA générative permet de **générer de nouvelles données réalistes et cohérentes**. Pour cela, elle nécessite un **très grand nombre de données** d'entraînement pour apprendre la distribution sous jacente. Il est d'usage d'utiliser des modèles préexistants et de les **fine tuner**. C'est un domaine récent et en pleine croissance.

1. Explications en langage naturel des scénarios Inbound par IA générative:

- Permettra une meilleur prise de décision pour les scénarios Inbound.

- Nécessitera une base de données historiques sur les scénarios sélectionnés.

2. Simulations par langage naturel grâce à l'IA générative:

- Permettra de faire des simulations rapidement et intuitivement.

- Nécessitera un environnement d'exécution qui n'existe pas pour l'instant.

3. Méthodes de simulation par IA générative et robustesse des scénarios Inbound:

- Permettra une exploration de la robustesse des scénarios proposés.

- Coûteux et complexe à implémenter.

Bilan du projet

Ce projet a été pour nous l'occasion de:

- D'approfondir nos connaissances sur l'intelligence artificielle générative, un secteur en pleine croissance qu'il sera inévitable de maîtriser à l'avenir.
- Découvrir les control towers Renault et d'apprendre à penser des utilisations concrètes et pratiques.
- Explorer des cas d'utilisation concrets de l'IA générative et certaines méthodes d'implémentation.