



Filière : Génie Informatique

COLLECTE DE DONNEES ET FINE-TUNING POUR UN SYSTEME INTELLIGENT D'ASSISTANCE TECHNIQUE

Entreprise d'Accueil : FORGES DE BAZAS



Période de Stage : 01 Juillet 2024 – 09 Septembre 2024

Réalisé Par : Ali EL-AZZAOUY

Encadré Par : M. Ahmed Salim LACHKER

Année Universitaire : 2023 / 2024

DEDICACES

A qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit,

A toi mon cher Père MOHAMMED

Mon amour et ma profonde reconnaissance,

A celle qui m'a toujours motivé par, son amour, son soutien, par tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie

A toi ma chère Mère FADMA

Toutes mes joies, mon amour et de mon éternelle gratitude, Et mes frères ABDALILAH, AHMAD, ma sœur JAMILA pour leurs amours et leurs encouragements.

A tous mes amis de l'AIAC

, Qu'ils trouvent en ce travail, l'hommage de ma gratitude, pour leur présence et les bons moments passés durant cette année.

*A tous mes amis **Forges de Bazas**, pour le travail que nous avons commencé ensemble, et les bons moments que nous avons passé ensemble.*

À toutes les personnes qui ont marqué ma vie, et qui ont fait de toutes ces années des instants de joie et de bonheur, merci.

REMERCIEMENT

Nous tenions à remercier de prime abord Dieu pour sa bonté et sa gratitude dans lesquelles nous ne pourrions en aucun cas être ici.

Il nous est agréable d'exprimer ma reconnaissance auprès de toutes les personnes, dont l'intervention au cours de ce projet, de près ou de loin, a favorisé son aboutissement.

Un grand merci à **Mr. Lachkar Ahmed Salim** notre encadrant à **Forges de Bazas**, pour ses conseils, sa confiance pendant cette aventure, son encadrement de haut niveau, et l'intérêt particulier qu'il a porté à notre travail malgré ses préoccupations

Nous tenions à remercier Forges de Bazas, pour nous avoir accueilli pendant ce stage pour ce fabuleux projet auquel nous avons participé, pour tout ce que nous avons vu et vécu durant ces deux mois.

Ensuite, nous remercions l'ensemble des collaborateurs de Forges de Bazas pour leur sympathie, leur soutien, ainsi que les multiples conseils qu'ils nous ont prodigué au cours du stage, et qui nous ont formé et initié à la vie professionnelle en s'assurant que le stage se passe dans les meilleures conditions.

Nous témoignons aussi notre reconnaissance à nos professeurs et l'Académie Internationale Mohammed VI De l'Aviation Civile, pour les enseignements qu'ils nous ont apporté durant notre formation.

Un grand merci pour leurs aides, soutiens, suivis permanent et conseils précieux.

Table des Matières

DEDICACES	3
REMERCIEMENT	4
LISTE DES FIGURES.....	8
INTRODUCTION.....	10
CHAPITRE I : PRESENTATION DE L'ORGANISME D'ACCUEIL	11
1. INTRODUCTION :.....	12
2. CONTEXTE DE L'ORGANISME D'ACCUEIL :	12
2.1. Aperçu général sur l'entreprise Forges de Bazas :	12
2.2. Fiche technique de Forges de Bazas :	13
2.3. Présentation de fournisseurs :	14
2.3.1. L'entreprise SANY :	14
2.3.2. L'entreprise TOYOTA Material Handling :	15
2.3.3. Organigramme de l'entreprise :	16
2.4. Présentation de département d'accueil :	16
3. CONCLUSION :	17
INTRODUCTION GENERALE	18
1. CONTEXTE ET PROBLEMATIQUE :	18
2. OBJECTIFS DU PROJET :	19
3. METHODOLOGIE DE TRAVAIL :	20
CHAPITRE II : BENCHMARKING DES OUTILS, COMPRÉHENSION DES ARCHITECTURES DE LEURS APIS	23
1. INTRODUCTION :	24
2. DESCRIPTION ET ANALYSE DES OUTILS :	24
3. CHOIX DES CRITERES DE COMPARAISON :	29
4. TABLEAU COMPARATIF DES OUTILS GENERATIFS :	31

5.	ETUDE DE L'ARCHITECTURE DES APIS :	32
5.1.	Définition de API :	32
5.2.	Les modèles de langage IA pour le texte avec API publique :	33
5.3.	Architecture de Chat GPT :	33
5.4.	Les éléments constitutifs de ChatGPT :	35
6.	CONCLUSION :	36
CHAPITRE III : COLLECTE, COMPRÉHENSION, PRÉPARATION ET STRUCTURATION DES DONNÉES.....		38
1.	INTRODUCTION :	39
2.	IMPORTANCE DES DONNEES DANS L'ENTRAINEMENT DES MODELES :	39
3.	SOURCES DE DONNEES :	39
4.	METHODOLOGIES DE COLLECTE :	40
4.1.	Définition de GROQ :	40
4.2.	Pourquoi choisir le modèle LLAMA-3.1-70B-Versatile ?.....	41
4.3.	Problèmes et obstacles trouvés :	41
4.4.	Structuration des Données :	42
5.	CONCLUSION :	42
CHAPITRE IV : MODÉLISATION, ÉVALUATION ET DÉPLOIEMENT DES MODÈLES.....		43
1.	INTRODUCTION :	44
2.	MODELISATION :	44
2.1.	Utilisation de Hugging Face et Kaggle :	44
2.2.	Choix des modèles :	44
2.3.	Fine-Tuning du Modèle GPT-2 :	45
2.3.1.	Sélection du Modèle GPT-2 Small :	45
2.3.2.	Evaluation du modèle avant le fine-tuning :	47
2.3.3.	Fine Tuning du Modèle GPT-2 Small :	47
2.3.4.	Fine-Tuning du Modèle LLAMA 2 :	48
2.3.5.	Fine-Tuning du Modèle Gemma :	55
3.	DÉPLOIEMENT DES MODELES :	60

4. CONCLUSION :	60
CHAPITRE V : CREATION D'UN PROTOTYPE DE PIPELINE POUR ALIMENTER UN TABLEAU DE BORD EN TEMPS REEL	61
1. INTRODUCTION :	62
2. QU'EST-CE QUE L'ETL ?	62
3. BESOIN :	63
4. ARCHITECTURE DU SYSTEME :	63
5. TECHNOLOGIES UTILISEES :	64
5.1. HDFS :	64
5.2. Spark :	64
5.3. Kafka :	64
5.4. MySQL :	65
6. RESULTATS :	66
CONCLUSION GENERALE	68
WEBOGRAPHIE	70

Liste des Figures

Figure 1 : Logo de FORGES DE BAZAS	12
Figure 2 : Fiche Technique de FORGES DE BAZAS	13
Figure 3 : Portée Mondiale de SANY	15
Figure 4 : Organigramme de l'entreprise	16
Figure 5 : Méthodologie CRISP-DM	21
Figure 6 : Architecture de Chat GPT	33
Figure 7 : Architecture du modèle Transformer.....	35
Figure 8 : Tableau comparatif des modèles	41
Figure 9 : Logo de Hugging Face.....	44
Figure 10 : Logo de Kaggle.....	44
Figure 11 : Logo de OpenAI GPT-2	45
Figure 12 : Logo de LLAMA2.....	45
Figure 13 : Logo de Gemma	45
Figure 14 : Les différents tailles de GPT-2	46
Figure 15 : Métriques de performance pour divers datasets	47
Figure 16 : Comparaison des méthodes de fine-tuning : Full Finetuning, LoRA et QLoRA ..	51
Figure 17 : Résultats de l'entraînement du modèle.....	53
Figure 18 : Perte d'entraînement après 25 étapes	53
Figure 19 : Métriques de performance de l'entraînement	53
Figure 20 : Liste des fichiers sauvegardés	54
Figure 21 : Résultat du test.....	55

Figure 22 : License for Gemma.....	56
Figure 23 : T4 GPU de Google Colab.....	56
Figure 24 : Format des données pour l'entraînement de Gemma	57
Figure 25 : Modèle Gemma	58
Figure 26 : Spécifications des modèles	58
Figure 27 : ELI5 Photosynthesis Prompt pré-entraîné	59
Figure 28 : ELI5 Photosynthesis Prompt entraîné	60
Figure 29 : Producer.....	66
Figure 30 : Consumer.....	66
Figure 31 : Consumer 2.....	67

INTRODUCTION

À l'issue de ma formation en ingénierie des données, j'ai eu l'opportunité d'effectuer un stage d'initiation de deux mois au sein de l'entreprise FORGES DE BAZAS, sous la supervision de Monsieur Lachkar Ahmed Salim. Ce stage s'inscrit dans le cadre de ma formation et m'a permis de mettre en pratique les compétences techniques acquises, notamment dans le domaine du traitement des données et de l'intelligence artificielle.

FORGES DE BAZAS est une entreprise spécialisée dans l'industrie manufacturière, où j'ai pu réaliser des missions variées liées à l'ingénierie des données. Durant ce stage, j'ai pu extraire des données provenant de multiples sources pour constituer des ensembles de données complets. J'ai également automatisé le pipeline de données à l'aide de scripts sur mesure, optimisant ainsi le flux de collecte et de traitement des données.

Une partie importante de mon travail a consisté à comparer les performances de différents grands modèles de langage (LLM) pour identifier la solution la plus adaptée. J'ai également procédé à un ajustement fin de ces modèles en utilisant des ensembles de données spécialisés extraits de livres techniques, afin d'améliorer la précision des modèles. Enfin, j'ai développé et mis en œuvre un chatbot basé sur ces LLM ajustés, permettant une application concrète de mes acquis dans un cadre réel.

En parallèle, j'ai participé à la conception et au développement d'un tableau de bord en temps réel de suivi des ventes. Ce tableau de bord permet de visualiser les données de ventes en temps réel, offrant aux utilisateurs une meilleure compréhension des performances commerciales grâce à l'agrégation des données de vente en flux continu.

Ce stage a représenté un premier contact significatif avec le milieu professionnel, me permettant d'acquérir une expérience précieuse et de mettre en pratique mes connaissances théoriques. Il a également renforcé ma capacité à m'intégrer au sein d'une équipe, à collaborer efficacement et à gérer des projets complexes tout en répondant aux problématiques techniques rencontrées.

CHAPITRE I : PRESENTATION DE L'ORGANISME D'ACCUEIL

1. Introduction :

Dans ce chapitre, nous allons situer le projet dans son contexte organisationnel, nous allons représenter l'organisme d'accueil Forges de Bazas, et l'entreprise Toyota, ensuite nous allons situer le projet dans son contexte général, puis nous allons décrire la problématique, du côté organisationnel ainsi que fonctionnel, par la suite, nous mentionnons les outils et méthodologies utilisés.

2. Contexte de l'organisme d'accueil :

2.1. Aperçu général sur l'entreprise Forges de Bazas :

Forges de Bazas est un fournisseur leader de solutions de manutention au Maroc. L'entreprise est en affaires depuis plus de 70 ans et est un partenaire de confiance pour certaines des plus grandes entreprises du pays. Forges de Bazas propose une large gamme de produits et de services, y compris des chariots élévateurs, des équipements d'entrepôt et un service après-vente. L'entreprise est déterminée à fournir à ses clients des produits et services de la plus haute qualité, et elle innove constamment pour répondre aux besoins changeants de ses clients.



Figure 1 : Logo de FORGES DE BAZAS

Le domaine d'expertise de Forges de Bazas se concentre sur les équipements de manutention de construction. Fondée en 1950, cette entreprise est reconnue comme le premier importateur de chariots élévateurs au Maroc, sous l'égide de son fondateur, Francis Fenwick, établissant ainsi une filiale marocaine de renom. Depuis plus de trois décennies, Forges de Bazas détient l'exclusivité en tant qu'importateur officiel des chariots élévateurs TOYOTA Forklift, produits par le Japon, bénéficiant d'une réputation solide de fiabilité et de

professionnalisme reconnue dans tout le secteur. En 2009, le Groupe Aixor a acquis la société, avec l'ambition de restaurer sa position dominante sur le marché, une stature qu'elle a maintenue pendant presque cinquante ans.

En 2019, Forges de Bazas a élargi son portefeuille en devenant le distributeur et représentant exclusif au Maroc de la marque SANY, spécialisée dans les engins de construction. Cette expansion a été suivie en 2020 par l'acquisition de Forges de Bazas par Upline Group, une entité fondée en 1992 et partie intégrante de la Banque Populaire du Maroc, axée sur le développement des services de banque d'investissement.

En 2023, dans un élan de solidarité suite à un séisme dévastateur, Forges de Bazas a fait preuve d'un remarquable engagement communautaire en faisant don de sept équipements essentiels pour rétablir l'accès à une route vitale. Cette voie, cruciale pour la liaison entre Amizmiz et les zones sinistrées, a facilité l'acheminement de médicaments, de nourriture et d'autres approvisionnements indispensables, témoignant ainsi de l'engagement de l'entreprise envers la responsabilité sociale et le soutien aux communautés affectées.

2.2. Fiche technique de Forges de Bazas :

Le tableau suivant (Tableau 1) présente la fiche technique du Forges de Bazas :

Adresse	Route 111 Km 11,500 - Sidi Bernoussi - Sidi Bernoussi (AR)
Téléphone	05-22-66-98-50
E-mail	general@forgesdebazas.com
Site Web	www.forgesdebazas.ma
Forme juridique	Société Anonyme
Capital	20 500 000 DHS

Figure 2 : Fiche Technique de FORGES DE BAZAS

Forges de Bazas est un fournisseur leader de solutions de manutention au Maroc. L'entreprise est en affaires depuis plus de 70 ans et est un partenaire de confiance pour certaines des plus grandes entreprises du pays. Forges de Bazas propose une large gamme de produits et de services, y compris des chariots élévateurs, des équipements d'entrepôt et un service après-vente. L'entreprise est déterminée à fournir à ses clients des produits et services de la plus haute qualité, et elle innove constamment pour répondre aux besoins changeants de ses clients.

2.3. Présentation de fournisseurs :

2.3.1. L'entreprise SANY :

SANY Heavy Industry est un leader mondial dans la fabrication d'engins de chantier. Fondée en 1989, SANY s'est imposée comme un acteur clé de l'industrie, reconnu pour l'innovation de ses produits, la haute qualité de ses équipements, et son engagement envers la satisfaction client. Cet article explorera l'histoire de SANY, ses gammes de produits diversifiées, sa présence internationale, ainsi que son engagement en faveur de la durabilité.

SANY propose une vaste gamme d'engins de chantier adaptés aux besoins variés de ses clients. Parmi ses produits figurent des excavatrices, des grues de différentes configurations, des machines à béton, des équipements routiers, des machines de battage, des engins miniers, des machines portuaires, ainsi que des éoliennes. Grâce à cette diversité, SANY se positionne comme un fournisseur unique pour les entreprises de construction de toutes tailles.

Fort d'une présence mondiale solide, SANY dispose de sites de production répartis à travers le monde, ainsi que de bureaux de vente et de service dans plus de 100 pays. L'entreprise cultive des partenariats internationaux, collaborant avec des sociétés mondiales pour distribuer et soutenir ses produits. Cette portée internationale permet à SANY de répondre aux besoins de ses clients où qu'ils soient.

SANY Global Footprint



Figure 3 : Portée Mondiale de SANY

2.3.2. L'entreprise TOYOTA Material Handling :

Toyota Material Handling est une entreprise mondiale spécialisée dans la fabrication et la distribution de chariots élévateurs et d'autres équipements de manutention pour entrepôts. Filiale de Toyota Industries, elle occupe la première place mondiale dans le domaine de la manutention depuis 2001.

Fondée en 1967 aux États-Unis, l'entreprise est devenue un leader mondial dans les solutions de manutention. Elle est présente dans plus de 5 régions du monde, dont l'Europe, le Japon, l'Amérique du Nord, la Chine et les marchés internationaux.

Toyota Material Handling propose une large gamme d'équipements de manutention, y compris des chariots élévateurs, des transpalettes, des chariots à mât rétractable, des préparateurs de commandes, des véhicules à guidage automatique et des tracteurs de remorquage. Ils fournissent également des services de gestion de flotte et d'ingénierie et de conception d'automatisation avancée.

Toyota Material Handling est engagée envers la durabilité et propose une gamme de produits électriques et écoénergétiques pour aider à réduire les émissions et la consommation d'énergie. Ils investissent également dans des installations de fabrication qui utilisent des pratiques durables.

2.3.3. Organigramme de l'entreprise :

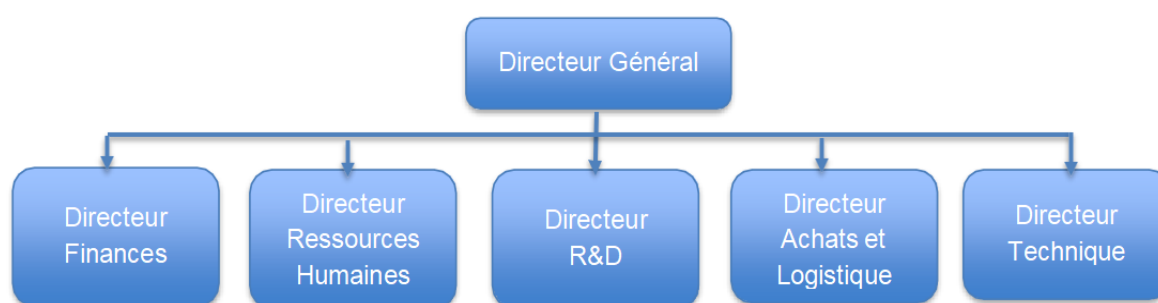


Figure 4 : Organigramme de l'entreprise

2.4. Présentation de département d'accueil :

Nous avons intégré le département de recherche et développement de SANY, dirigé par Monsieur Salim Lachkar. Ce département joue un rôle clé au sein de l'entreprise, se concentrant sur l'innovation et l'amélioration continue des produits et services.

Le département R&D collabore étroitement avec le service technique pour mettre en œuvre et tester les innovations sur le terrain. Parmi ses responsabilités, il développe de nouvelles technologies et optimise les performances des équipements pour répondre aux besoins des clients.

Le service technique, quant à lui, assure la réparation des engins, tant au sein de l'atelier que directement chez les clients, avec pour objectif de garantir un service après-vente réactif et de qualité. De plus, le service technique prépare les rapports de garantie et maintient une communication efficace avec le fournisseur SANY, garantissant une prise en charge optimale des équipements.

L'équipe technique joue également un rôle clé dans la formation continue de ses membres, assurant une amélioration constante de leurs compétences et interventions. Le département R&D contribue activement à cette formation en introduisant les nouvelles technologies et méthodes développées. De plus, le service technique s'engage dans une démarche d'amélioration continue, cherchant en permanence à optimiser ses processus et à améliorer la qualité de ses interventions. Cette approche globale permet à Sany de répondre efficacement aux besoins de ses clients, tout en maintenant un niveau élevé de performance et de fiabilité de ses équipements.

3. Conclusion :

Ce chapitre a introduit le cadre général de déroulement de ce projet de fin d'études, en présentant l'organisme d'accueil « Forges de Bazas » son domaine d'activité ses services et ses produits.

Dans le chapitre suivant, nous allons présenter avec précision notre problématique PFA.

INTRODUCTION GENERALE

1. Contexte et problématique :

Dans le domaine industriel, une préoccupation majeure réside dans la préparation efficace des machines pour garantir une production optimale. Cela revêt une importance cruciale pour maintenir la performance, la sécurité et la longévité des équipements dans des secteurs critiques. Cependant, la gestion des données techniques nécessaires à cette préparation peut s'avérer complexe, étant donné que ces informations proviennent souvent de multiples sources telles que des manuels techniques, des spécifications machines, et des protocoles de maintenance répartis sur des dizaines de documents PDF.

Les avancées récentes dans l'automatisation et l'intelligence artificielle offrent de nouvelles opportunités pour optimiser ce processus. Grâce à des outils performants, il est désormais possible d'extraire, structurer et analyser ces données de manière plus rapide et précise. En automatisant les pipelines de données, il devient envisageable d'améliorer la collecte et le traitement des informations techniques, réduisant ainsi les erreurs humaines et accélérant la prise de décision.

C'est dans ce contexte que s'inscrit notre projet de fin d'études, dont la problématique est de comparer les performances des modèles de langage (LLMs) pour identifier les solutions les plus efficaces dans l'extraction des données techniques à partir de multiples sources. Ce projet vise également à affiner ces modèles avec des jeux de données spécialisés, extraits de livres techniques, afin d'améliorer leur précision pour des applications industrielles réelles, notamment la mise en place de chatbots intelligents capables de fournir une assistance technique en temps réel.

2. Objectifs du projet :

Dans un environnement industriel en constante évolution, l'automatisation et l'analyse des données techniques jouent un rôle clé dans l'optimisation de la performance des machines. Ce projet s'inscrit dans cette dynamique en cherchant à exploiter le potentiel des modèles de langage (LLMs) pour extraire et traiter des données complexes provenant de multiples documents techniques. Nous nous fixons plusieurs objectifs ambitieux :

1. Effectuer une analyse détaillée des différentes méthodes d'extraction de données à partir de multiples sources, notamment les fichiers PDF techniques.
2. Comparer les performances de divers modèles de langage dans l'extraction et la structuration des données liées à la préparation des machines industrielles.
3. Automatiser le pipeline de données en concevant et développant des scripts personnalisés pour améliorer l'efficacité de la collecte et du traitement des informations.
4. Affiner les modèles de langage avec des jeux de données spécialisés, extraits de livres techniques et autres sources, afin d'augmenter leur précision dans des applications industrielles spécifiques.
5. Développer un chatbot basé sur ces modèles de langage, capable de fournir une assistance technique en temps réel sur la préparation des machines industrielles.
6. Intégrer les résultats obtenus dans une application conviviale permettant aux utilisateurs de consulter les informations pertinentes sur la préparation des machines de manière rapide et efficace.
7. Fournir aux techniciens et ingénieurs des informations précises et facilement accessibles, leur permettant d'optimiser la maintenance et la préparation des équipements industriels.

Ainsi, la première partie du projet sera consacrée à une étude théorique des mécanismes des modèles de langage et des méthodes d'extraction de données. Ensuite, nous comparerons leurs performances dans un contexte industriel, en les formant sur des données spécifiques aux machines et à la préparation technique. Enfin, nous développerons une application qui facilitera l'accès à ces informations critiques via un chatbot interactif, aidant à réduire les temps d'arrêt des machines et à améliorer l'efficacité globale de l'usine.

3. Méthodologie de travail :

Notre objectif principal est de mener une étude approfondie sur l'extraction et l'automatisation des données techniques à partir de multiples sources dans le contexte industriel, tout en développant et en validant des modèles de langage adaptés à cette problématique. Pour cela, nous prévoyons de mettre en œuvre la méthodologie CRISP-DM (Cross-Industry Standard Process for Data Mining), un cadre éprouvé pour guider les projets d'exploration de données à travers six étapes essentielles :

- **Compréhension des besoins métier** : Identifier les spécifications techniques nécessaires à la préparation des machines industrielles, les types de données pertinentes pour les techniciens, et les sources principales d'information (manuels, protocoles, etc.).
- **Compréhension des données** : Étudier les documents techniques disponibles, évaluer la qualité et la cohérence des données extraites des fichiers PDF, et analyser les variables clés liées à la préparation des machines (type de machine, tâches à accomplir, échéances de maintenance, etc.).
- **Préparation des données** : Nettoyer, transformer et intégrer les données issues des multiples fichiers PDF techniques, en veillant à les structurer dans une base de données cohérente et exploitable pour les modèles d'IA.
- **Modélisation** : Sélectionner et comparer les outils et modèles de langage naturel (LLMs) les plus performants pour l'extraction de données techniques. Développer, tester et interpréter les résultats des modèles pour améliorer leur précision dans des applications industrielles.
- **Évaluation et mise en œuvre** : Tester la performance du modèle final sur des données réelles issues de documents techniques et déployer les solutions dans un environnement pratique, comme un chatbot intelligent capable de fournir une assistance technique en temps réel.

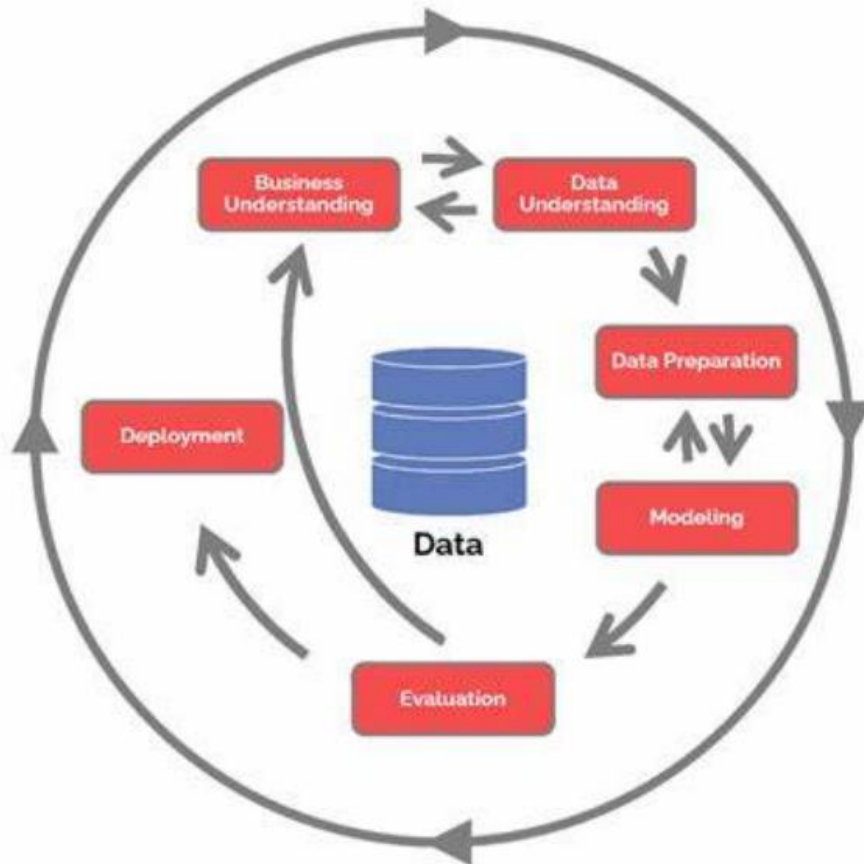


Figure 5 : Méthodologie CRISP-DM

Grâce à cette méthode itérative, nous pourrions continuer à améliorer nos modèles en nous assurant qu'ils répondent de manière optimale aux besoins spécifiques du secteur industriel, notamment dans l'automatisation des processus de préparation des machines.

Notre projet se décompose en quatre chapitres fondamentaux :

- ✓ **Chapitre 1** : Analyse comparative des outils de traitement du langage naturel et compréhension de leurs architectures et API pour l'extraction de données techniques
- ✓ **Chapitre 2** : Collecte, compréhension, préparation et structuration des données extraites des PDF techniques pour les rendre compatibles avec nos modèles d'intelligence artificielle.

- ✓ **Chapitre 3 :** Modélisation, évaluation et déploiement des modèles d'IA, en explorant différentes architectures de Deep Learning et stratégies pour améliorer la précision des chatbots industriels.

Les différents chapitres illustrent les étapes essentielles de notre étude, visant à automatiser l'extraction de données et à intégrer des solutions d'intelligence artificielle pour améliorer la gestion et la préparation des machines industrielles.

CHAPITRE II : BENCHMARKING DES OUTILS, COMPRÉHENSION DES ARCHITECTURES DE LEURS APIS

1. Introduction :

L'intelligence artificielle générative (IA générative) est un domaine en pleine expansion qui transforme la manière dont nous créons du contenu. Cette technologie permet de générer automatiquement du texte, des images, de l'audio et des vidéos, ouvrant ainsi de nouvelles perspectives dans divers domaines tels que l'art, la communication, l'éducation et la recherche.

Dans ce projet, nous nous concentrerons sur l'IA générative de texte. Aujourd'hui, il existe de nombreux outils et modèles d'IA générative de texte, chacun avec ses propres forces et faiblesses. Pour choisir les outils les plus adaptés à nos besoins, nous commencerons par définir des critères de comparaison pertinents. Ensuite, nous sélectionnerons trois modèles d'IA générative de texte et analyserons l'architecture de leurs API.

Ce chapitre présentera les résultats de notre étude. Nous comparerons les modèles sélectionnés en fonction des critères définis, en discutant de leurs avantages et inconvénients. De plus, nous fournirons des recommandations pour le choix d'un modèle d'IA générative de texte en fonction de l'application souhaitée.

2. Description et analyse des outils :

Le paysage des outils d'IA générative pour la création de contenu évolue constamment, avec l'émergence régulière de nouveaux acteurs et de nouvelles fonctionnalités. Il est essentiel de bien comprendre les divers types d'outils disponibles ainsi que leurs capacités respectives avant de faire un choix éclairé.

✓ ChatGPT :

Fondé par OpenAI en 2021, ChatGPT est un modèle de langage basé sur l'architecture GPT (Generative Pre-trained Transformer). Ce système novateur a été développé pour générer du texte de manière conversationnelle, offrant une réponse intelligente à une variété de requêtes et de tâches. Le format de dialogue permet à ChatGPT de répondre à des questions de suivi, d'admettre ses erreurs, de contester des hypothèses incorrectes et de rejeter des demandes inappropriées. Ce modèle a été formé en utilisant l'apprentissage par renforcement à partir des retours humains (RLHF), en utilisant les mêmes méthodes que InstructGPT, mais avec quelques

différences dans la configuration de la collecte de données. ChatGPT est fine-tuné à partir d'un modèle de la série GPT-3.5, qui a terminé son entraînement début 2022.

✓ **BERT :**

BERT (Bidirectional Encoder Representations from Transformers) est un GenAI introduit par Google en octobre 2018, BERT représente un jalon majeur dans le domaine du traitement du langage naturel (NLP). Fondé sur l'architecture des Transformers, ce modèle a connu une amélioration significative par rapport à ses prédécesseurs, devenant un outil incontournable dans diverses applications de NLP

✓ **XLNet :**

Fruit de la collaboration entre Google et la Carnegie Mellon University (CMU), XLNet a vu le jour en juin 2019. Ce modèle révolutionnaire adopte une approche autoregressive généralisée pour la compréhension du langage, combinant les forces des modèles AR (comme BERT) et AE (comme GPT), surpassant ainsi BERT sur de nombreuses tâches NLP.[3]

✓ **T5 :**

Lancé par Google en février 2020, T5 propose une approche innovante en transformant toutes les tâches NLP en un format texte-vers-texte. Grâce à cette simplification, le modèle peut être déployé pour la génération de texte, la traduction, la compréhension, et bien d'autres applications.[4]

✓ **BART :**

BART (Bidirectional and Auto-Regressive Transformers), conçu par Facebook AI et introduit en juillet 2020, BART se distingue en tant qu'auto encodeur de débruitage pour le pré-entraînement de modèles séquence-à-séquence. Ses performances remarquables dans la génération de texte, la traduction et le résumé en font un modèle de référence dans le domaine du NLP.

✓ **GEMINI :**

GEMINI, le modèle d'IA multimodale de Google AI, est capable de comprendre et de générer divers types de données comme le texte, les images, l'audio, la vidéo et le code. Cette

technologie avancée offre plusieurs fonctionnalités innovantes : dialogue multimodal, multilinguisme, création de jeux, résolution d'énigmes visuelles, génération d'images et de texte, raisonnement logique et spatial, traduction visuelle et compréhension culturelle. Annoncé en décembre 2023, GEMINI est disponible en version bêta fermée depuis janvier 2024.

✓ **Bing AI :**

Bing AI, modèle d'IA génératif de Microsoft, peut générer du texte, traduire des langues, rédiger divers contenus créatifs et répondre à vos questions de manière informative. Cet outil polyvalent peut être utilisé pour rédiger du contenu, traduire des langues, répondre à des questions et créer divers types de contenus créatifs, tels que des articles de blog, des poèmes, des scripts, etc. Annoncé en mars 2024, Bing AI est disponible en version bêta fermée depuis avril 2024.

✓ **LLAMA :**

LLAMA, modèle d'IA génératif de Meta AI, est un outil puissant capable de générer du texte, de traduire des langues, de répondre à des questions et de créer divers types de contenu créatif. Annoncé en mai 2024, LLAMA est disponible en version bêta fermée depuis juin 2024. Utilisable pour la rédaction de contenu, la traduction, la création de contenu créatif et la réponse à des questions, LLAMA offre une gamme de fonctionnalités pour diverses applications.

✓ **Asper AI :**

Asper AI est une plate-forme développée par Jasper AI d'intelligence artificielle qui permet aux utilisateurs de produire instantanément des copies de type humain pour les articles de blog, les publicités sur les réseaux sociaux, les e-mails, les pages de destination, etc. La plateforme utilise la technologie GPT-3 pour créer cette copie, qui est surtout connue pour être la pierre angulaire de ChatGPT.

Jasper AI a été lancé en janvier 2021, et l'équipe derrière la plateforme compte désormais plus de 80 membres. L'équipe vise à améliorer Jasper au point où il peut aider à rationaliser le processus d'écriture impliqué dans divers rôles – ou à remplacer entièrement ces rôles.

Le modèle puissant de Jasper peut créer du contenu écrit dans 26 langues, dont l'anglais, le portugais et le japonais. Compte tenu de ses capacités et de son accessibilité, Jasper a remporté

de nombreux prix de l'industrie et est noté 4,9/5 dans plus de 3 000 avis sur le Web. Il est gratuit (niveau limité)

✓ **Cohere AI :**

Cohere est une société d'IA spécialisée dans les grands modèles de langage (LLM), qui propose une plateforme mise à disposition via une API permettant aux développeurs de :

- Tirez parti des LLM prédéfinis de Cohere pour effectuer des tâches courantes de saisie de texte, telles que : résumer, classer et trouver les similitudes dans le contenu, c'est-à-dire traitement du langage naturel (NLP). - Créer leurs propres modèles de langage, basés sur le travail déjà effectué par Cohere, qui peuvent être personnalisés en fonction de leurs propres données de formation.
- OpenAI a récemment attiré l'attention pour ChatGPT et son incursion dans les solutions d'entreprise, mais Cohere AI propose depuis un certain temps déjà des modèles de langage étendus (LLM) accessibles et facilement déployables aux entreprises. Et bien que Cohere soit en concurrence avec des géants de l'industrie tels que Google et OpenAI dans le domaine des LLM, la société est relativement obscure par rapport à ses homologues.
- La startup basée à Toronto souhaite rendre sa technologie en constante amélioration accessible à tous les développeurs d'une manière qui ne soit pas prohibitive et qui ne soit pas difficile car la barrière à l'entrée est faible. Cohere a un accès anticipé payant.

✓ **Copy AI :**

Copy AI est un rédacteur IA avancé qui peut, entre autres, vous aider à trouver de nouvelles idées, à rédiger du contenu et à créer des publicités sur les réseaux sociaux. Elle utilise des algorithmes avancés, souvent basés sur des modèles linguistiques comme GPT-4, pour générer du texte de manière autonome. Copy ai a été fondé en 2020 par deux personnes : Paul Yacoubian et Chris Lu.

Cet outil vise à assister les utilisateurs dans la création de contenu écrit, que ce soit pour des besoins marketing, pour des blogs, des emails ou même pour des postes sur les réseaux sociaux. Copy AI promet d'optimiser le processus créatif en fournissant des suggestions de contenu, des formulations et des idées innovantes. Gratuit (niveau limité), payant.

✓ **Anthropic's Claude :**

Claude est un grand modèle de langage (LLM) construit par Anthropic. Il est entraîné pour être un assistant utile, honnête et inoffensif avec un ton conversationnel.

L'outil est capable de générer du texte, en réponse à une requête posée par l'utilisateur. Claude semble se distinguer de ses concurrents tels que ChatGPT ou Bing Chat, car il a été conçu à partir d'un « entraînement constitutionnel », dans le but de devenir « utile, honnête et inoffensif ». Ainsi, le chatbot rejette automatiquement les requêtes malintentionnées ou préjudiciables.

Claude dispose d'une version gratuite, accessible au grand public via un navigateur web, et propose également une API. Il existe également une version payante, nommée Claude Pro, et accessible pour 20\$ par mois, qui permet de : générer davantage de requêtes (5 fois plus que l'offre gratuite), bénéficier d'un accès prioritaire en cas de forte affluence, ainsi que d'un accès anticipé aux nouvelles fonctionnalités.

Claude est également en phase bêta fermée avec un accès restreint.

✓ **LaMDA de Google :**

LaMDA a été conçu pour permettre aux produits Google d'engager des conversations en langage naturel avec les utilisateurs finaux. Ce modèle est souvent associé à Google Bard, considéré comme un concurrent direct de ChatGPT d'OpenAI.

LaMDA a également fait l'objet d'une attention particulière suite à un article du Washington Post, qui a suscité des débats autour de la question de la sensibilité des logiciels d'IA. L'histoire, devenue virale, a alimenté les spéculations sur la pertinence du test de Turing et inspiré des discussions sur la nécessité d'un cadre réglementé pour une IA responsable.

Développé par Google AI, LaMDA a été présenté pour la première fois lors de la conférence Google I/O en mai 2021. Bien que toujours en développement, LaMDA n'est pas encore accessible au grand public

3. Choix des critères de comparaison :

L'intelligence artificielle générative (IA générative) ouvre de nouvelles perspectives pour la création de contenu dans le domaine de la préparation des machines. Avec une multitude d'outils disponibles, il est crucial de sélectionner celui qui correspond parfaitement à nos besoins et objectifs. Voici les critères clés à considérer :

✓ **Licence et Open Source :**

Certains outils d'IA générative sont accessibles en tant que logiciels open source, ce qui signifie que leur code est librement disponible et modifiable par la communauté. D'autres sont des solutions propriétaires et exigent l'acquisition d'une licence payante pour leur utilisation. Pour ceux qui cherchent une solution flexible et adaptable, les outils open source offrent généralement davantage de possibilités de personnalisation.

✓ **Facilité d'utilisation :**

L'évaluation de la convivialité d'un outil comprend plusieurs aspects tels que la simplicité de l'installation, la facilité de configuration et l'ergonomie de l'interface utilisateur. Certains outils offrent des interfaces conviviales adaptées aux utilisateurs novices, tandis que d'autres exigent des compétences techniques plus avancées.

✓ **Performance et Précision :**

L'évaluation des performances des outils d'IA générative se concentre sur la qualité des modèles générés. Cela inclut la précision dans la génération de texte, d'images ou d'autres types de contenu. Les métriques telles que la perplexité pour le texte ou la qualité visuelle pour les images sont prises en compte pour évaluer la qualité des résultats.

✓ **Adaptabilité :**

Certains outils sont spécialisés dans un domaine spécifique, comme la génération de texte ou d'images, tandis que d'autres sont conçus pour être plus polyvalents. Le choix dépend des besoins spécifiques de l'utilisateur et du domaine dans lequel il souhaite utiliser l'outil.

✓ **Support et Documentation :**

La disponibilité d'une documentation complète et de ressources de support est essentielle pour faciliter l'utilisation et la résolution des problèmes éventuels. Les forums de support et les communautés en ligne peuvent également être précieux pour obtenir de l'aide et des conseils.

✓ **Évolutivité :**

Pour les utilisateurs envisageant une utilisation à grande échelle, il est crucial de s'assurer que l'outil est capable de gérer des charges de travail importantes et qu'il peut être facilement adapté à l'évolution des besoins.

✓ **Intégration :**

Certains outils sont conçus pour s'intégrer facilement à d'autres systèmes via des API ou des interfaces standard, tandis que d'autres nécessitent des adaptations plus complexes pour fonctionner avec les applications existantes de l'utilisateur.

✓ **Sécurité et Confidentialité :**

Pour les utilisateurs traitant des données sensibles, il est essentiel de s'assurer que l'outil respecte les normes de sécurité et de confidentialité. Certains outils offrent la possibilité d'être utilisés localement pour garantir la confidentialité des données.

✓ **Communauté et Mises à Jour :**

La présence d'une communauté active de développeurs et de mises à jour fréquentes sont des indicateurs de la vitalité et de la fiabilité de l'outil sur le long terme.

✓ **Coût :**

Les coûts associés à chaque outil, tels que les licences, les frais d'utilisation ou de maintenance, doivent être pris en compte en fonction du budget et des avantages offerts par chaque solution.

✓ **Temps d'exécution :**

Considérer la vitesse à laquelle chaque outil génère du contenu et le temps nécessaire pour obtenir des résultats. Évaluer l'efficacité et la rapidité de l'outil dans le processus de génération.

4. Tableau comparatif des outils génératifs :

Nous avons mené une analyse approfondie de plusieurs outils d'IA générative populaires, en nous basant sur des critères clés tels que l'Open Source, la Facilité d'utilisation, la Précision et le Temps d'exécution.

Le tableau ci-dessous présente nos résultats, offrant une comparaison concise des fonctionnalités et des performances de chaque outil.

Outil	Open Source	Facilité	Précision	Temps d'exécution
ChatGPT	Non	Moyenne	Bonne	Rapide
BERT	Oui	Difficile	Excellente	Lent
XLNet	Oui	Difficile	Excellente	Lent
T5	Oui	Difficile	Excellente	Moyen
BART	Oui	Difficile	Excellente	Lent
GEMINI	Non	Bonne	Excellente	Rapide
Bing AI	Non	Moyenne	Bonne	Rapide
LLAMA	Non	N/A	N/A	N/A
Asper AI	Non	N/A	N/A	N/A
Cohere AI	Non	N/A	N/A	N/A
Copy ai	Non	Facile	Bonne	Rapide
Anthropic's Claude	Non	N/A	N/A	N/A
LaMDA de Google	Non	N/A	N/A	N/A

Explication des classements :

Facilité d'utilisation :

- Facile.
- Moyen.
- Difficile.

Précision :

- Excellente.
- Bonne.
- Moyenne.

Temps d'exécution :

- Rapide.
- Moyen.
- Lent.

5. Etude de l'architecture des APIs :

5.1. Définition de API :

Une API, ou interface de programmation d'applications, est un ensemble de règles ou de protocoles qui permettent aux applications logicielles de communiquer entre elles pour échanger des données, des caractéristiques et des fonctionnalités. En utilisant des API, les développeurs peuvent rendre le développement d'applications plus facile en intégrant des données, des services et des fonctionnalités provenant d'autres applications, plutôt que de les créer à partir de zéro.

Les API sont aussi un moyen facile et sécurisé pour les propriétaires d'applications de mettre à la disposition des services internes de leur organisation les données et les fonctionnalités de leurs applications. Ces données et fonctionnalités peuvent également être partagées ou commercialisées par les propriétaires d'applications avec des partenaires commerciaux ou des tiers.

5.2. Les modèles de langage IA pour le texte avec API publique :

Voici une liste de modèles de langage IA pour le texte avec API publique :

Modèles open-source :

- ✓ Bard :(basé sur Transformer).
- ✓ T5 :(basé sur Transformer).
- ✓ GPT-J :(basé sur Transformer).
- ✓ Bloom :(basé sur Transformer).

Modèles payants :

- ✓ Cohere : (large éventail de fonctionnalités).
- ✓ Jasper AI (bon pour la rédaction de contenu).
- ✓ Copy.ai :(bon pour le marketing et la publicité).

Modèles en cours de développement :

- ✓ LaMDA :(développé par Google AI).
- ✓ Claude : (développé par Anthropic AI).

5.3. Architecture de Chat GPT :

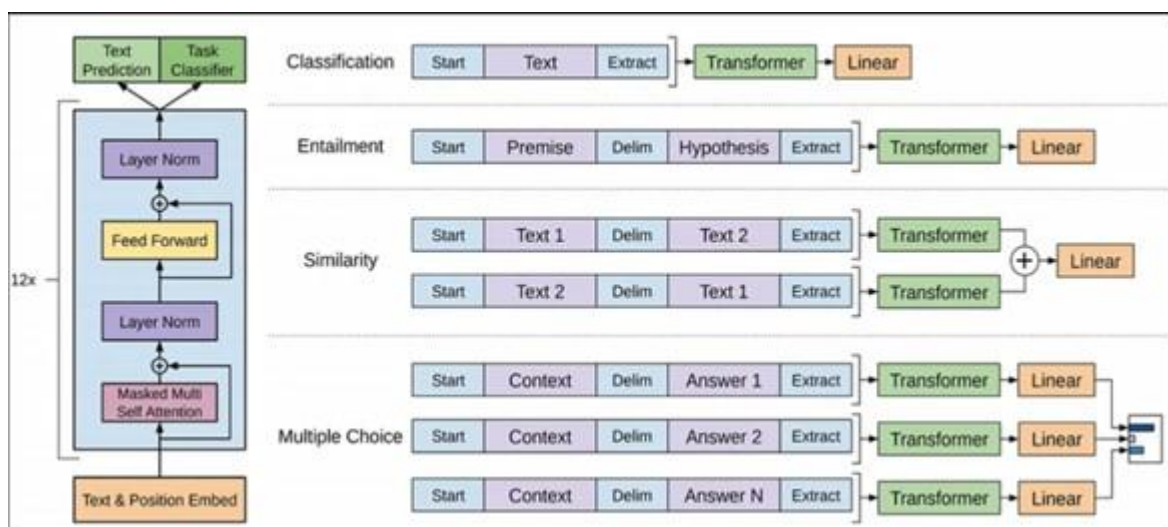


Figure 6 : Architecture de Chat GPT

ChatGPT est une variante du modèle GPT (Generative Pre-training Transformer), qui est un type d'architecture de réseau neuronal basé sur des transformateurs. Le modèle est entraîné sur un grand ensemble de données textuelles et est capable de générer un texte semblable à celui d'un humain en réponse à une requête donnée.

L'architecture de ChatGPT comprend un encodeur et un décodeur, similaire à l'architecture traditionnelle des transformateurs. L'encodeur est composé de plusieurs couches d'auto-attention et de réseaux neuronaux à propagation avant, qui sont utilisés pour traiter et comprendre le texte d'entrée. Le décodeur est également composé de plusieurs couches d'auto-attention et de réseaux neuronaux à propagation avant, qui sont utilisés pour générer le texte de sortie.

Le modèle comprend également une tête de modèle de langage, qui est une couche linéaire avec des poids qui sont appris lors de la pré-entraînement. Cela est utilisé pour prédire le jeton suivant dans la séquence, étant donné les jetons précédents.

De plus, ChatGPT comprend également une tête de génération de dialogue, qui est une couche linéaire avec des poids qui sont appris lors du réglage fin du modèle sur des données conversationnelles. La tête de génération de dialogue est utilisée pour générer la réponse à une requête donnée dans le contexte d'un dialogue.

Dans l'ensemble, l'architecture de ChatGPT est conçue pour générer un texte semblable à celui d'un humain, et elle utilise une architecture de réseau neuronal basée sur des transformateurs qui comprend un encodeur, un décodeur, une tête de modèle de langage et une tête de génération de dialogue.

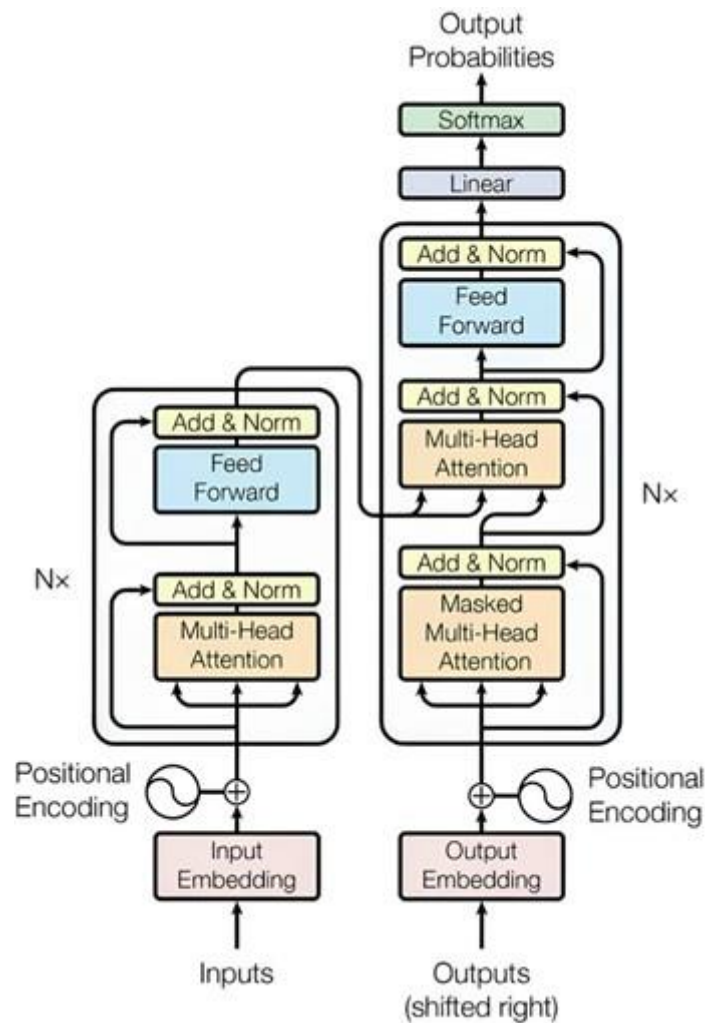


Figure 7 : Architecture du modèle Transformer

5.4. Les éléments constitutifs de ChatGPT :

1. **The Transformer architecture** : L'architecture Transformer est le fondement de ChatGPT. C'est une architecture de réseau neuronal qui utilise des mécanismes d'auto-attention pour traiter des séquences d'entrée. L'architecture Transformer est capable de traiter des séquences d'entrée de longueurs variables et permet un traitement parallèle de l'entrée.
2. **The Encoder** : L'Encodeur est composé de plusieurs couches de self-attention et de réseaux neuronaux à propagation avant. Il traite et comprend le texte d'entrée.

3. **The Decoder** : Le décodeur est également composé de plusieurs couches de self-attention et de réseaux neuronaux à propagation avant. Il génère le texte de sortie.
4. **The Language Model Head** : La tête du modèle de langage est une couche linéaire dont les poids sont appris lors de la préformation. Elle est utilisée pour prédire le prochain jeton dans la séquence, étant donné les jetons précédents.
5. **The Dialogue Generation Head** : La tête de génération de dialogue est une couche linéaire dont les poids sont appris lors du fine-tuning du modèle sur des données conversationnelles. Elle est utilisée pour générer la réponse à une invitation donnée dans le contexte d'un dialogue.
6. **Pre-training** : ChatGPT est pré-entraîné sur un vaste ensemble de données textuelles, ce qui lui permet de générer un texte semblable à celui d'un humain en réponse à une invitation donnée.
7. **Fine-Tuning** : Le modèle est ajusté sur des données conversationnelles afin d'améliorer sa capacité à générer des réponses dans le contexte d'un dialogue.

Ces éléments fonctionnent ensemble pour fournir au modèle la capacité de comprendre le texte d'entrée et de générer un texte semblable à celui d'un humain en réponse à une invitation donnée.

6. Conclusion :

Dans cette analyse approfondie, nous avons exploré le paysage des outils d'IA générative populaires, en nous concentrant sur des critères clés tels que l'Open Source, la Facilité d'utilisation, la Précision et le Temps d'exécution. Le tableau comparatif et les classements présentés offrent une perspective concise sur les fonctionnalités et les performances de chaque outil.

L'étude approfondie de l'architecture de l'API ChatGPT nous a permis de comprendre les principes fondamentaux de son fonctionnement et d'établir une base solide pour l'analyse des autres outils. Cette approche méthodique sera appliquée aux autres API afin de fournir une évaluation complète et objective.

En ce qui concerne la sélection des outils les plus appropriés pour notre projet d'étude, nous choisirons de nous concentrer sur les trois outils d'IA générative parmi ceux analysés :

1. GPT-2 se distingue par sa facilité d'utilisation, le rendant accessible même aux utilisateurs novices. De plus, il offre une bonne qualité de génération de texte et est particulièrement rapide dans l'exécution de ses tâches.
2. LLAMA 2 est reconnu pour sa précision élevée dans la génération de texte, ce qui en fait l'un des outils les plus fiables dans ce domaine. Son caractère open source permet aux utilisateurs d'accéder au code source et de le personnaliser selon leurs besoins spécifiques.

Gemma, développé par Google, se démarque par sa convivialité, offrant une expérience d'utilisation relativement simple grâce à des interfaces intuitives. De plus, il est efficace dans l'exécution de ses tâches, ce qui le rend adapté aux projets nécessitant des réponses rapides.

CHAPITRE III : COLLECTE, COMPRÉHENSION, PRÉPARATION ET STRUCTURATION DES DONNÉES

1. Introduction :

Ce chapitre détaille les méthodes utilisées pour extraire des données à partir de plusieurs fichiers PDF afin de construire un dataset destiné à l'analyse des informations techniques relatives aux machines industrielles. Les PDF contiennent des données cruciales sur la préparation des machines et d'autres informations techniques. L'objectif principal est de transformer ces données en un format structuré utilisable pour l'entraînement de modèles d'intelligence artificielle.

2. Importance des Données dans l'Entraînement des Modèles :

Les modèles de prédiction utilisés pour l'analyse des informations techniques relatives aux machines industrielles jouent un rôle essentiel dans l'optimisation des processus de maintenance, la gestion de la production, et l'amélioration de la performance des équipements. Cependant, ces modèles sont souvent confrontés à des limitations dues à des ensembles de données incomplets ou biaisés, ce qui peut affecter leur capacité à fournir des prévisions précises et exploitables. Pour remédier à ces lacunes, il est crucial d'intégrer des données issues de diverses sources, telles que les manuels techniques, les rapports de maintenance, et les relevés d'utilisation des machines. Ces sources de données permettent d'offrir une vue d'ensemble des performances et des défaillances des machines, en couvrant une large gamme de paramètres techniques, d'environnements opérationnels, et de conditions de maintenance. L'incorporation de ces données diversifiées permet aux modèles de mieux comprendre les subtilités des interactions entre les différentes composantes des machines, conduisant ainsi à des recommandations plus précises et adaptées pour l'amélioration des processus industriels.

3. Sources de Données :

Pour notre stage, les données nécessaires, qui concernent des informations techniques relatives aux machines industrielles, sont essentiellement focalisées sur les documents au format PDF fournis par l'entreprise.

4. Méthodologies de Collecte :

Nous avons principalement utilisé l'API de Groq pour extraire des informations pertinentes des documents. Voici comment nous avons procédé :

Nous avons exploité le modèle LLAMA-3.1-70B Versatile via l'API de Groq afin de générer des questions et des réponses à partir de multiples fichiers PDF. Cette approche nous a permis de démontrer comment extraire des informations précieuses des PDF en utilisant l'API de Groq.

Grâce à ce modèle avancé, nous avons pu exploiter les connaissances cachées dans ces documents techniques et ainsi enrichir notre dataset avec des informations essentielles.

4.1. Définition de GROQ :

Groq est une entreprise technologique spécialisée dans le développement de matériel et de logiciels pour l'accélération des calculs d'intelligence artificielle (IA) et d'apprentissage automatique (Machine Learning). Leur principale innovation repose sur une architecture de processeur spécialement conçue pour des calculs massivement parallèles, permettant d'exécuter des modèles d'IA complexes de manière ultra-rapide et efficace. Les solutions de Groq sont utilisées pour des tâches intensives en calcul, comme le traitement de réseaux neuronaux profonds, offrant des performances supérieures en termes de latence et de consommation énergétique par rapport aux processeurs traditionnels.



4.2. Pourquoi choisir le modèle LLAMA-3.1-70B-Versatile ?

Il est généralement recommandé de générer des questions-réponses à partir d'un PDF en envoyant des parties du PDF dans des requêtes séparées via l'API de Groq. Cependant, cette approche peut entraîner une perte de la connexion contextuelle entre les parties, ce qui peut réduire la précision des questions-réponses. Pour éviter cela, il est essentiel de choisir un modèle en fonction de sa capacité à traiter les jetons d'entrée, de sortie et de résumé.

Modèle	Requêtes par minute	Requêtes par jour	Jetons par jour	Jetons par minute
llama-7b-it	5,000	14,400	131,072	1,000,000
llama-7b-it	5,000	14,400	131,072	1,000,000
llama-2-bb-it	30	14,400	131,072	1,000,000
llama-3.1-40b-reasoning	30	14,400	131,072	1,000,000
llama-3.1-70b-versatile	100	14,400	131,072	1,000,000
llama-3.1-8b-instant	30	14,400	131,072	1,000,000
llama-guard-3-8b	30	14,400	131,072	1,000,000
llama-3-70b-8192	30	14,400	131,072	1,000,000
llama-3-8b-8782	30	14,400	131,072	1,000,000
llama-3-groq-70b-8192-tool-use-preview	30	14,400	131,072	1,000,000
llama-3-groq-8b-8192-tool-use-preview	30	14,400	131,072	1,000,000

Figure 8 : Tableau comparatif des modèles

Comme vous pouvez le voir, les modèles LLAMA sont les meilleurs pour répondre à nos besoins. Les différences entre les modèles LLAMA résident principalement dans leurs configurations mathématiques. Nous allons donc choisir le modèle LLAMA-3.1-70B-Versatile.

4.3. Problèmes et obstacles trouvés :

La technique de conservation du contexte est essentielle pour l'extraction d'informations et la génération de questions à partir de documents volumineux, tels que les fichiers PDF. En envoyant le texte extrait de chaque page accompagné d'un résumé du texte précédent, nous permettons au modèle de maintenir une compréhension cohérente du contenu global. Pour ce faire, nous utilisons une fonction de résumé pour condenser les textes en morceaux plus digestes. Chaque morceau est traité par un modèle de résumé, comme sshleifer/distilbart-cnn-12-6, afin de produire des résumés concis qui capturent l'essentiel des informations. Ces

résumés sont ensuite combinés pour créer un résumé global qui est intégré avec le texte de la page actuelle, améliorant ainsi la qualité des questions et des réponses générées par le modèle. Cette approche assure que les informations pertinentes de chaque page sont prises en compte dans le contexte global du document, permettant une analyse plus précise et complète.

4.4. Structuration des Données :

Les données préparées ont été structurées en fonction des exigences des modèles d'IA Générative que nous allons utiliser, notamment les modèles Gemma et LLAMA2. Les données collectées.

Dans les documents ont été particulièrement importantes pour le fine-tuning de ces modèles. Et pour faciliter le fine-tuning, nous avons structuré les données sous forme de questions-réponses.

Par exemple :

- **Question :** Can a wheel loader be adapted with optional equipment?
- **Réponse :** Yes, with forks, scrapers, or different bucket types.

Cette structuration permet une meilleure utilisation des données dans les modèles d'IA, en fournissant des paires claires et concises pour l'apprentissage.

5. Conclusion :

Ce chapitre a couvert les étapes essentielles de la gestion des données, depuis leur collecte jusqu'à leur préparation pour l'analyse. En suivant une méthodologie rigoureuse, nous avons assuré que les données étaient de haute qualité, cohérentes et prêtes pour la modélisation. Ces étapes sont fondamentales pour garantir le succès des analyses et des modèles d'IA générative que nous développerons dans les chapitres suivants

CHAPITRE IV : MODÉLISATION, ÉVALUATION ET DÉPLOIEMENT DES MODÈLES

1. Introduction :

Dans ce chapitre, nous abordons le cœur de notre projet : la modélisation, l'évaluation et le déploiement des modèles d'intelligence artificielle générative. Nous explorons différentes architectures de Deep Learning et intégrons des API de modèles existants tels que ChatGPT, Gemini et Bing pour optimiser nos prédictions. Ce chapitre décrit en détail les étapes de sélection, de développement, de test, d'interprétation des résultats, ainsi que les stratégies de déploiement des modèles dans notre application web.

2. Modélisation :

2.1. Utilisation de Hugging Face et Kaggle :

Pour obtenir et fine-tuner nos modèles, nous avons utilisé les plateformes Hugging Face et Kaggle, qui offre une large gamme d'outils et de ressources pour travailler avec des modèles de langage avancés.



Figure 9 : Logo de Hugging Face



Figure 10 : Logo de Kaggle

2.2. Choix des modèles :

Pour notre projet, nous avons sélectionné deux modèles d'IA générative basés sur des critères de performance et de pertinence pour notre domaine d'application. Les outils utilisés incluent :

GPT 2 : GPT-2, ou Generative Pre-trained Transformer 2, est un modèle de langage avancé développé par OpenAI. Il se base sur l'architecture de son prédécesseur, GPT, en utilisant le modèle de transformateur pour générer du texte de manière humaine.



Figure 11 : Logo de OpenAI GPT-2

LLAMA 2 : LLAMA 2 est un modèle de langage avancé développé par Meta AI (anciennement connu sous le nom de Facebook AI).



Figure 12 : Logo de LLAMA2

Gemma : Modèle d'IA générative avancé de Google, conçu pour fournir des prédictions précises et des réponses contextuelles.



Figure 13 : Logo de Gemma

2.3. Fine-Tuning du Modèle GPT-2 :

2.3.1. Sélection du Modèle GPT-2 Small :

Le modèle GPT-2 (Generative Pre-trained Transformer 2) est pré-entraîné sur une vaste quantité de texte, ce qui lui permet de capturer une compréhension profonde du langage.

Cependant, pour notre application spécifique, il est nécessaire de fine-tuner (affiner) le modèle sur nos propres données.

GPT-2 est disponible en plusieurs tailles, en fonction du nombre de paramètres :

- ❖ GPT-2 Small ('gpt2') : 124 millions de paramètres.
- ❖ GPT-2 Medium ('gpt2-medium') : 345 millions de paramètres.
- ❖ GPT-2 Large ('gpt2-large') : 774 millions de paramètres.
- ❖ GPT-2 XL ('gpt2-xl') : 1,5 milliard de paramètres.

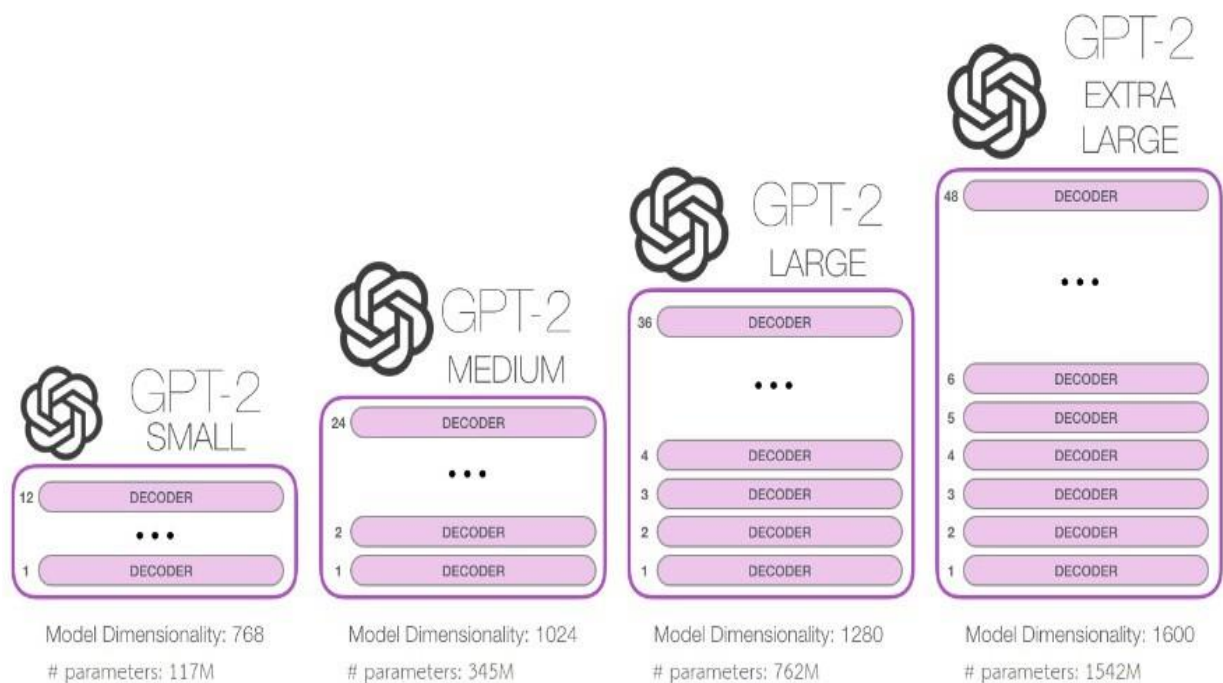


Figure 14 : Les différents tailles de GPT-2

2.3.2. Evaluation du modèle avant le fine-tuning :

Le modèle atteint les résultats suivants sans aucun ajustement préalable (zero-shot) :

Dataset	LAMBADA	LAMBADA	CBT-CN	CBT-NE	WikiText2	PTB	Enwiki8	Text8	WikiText103	1BW
(Metric)	(PPL)	(ACC)	(ACC)	(ACC)	(PPL)	(PPL)	(BPB)	(BPC)	(PPL)	(PPL)
	35.13	45.99	87.65	83.4	29.41	65.85	1.16	1.17	37.50	75.20

Figure 15 : Métriques de performance pour divers datasets

2.3.3. Fine Tuning du Modèle GPT-2 Small :

Le fine-tuning de GPT-2 implique de réentraîner le modèle pré-entraîné sur notre jeu de données. Cela permet d'adapter le modèle au contexte, tout en bénéficiant des connaissances générales acquises durant l'entraînement initial :

a. Préparation des Données :

Les données d'entraînement sont collectées et nettoyées. Dans le code, les fichiers texte sont lus et les lignes non vides sont extraites pour former un ensemble de données unique.

b. Chargement du Modèle et du Tokenizer :

Dans cette section, nous utilisons principalement la bibliothèque transformers pour charger le tokenizer et le modèle GPT-2 pré-entraîné. La bibliothèque transformers fournit une interface conviviale pour travailler avec une variété de modèles de traitement du langage naturel, y compris GPT-2. Le chargement du tokenizer et du modèle est réalisé à l'aide des classes GPT2Tokenizer et GPT2LMHeadModel de la bibliothèque transformers. Ces classes fournissent des fonctionnalités avancées pour la tokenisation du texte et la gestion du modèle GPT-2

c. Configuration des Arguments d'Entraînement :

Configuration des Arguments d'Entraînement Les paramètres d'entraînement sont définis, incluant la taille des lots, le nombre d'époques, et le taux d'apprentissage. Ces paramètres influencent la vitesse et la qualité de l'entraînement.

d. Entraînement du Modèle :

Le modèle est entraîné sur les données préparées. Durant cette phase, les poids du modèle sont ajustés pour minimiser l'erreur entre les prédictions du modèle et les vérités terrain.

e. Tokenisation du Prompt :

Le texte de départ est tokenisé et converti en une séquence de jetons. Les jetons sont passés à travers le modèle pour générer une séquence de texte. Les paramètres tels que la longueur maximale de la séquence, le nombre de faisceaux (beams), et la température (qui contrôle la diversité des sorties) peuvent être ajustés pour obtenir des résultats différents.

f. Décodage :

La séquence de jetons générée est convertie en texte lisible.

2.3.4. Fine-Tuning du Modèle LLAMA 2 :

a. Sélection du Modèle LLAMA 2 7b-chat-hf :

Dans ce projet, nous explorerons LLAMA-2 et montrerons comment le régler le fine-tuning sur un nouvel ensemble de données à l'aide de Google Colab. De plus, nous aborderons de nouvelles méthodologies et techniques de réglage fin qui peuvent réduire l'utilisation de la mémoire et accélérer le processus d'entraînement.

Nous avons choisi le modèle LLAMA 2 7b-chat-hf disponible sur Hugging Face pour plusieurs raisons :

- 1. Taille du modèle :** Le modèle LLAMA 2 7b-chat-hf est suffisamment grand pour capturer des informations complexes dans les données, mais il n'est pas trop volumineux pour entraîner efficacement sur des ressources limitées.
- 2. Performance :** LLAMA-2 a montré de bonnes performances dans diverses tâches de traitement du langage naturel, ce qui en fait un choix solide pour notre projet.
- 3. Disponibilité :** Le modèle est facilement accessible via Hugging Face, ce qui nous permet de l'intégrer rapidement dans notre flux de travail.

b. Comprendre le fine-tuning du modèle LLAMA 2 :

LLAMA 2 est une collection de LLM (Large Language Models) open source de deuxième génération de Meta, livrée avec une licence commerciale. Elle est conçue pour gérer un large éventail de tâches de traitement du langage naturel, avec des modèles allant de 7 milliards à 70 milliards de paramètres.

LLAMA-2-Chat, optimisé pour le dialogue, a montré des performances similaires à celles de modèles fermés populaires tels que ChatGPT et PaLM. Nous pouvons même améliorer les performances du modèle en le réglant finement sur un ensemble de données conversationnelles de haute qualité.

Le réglage fin (fine-tuning) en apprentissage automatique consiste à ajuster les poids et les paramètres d'un modèle pré-entraîné sur de nouvelles données afin d'améliorer ses performances sur une tâche spécifique. Cela implique d'entraîner le modèle sur un nouvel ensemble de données spécifique à la tâche tout en mettant à jour les poids du modèle pour s'adapter aux nouvelles données.

Il est impossible de régler finement les LLM sur du matériel grand public en raison de la mémoire vidéo insuffisante et des capacités de calcul limitées. Cependant, dans ce projet, nous surmonterons ces défis de mémoire et de calcul et entraînerons notre modèle à l'aide d'une version gratuite de Google Colab.

c. Fine-Tuning de LLAMA 2 :

Pour fine-tuner le modèle LLAMA 2 avec 7 milliards de paramètres sur un GPU T4. Nous avons utilisé un GPU gratuit sur Google Colab.

Le GPU Colab T4 dispose de seulement 16 Go de VRAM, ce qui est à peine suffisant pour stocker les poids de LLAMA 2-7b. Cela signifie que le fine-tuning complet n'est pas possible, et nous devons utiliser des techniques en termes de paramètres, comme LoRA ou QLoRA.

Nous utiliserons la technique QLoRA pour fine-tuner le modèle avec une précision de 4 bits et optimiser l'utilisation de la VRAM. Pour cela, nous utiliserons l'écosystème Hugging Face des bibliothèques LLM : transformers, accelerate, peft, trl et bitsandbytes.

d. Configuration du modèle :

Au lieu d'attendre quelques jours pour obtenir une confirmation pour accéder au modèle LLAMA-2 officiel de Meta sur Hugging Face, nous utiliserons le modèle LLAMA-2-7b-chat-hf de NousResearch comme modèle de base. Il est identique à l'original, mais facilement accessible.



e. Chargement du jeu de données, du modèle et du tokenize :

Nous allons charger le jeu de données depuis le hub Hugging Face. Ce jeu de données contient des échantillons et a été traité pour correspondre au format de prompt de LLAMA 2.

f. Configuration de la quantification en 4 bits :

La quantification en 4 bits via QLoRA permet de fine-tuner efficacement de grands modèles de langage (LLM) sur du matériel grand public tout en conservant une haute performance. Cela améliore considérablement l'accessibilité et la convivialité pour les applications réelles.

QLoRA quantifie un modèle de langage pré-entraîné à 4 bits et fige les paramètres. Un petit nombre de couches adaptatives de faible rang, entraînaibles, sont ensuite ajoutées au modèle.

Pendant le fine-tuning, les gradients sont rétro propagés à travers le modèle quantifié en 4 bits figé, uniquement dans les couches adaptatives de faible rang. Ainsi, l'ensemble du modèle pré-entraîné reste fixé à 4 bits tandis que seuls les adaptateurs sont mis à jour. De plus, la quantification en 4 bits ne nuit pas aux performances du modèle.

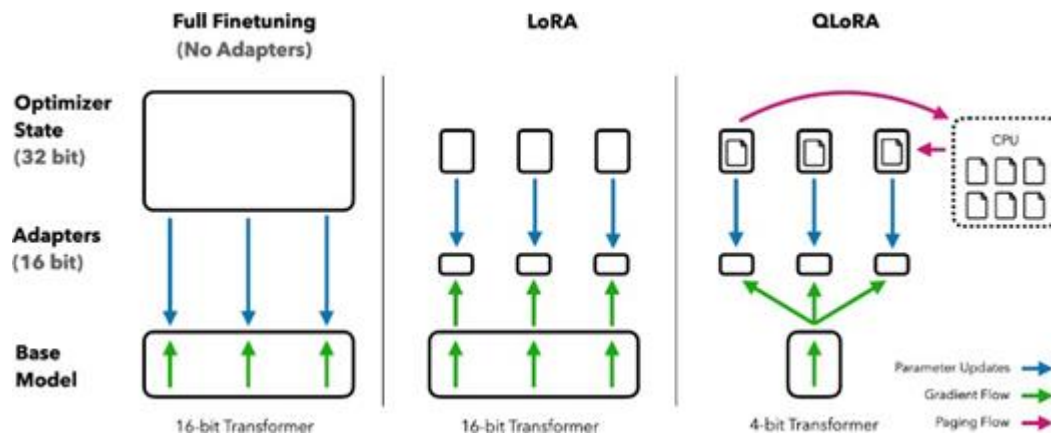


Figure 16 : Comparaison des méthodes de fine-tuning : Full Finetuning, LoRA et QLoRA

g. Chargement du modèle LLAMA 2 :

Nous allons maintenant charger le modèle en utilisant une précision de 4 bits avec le type de calcul "float16" depuis Hugging Face pour un entraînement plus rapide.

h. Chargement du tokenizer :

Ensuite, nous allons charger le tokenizer depuis Hugging Face et définir padding_side sur "right" pour résoudre le problème avec fp16.

i. Paramètres PEFT :

Le fine-tuning traditionnel des modèles de langage pré-entraînés (PLMs) nécessite la mise à jour de tous les paramètres du modèle, ce qui est coûteux en termes de calcul et nécessite une énorme quantité de données.

Le Fine-Tuning Efficace en Paramètres (PEFT) fonctionne en ne mettant à jour qu'un petit sous-ensemble des paramètres les plus influents du modèle, le rendant beaucoup plus efficace.

j. Paramètres d'entraînement :

- ✓ **output_dir** : Le répertoire de sortie où les prédictions du modèle et les checkpoints seront stockés.
- ✓ **num_train_epochs** : Une époque d'entraînement.
- ✓ **fp16/bf16** : Désactiver l'entraînement en fp16/bf16.
- ✓ **per_device_train_batch_size** : Taille des lots par GPU pour l'entraînement.
- ✓ **per_device_eval_batch_size** : Taille des lots par GPU pour l'évaluation.

- ✓ **gradient_accumulation_steps** : Nombre d'étapes nécessaires pour accumuler les gradients pendant le processus de mise à jour.
- ✓ **gradient_checkpointing** : Activation de la sauvegarde des gradients.
- ✓ **max_grad_norm** : Clipping des gradients.
- ✓ **learning_rate** : Taux d'apprentissage initial.
- ✓ **weight_decay** : La décroissance de poids s'applique à toutes les couches sauf les poids bias/LayerNorm.
- ✓ **optim** : Optimiseur du modèle (optimiseur AdamW).
- ✓ **lr_scheduler_type** : Type de planification du taux d'apprentissage.
- ✓ **max_steps** : Nombre d'étapes d'entraînement.
- ✓ **warmup_ratio** : Ratio d'étapes pour un warmup linéaire.
- ✓ **group_by_length** : Peut améliorer significativement les performances et accélérer le processus d'entraînement.
- ✓ **save_steps** : Sauvegarder le checkpoint tous les 25 pas de mise à jour.
- ✓ **logging_steps** : Enregistrer les logs tous les 25 pas de mise à jour.

k. Fine-tuning du modèle :

Le Fine-Tuning supervisé (SFT) est une étape clé dans l'apprentissage par renforcement à partir du retour d'information humain (RLHF). La bibliothèque TRL de HuggingFace fournit une API facile à utiliser pour créer des modèles SFT et les entraîner sur notre jeu de données avec seulement quelques lignes de code. Elle est équipée d'outils pour entraîner des modèles de langage en utilisant l'apprentissage par renforcement, en commençant par le fine-tuning supervisé, puis la modélisation de la récompense, et enfin l'optimisation de la politique proximale (PPO).

Nous allons fournir à SFT Trainer le modèle, le jeu de données, la configuration LoRA, le tokenizer et les paramètres d'entraînement.

Nous allons utiliser '. train ()' pour affiner le modèle LLAMA 2 sur un nouveau jeu de données.

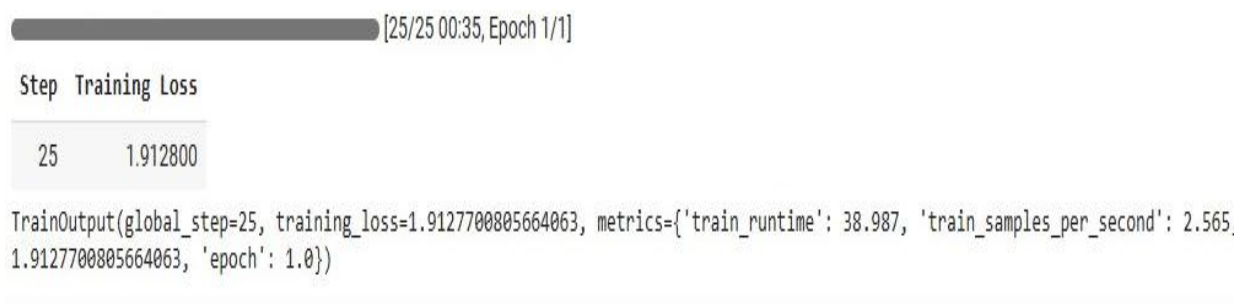


Figure 17 : Résultats de l'entraînement du modèle

I. Résultats d'Entraînement :

Step	Training Loss
25	1.9128

Figure 18 : Perte d'entraînement après 25 étapes

Métrique	Valeur
Temps d'entraînement (s)	38.987
Échantillons par seconde	2.565
Étapes par seconde	0.641
Total FLOs	178039485235200.0
Perte d'entraînement	1.9128
Époque	1.0

Figure 19 : Métriques de performance de l'entraînement

La perte d'entraînement de 1.9128 après 25 étapes indique que le modèle a encore des erreurs significatives dans ses prédictions. Une perte plus faible est généralement meilleure car cela signifie que le modèle prédit plus précisément.

- **Temps d'entraînement** : L'entraînement a duré environ 39 secondes, ce qui montre que le processus d'entraînement était relativement rapide.
- **Débit de traitement** : Le modèle a traité environ 2.565 échantillons par seconde et a complété 0.641 étapes par seconde. Ces métriques indiquent la vitesse et l'efficacité de l'entraînement.
- **Complexité Computationnelle** : Le nombre total d'opérations en virgule flottante (FLOs) est de 178039485235200.0, ce qui montre la complexité des calculs nécessaires pour l'entraînement du modèle.

Après avoir entraîné le modèle, nous allons sauvegarder l'adaptateur de modèle et les tokenizers.

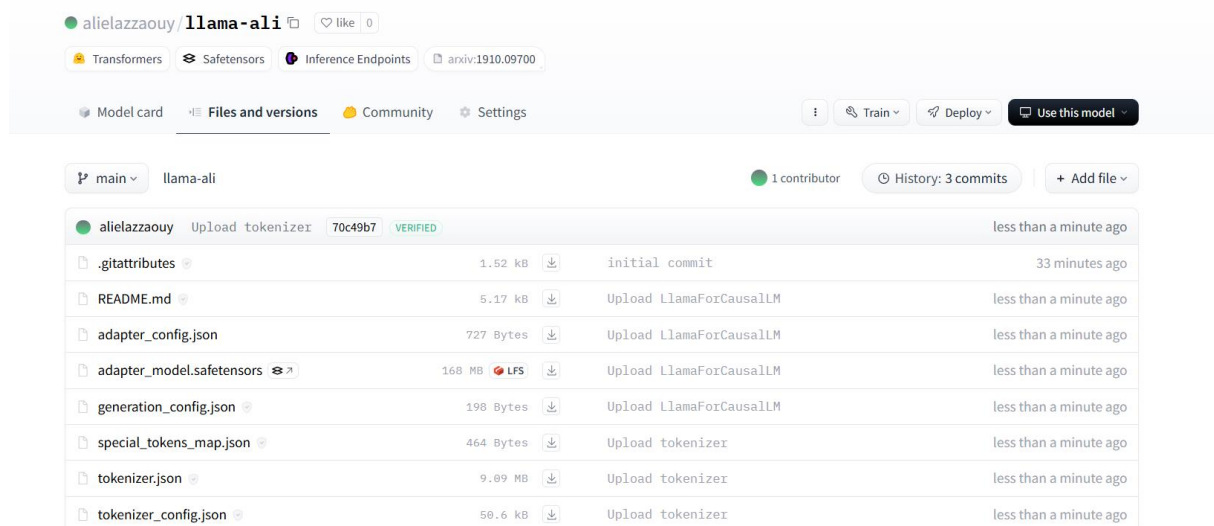


Figure 20 : Liste des fichiers sauvegardés

Pour tester notre modèle affiné, nous utiliserons le pipeline de génération de texte des transformers et poserons des questions simples comme « Quels sont les effets de fibres sur Descente d'organes ? ».

```
23]: FastLanguageModel.for_inference(model)
inputs = tokenizer(
    [
        qa_prompt.format(
            "Is SANY responsible for any consequences resulting from using the machine outside its specified range?", # instruction
            "", # output - Leave this blank for generation!
        )
    ], return_tensors = "pt").to("cuda")

from transformers import TextStreamer
text_streamer = TextStreamer(tokenizer)
_ = model.generate(**inputs, streamer = text_streamer, max_new_tokens = 128)

<|begin_of_text|>Below is an input that provides context. Write a response that appropriately completes the request.

### Input:
Is SANY responsible for any consequences resulting from using the machine outside its specified range?

### Response:
No. SANY is not responsible for any consequences resulting from using the machine outside its specified range.<|end_of_text|>
```

Figure 21 : Résultat du test

2.3.5. Fine-Tuning du Modèle Gemma :

a. Sélection du Modèle Gemma 2b en :

Nous avons sélectionné le modèle Gemma 2b en pour Gemini, disponible sur Hugging Face. Gemma (le mot latin pour « pierre précieuse ») est une famille de modèles ouverts texte-texte, avec décodeur uniquement, développés par diverses équipes de Google, en particulier Google DeepMind. Il s'inspire des modèles Gemini et est conçu pour être léger et compatible avec tous les principaux Frameworks.

Google a publié des poids de modèle pour deux tailles de Gemma, à savoir Gemma 2B et Gemma 7B, qui sont disponibles avec des variantes pré-entraînées et adaptées aux instructions telles que Gemma 2B-it et Gemma 7B-it. Ce modèle est bien adapté pour des tâches de prédiction et de génération de texte dans notre domaine d'application.

Accès à Gemma :

Pour accéder à Gemma, nous avons suivi les étapes suivantes : Premièrement nous avons suivi les instructions fournies pour configurer Gemma sur Kaggle.com.

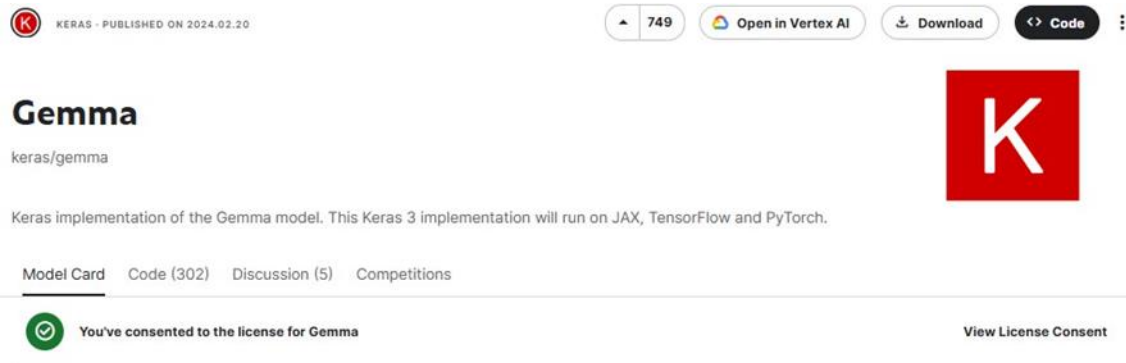


Figure 22 : License for Gemma

Configuration de l'environnement Colab :

Ensuite, nous avons configuré l'environnement Colab en sélectionnant le runtime approprié avec un GPU T4 et en configurant les clés d'API Kaggle dans l'environnement Colab à l'aide des fonctions fournies.

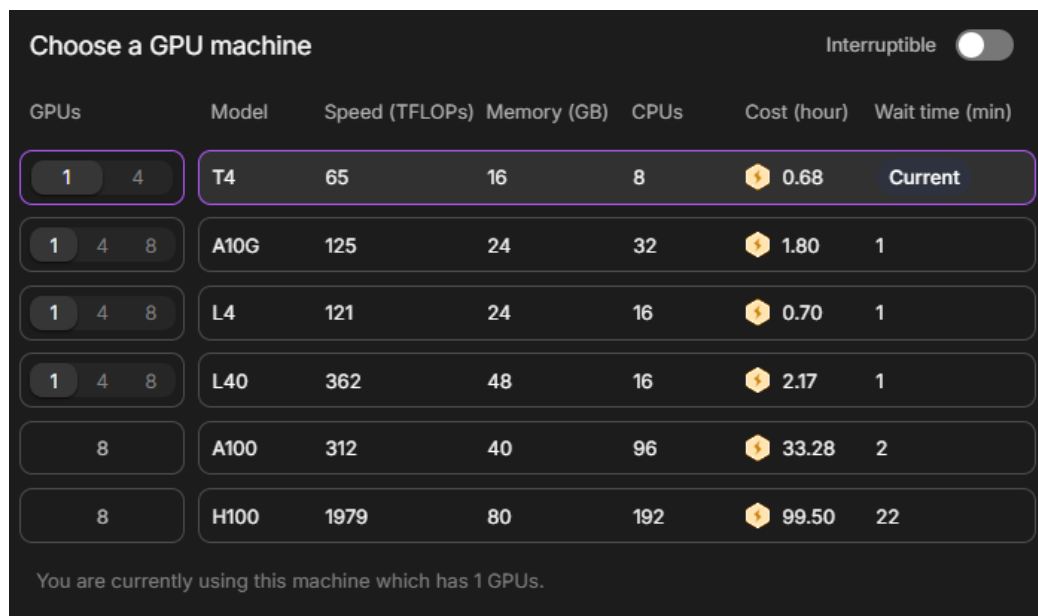


Figure 23 : T4 GPU de Google Colab

Chargement du Jeu de Données :

Nous avons commencé par charger le jeu de données "diseases foods to avoid" héberger dans huggingface utilisé pour le fine-tuning. Après cela, nous avons effectué des opérations de filtrage et de nettoyage pour éliminer les exemples de données non pertinentes. Enfin, nous avons mis en forme les données dans un format adapté à l'entraînement du modèle, en veillant à ce qu'elles soient prêtes pour le processus de fine-tuning.

```
⚡ ~ nvidia-smi
Wed Sep 18 21:22:34 2024

+-----+
| NVIDIA-SMI 535.183.06                  Driver Version: 535.183.06   CUDA Version: 12.2   |
+-----+-----+
| GPU  Name                Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |      Memory-Usage | GPU-Util  Compute M. |
|                                           | MIG M.         |
+-----+-----+
|  0   Tesla T4               Off        | 00000000:00:1E:0 Off |                    0 |
| N/A   29C    P8              8W / 70W |  0MiB / 15360MiB |      0%      Default |
+-----+-----+

+-----+
| Processes: |
| GPU  GI   CI        PID   Type   Process name                        GPU Memory |
|      ID   ID                                   |             Usage |
+-----+-----+
|  No running processes found |
+-----+

⚡ ~ |
```

Figure 24 : Format des données pour l'entraînement de Gemma

Chargement du modèle :

La bibliothèque KerasNLP fournit des implémentations de nombreuses architectures de modèles populaires. Nous avons créé un modèle à l'aide de Gemma Causal LM, un modèle Gemma de bout en bout pour la modélisation du langage causal. Un modèle de langage causal prédit le token suivant en fonction des tokens précédents.

Delimiter: .		question	answer
1		What is the purpose of this manual?	This manual provides operation and maintenance information for the SW405K wheel loader.
2		What could happen if the machine is operated or maintained improperly?	Death or serious injury could result.
3		Who should operate and maintain this machine?	Only trained and experienced personnel should operate and maintain this machine.
4		Why is it important to read and understand this manual?	Before operation or service, and all personnel involved with the machine should periodically read it to remain knowledgeable on its operation and service.
5		What is the purpose of the items addressed in this manual?	Point out possible hazardous situations, increase machine efficiency, prolong the service life of the machine, and reduce maintenance costs.
6		Can changes in the design of this machine lead to new information not covered in this manual?	Yes, continuing improvements in the design of this machine can lead to changes which may not be covered in this manual.
7		What should you do if you have questions about the information in this manual?	Contact a SANY dealer for the latest available information on the machine or to answer any questions regarding information in this manual.
8		What documentation package is provided for this machine?	The documentation package includes the operation and maintenance manual, parts manual, and maintenance log.
9		What must be stored in the machine or be accessible to the operator at all times?	A copy of the operation and maintenance manual must be stored in the machine or be accessible to the operator at all times.
10		What is the purpose of the SW405K wheel loader operation and maintenance manual?	The purpose of this manual is to provide operation and maintenance information for the SW405K wheel loader.
11		What is the consequence of unsafe operation or maintenance of the SW405K wheel loader?	Unsafe operation and maintenance of this machine could result in death or serious injury.
12		Who should operate and maintain the SW405K wheel loader?	This machine must be operated and maintained by trained and experienced personnel.
13		What should be done before operating or servicing the SW405K wheel loader?	Do not operate or work on this machine without first reading and understanding this operation and maintenance manual.
14		How can the operator or service personnel benefit from reading this manual?	Identify hazardous situations, increase machine efficiency during operation, prolong the service life of the machine, and reduce maintenance costs.
15		Can changes in the design of the SW405K wheel loader lead to changes not covered in the manual?	Contact a SANY dealer for the latest available information on the machine or to answer any questions regarding information in this manual.
16		What should be done with the documentation of the SW405K wheel loader?	The documentation should not be used with any other machine and should be made available to all service personnel.
17		What should be done with the operation and maintenance manual be stored or kept?	It should be provided to the new owner if the machine is sold. It should also be made available to maintenance personnel when servicing the machine.
18		What is the purpose of this manual?	To provide operation and maintenance information for the SW405K wheel loader.
19		What is the consequence of unsafe operation or maintenance of this machine?	Unsafe operation and maintenance of this machine could result in death or serious injury.

Figure 25 : Modèle Gemma

La méthode `from_preset` instancie le modèle à partir d'une architecture et de pondérations prédéfinies. Dans le code ci-dessus, la chaîne « `gemma_2b_en` » spécifie l'architecture prédéfinie - un modèle Gemma avec 2 milliards de paramètres.

Voici un tableau qui compare Gemma 2b et Gemma 7b :

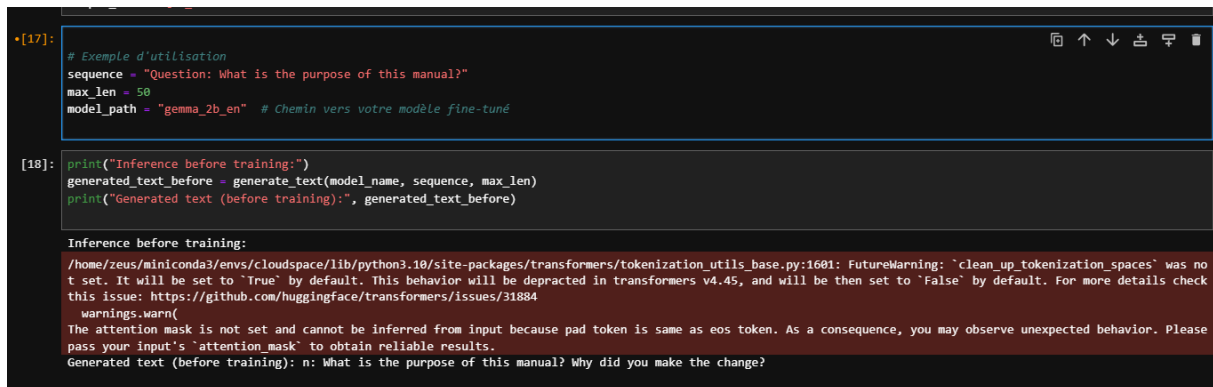
Parameters size	Input	Output	Tuned versions	Intended platforms
2B	Text	Text	Pretrained Instruction tuned	Mobile devices and laptops
7B	Text	Text	Pretrained Instruction tuned	Desktop computers and small servers

Figure 26 : Spécifications des modèles

b. Inférence avant Fine Tuning :

Dans cette section, nous avons interrogé le modèle avec différentes invites pour voir comment il répond.

Exemple : What is the purpose of this manual ?



```
[17]: # Exemple d'utilisation
sequence = "Question: What is the purpose of this manual?"
max_len = 50
model_path = "gemma_2b_en" # Chemin vers votre modèle fine-tuné

[18]: print("Inference before training:")
generated_text_before = generate_text(model_name, sequence, max_len)
print("Generated text (before training):", generated_text_before)

Inference before training:
/home/zeus/miniconda3/envs/cloudspace/lib/python3.10/site-packages/transformers/tokenization_utils_base.py:1681: FutureWarning: `clean_up_tokenization_spaces` was not set. It will be set to `True` by default. This behavior will be deprecated in transformers v4.45, and will be then set to `False` by default. For more details check this issue: https://github.com/huggingface/transformers/issues/31884
  warnings.warn(
The attention mask is not set and cannot be inferred from input because pad token is same as eos token. As a consequence, you may observe unexpected behavior. Please pass your input's `attention_mask` to obtain reliable results.
Generated text (before training): n: What is the purpose of this manual? Why did you make the change?
```

Figure 27 : ELI5 Photosynthesis Prompt pré-entraîné

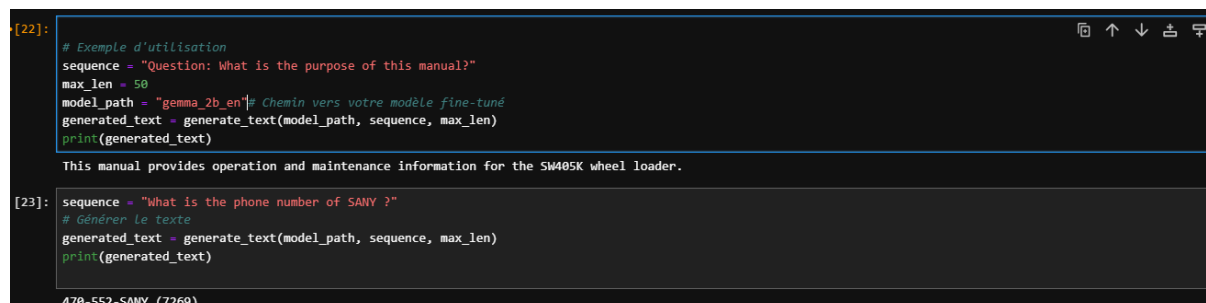
Ce dernier modèle de réponse contient des mots qui pourraient ne pas être faciles à comprendre, comme velvety.

c. Fine-Tuning avec KERAS et LoRA :

Nous avons activé LoRA (Low Rank Adaptation) sur le modèle Gemma pour réduire le nombre de paramètres entraînaables et accélérer le processus de fine-tuning. Cette étape a impliqué une explication détaillée de la façon dont LoRA fonctionne et comment il impacte le modèle. Ensuite, nous avons configuré l'entraînement du modèle Gemma après l'activation de LoRA en choisissant un optimiseur approprié (AdamW), en déterminant la longueur maximale de la séquence d'entrée, et en compilant le modèle avec des métriques adaptées. Enfin, nous avons entraîné le modèle Gemma avec les données prétraitées, en surveillant attentivement les métriques de performance pour évaluer l'efficacité du fine-tuning.

d. Inférence après Fine Tuning :

Après le fine tuning, les réponses suivent les instructions fournies dans le prompt.



```
[22]: # Exemple d'utilisation
sequence = "Question: What is the purpose of this manual?"
max_len = 50
model_path = "gemma_2b_en" # Chemin vers votre modèle fine-tuné
generated_text = generate_text(model_path, sequence, max_len)
print(generated_text)

This manual provides operation and maintenance information for the SW405K wheel loader.

[23]: sequence = "What is the phone number of SANY ?"
# Générer le texte
generated_text = generate_text(model_path, sequence, max_len)
print(generated_text)

470-552-SANY (7269).
```

Figure 28 : ELI5 Photosynthesis Prompt entraîné

3. Déploiement des Modèles :

Pour garantir une accessibilité optimale de nos modèles, nous avons procédé à leur déploiement. Ainsi, nous avons choisi de publier notre modèle Llama2 sur Hugging Face, permettant une large diffusion et une facilité d'utilisation pour la communauté. Parallèlement, nous avons également poussé notre modèle Gemma sur Hugging Face également, offrant ainsi une autre option pour les utilisateurs souhaitant exploiter nos travaux dans des environnements de données compétitifs et collaboratifs. Une fois ces déploiements effectués, nous visons ensuite à intégrer nos modèles dans une application Streamlit. Cette interface utilisateur interactive permet de tester et de visualiser les performances de nos modèles de manière intuitive et conviviale, facilitant ainsi leur adoption par un public plus large.

4. Conclusion :

Ce chapitre a mis en lumière l'ensemble du processus de modélisation, d'évaluation et de déploiement des modèles d'intelligence artificielle générative, qui constituent le cœur de notre projet. En passant par l'utilisation des plateformes Hugging Face et Kaggle, nous avons été en mesure de sélectionner, fine-tuner et évaluer des modèles de pointe tels que LLAMA 2 et Gemma, afin d'optimiser les prédictions dans notre domaine spécifique.

CHAPITRE V : CREATION D'UN PROTOTYPE DE PIPELINE POUR ALIMENTER UN TABLEAU DE BORD EN TEMPS REEL

1. Introduction :

Le projet de Tableau de Bord des Ventes en Temps Réel est une application complète de traitement de données en temps réel et de visualisation web. Ce projet vise à ingérer, traiter, et visualiser les données de ventes en temps réel. Les données sont transmises de Kafka à HDFS, traitées par Spark, puis visualisées sur un tableau de bord dynamique créé avec Flask. L'actualisation en temps réel du tableau de bord est assurée par WebSockets, permettant d'afficher constamment les données les plus récentes.

2. Qu'est-ce que l'ETL ?

L'ETL, pour "extraction, transformation et chargement", est un processus de gestion des données qui permet d'unifier et d'intégrer des informations issues de diverses sources en les combinant dans un seul ensemble cohérent, prêt à être stocké dans un entrepôt de données, un data Lake ou un autre système cible.

Introduit dans les années 1970 avec la montée en popularité des bases de données, l'ETL est devenu un processus central pour l'intégration et le traitement des données, principalement utilisé dans les projets d'entreposage de données pour les besoins analytiques et de calcul.

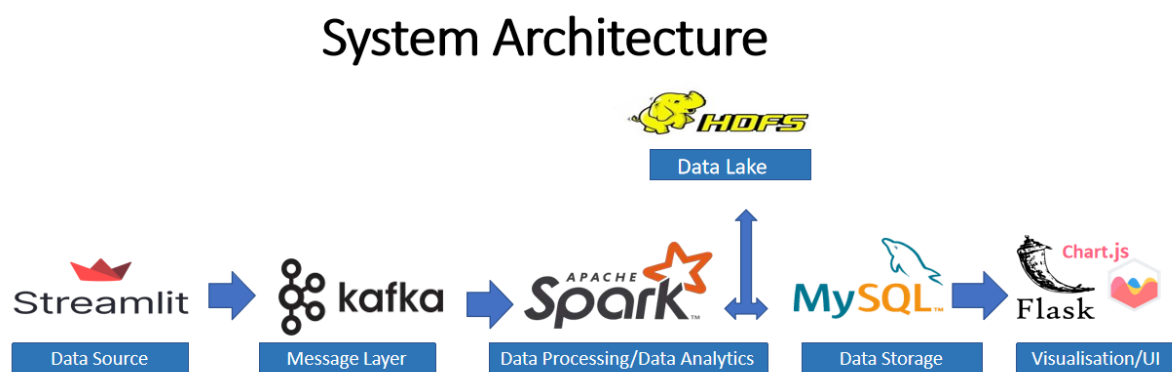
L'ETL joue un rôle essentiel dans les flux de travail liés à l'analyse des données et au machine Learning. Il permet de nettoyer et structurer les données selon des règles métier spécifiques afin de répondre aux besoins de la Business Intelligence, comme la génération de rapports mensuels, tout en permettant également des analyses plus complexes susceptibles d'améliorer les processus internes ou l'expérience utilisateur. Les entreprises utilisent généralement l'ETL pour:

- Extraire les données de systèmes existants.
- Nettoyer et harmoniser ces données afin d'améliorer leur qualité.
- Charger les données dans une base de données cible.

3. Besoin :

Dans un environnement où les décisions doivent être prises rapidement, disposer d'un outil capable de visualiser en temps réel les ventes et autres indicateurs commerciaux est crucial pour les entreprises. Ce projet répond au besoin de collecter et d'analyser des données en continu, tout en offrant une interface utilisateur intuitive pour une prise de décision efficace.

4. Architecture du Système :



L'architecture du système est composée des éléments suivants :

- 1. Flux de Données :** Les données en temps réel sont envoyées de Kafka à Spark pour être traitées et partitionnées par date et heure, avant d'être écrites dans HDFS en format Parquet.
- 2. Framework Web :** Flask sert l'application web frontale, affichant les données de ventes en temps réel avec des graphiques mis à jour dynamiquement. Les données sont extraites de MySQL et transmises au front-end via WebSockets pour des mises à jour instantanées.
- 3. HDFS :** Le système de fichiers distribué Hadoop (HDFS) stocke les données traitées de manière évolutive et fiable.
- 4. Base de Données MySQL :** Agit comme la couche de stockage des données traitées, qui sont ensuite interrogées par l'application Flask pour générer des visualisations en temps réel.

5. Technologies Utilisées :

5.1. HDFS :

HDFS est un système de fichiers distribué qui gère de grands ensembles de données s'exécutant sur du matériel de base. Il est utilisé pour faire évoluer un seul cluster Apache Hadoop vers des centaines (voire des milliers) de nœuds. HDFS est l'un des principaux composants d'Apache Hadoop, les autres étant MapReduce et YARN. HDFS ne doit pas être confondu avec ou remplacé par Apache HBase, qui est un système de gestion de base de données non relationnelle orienté colonnes qui repose sur HDFS et peut mieux prendre en charge les besoins des données en temps réel grâce à son moteur de traitement en mémoire.



5.2. Spark :

Spark est un système de traitement rapide et parallèle. Il fournit des APIs de haut niveau en Java, Scala, Python et R, et un moteur optimisé qui supporte l'exécution des graphes. Il supporte également un ensemble d'outils de haut niveau tels que Spark SQL pour le support du traitement de données structurées, MLlib pour l'apprentissage des données, GraphX pour le traitement des graphes, et Spark Streaming pour le traitement des données en streaming.



5.3. Kafka :

Kafka est une plateforme de messagerie distribuée open-source, initialement développée par LinkedIn avant d'être adoptée par la Fondation Apache. Conçu pour traiter des flux massifs de données en temps réel, Kafka se distingue par sa durabilité, son évolutivité et son architecture distribuée. Il est largement utilisé pour créer des pipelines de données, diffuser des flux de données en continu, gérer des événements en temps réel, et répondre à divers besoins similaires.

Grâce à son modèle de journal de transactions distribué, Kafka assure un stockage fiable et efficace des données tout en permettant aux applications de les lire et de les écrire en continu. Ce système est très prisé dans les architectures modernes de traitement des données, notamment dans les environnements de streaming et de traitement en temps réel.



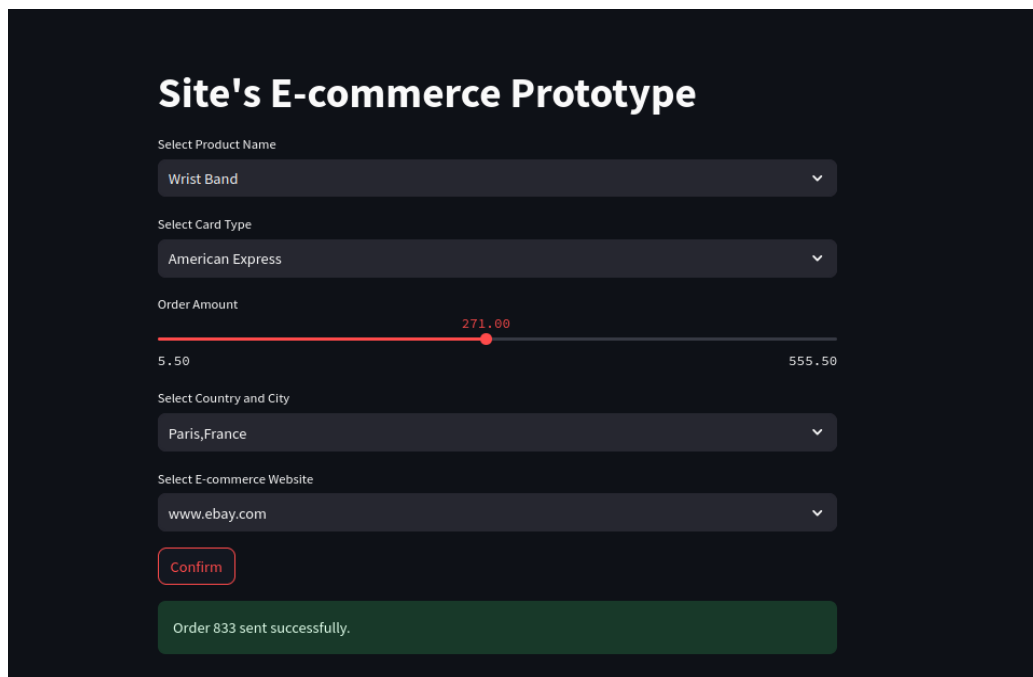
5.4. MySQL :

MySQL est un système de gestion de bases de données relationnelles open-source, largement utilisé pour le stockage, la gestion et la récupération de données structurées. Développé à l'origine par MySQL AB et maintenant maintenu par Oracle Corporation, il est reconnu pour sa simplicité, sa rapidité et sa fiabilité.

MySQL fonctionne selon un modèle client-serveur, où le serveur MySQL gère les bases de données et répond aux requêtes SQL (Structured Query Language) envoyées par les clients. Il prend en charge les opérations complexes telles que les transactions, les index, les jointures, et les relations entre les tables, tout en garantissant l'intégrité des données.



6. Résultats :



The image shows a dark-themed web form titled "Site's E-commerce Prototype". It contains several input fields: "Select Product Name" with a dropdown menu showing "Wrist Band"; "Select Card Type" with a dropdown menu showing "American Express"; "Order Amount" with a slider ranging from 5.50 to 555.50, currently set at 271.00; "Select Country and City" with a dropdown menu showing "Paris, France"; and "Select E-commerce Website" with a dropdown menu showing "www.ebay.com". Below these fields is a red "Confirm" button. At the bottom, a green message box states "Order 833 sent successfully."

Figure 29 : Producer

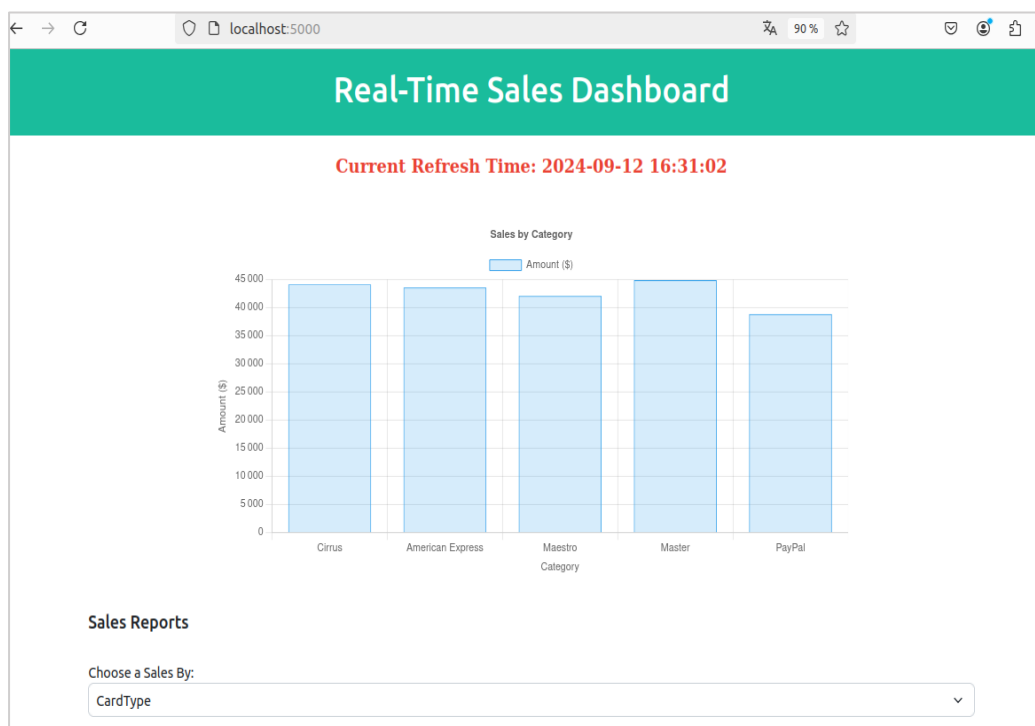


Figure 30 : Consumer

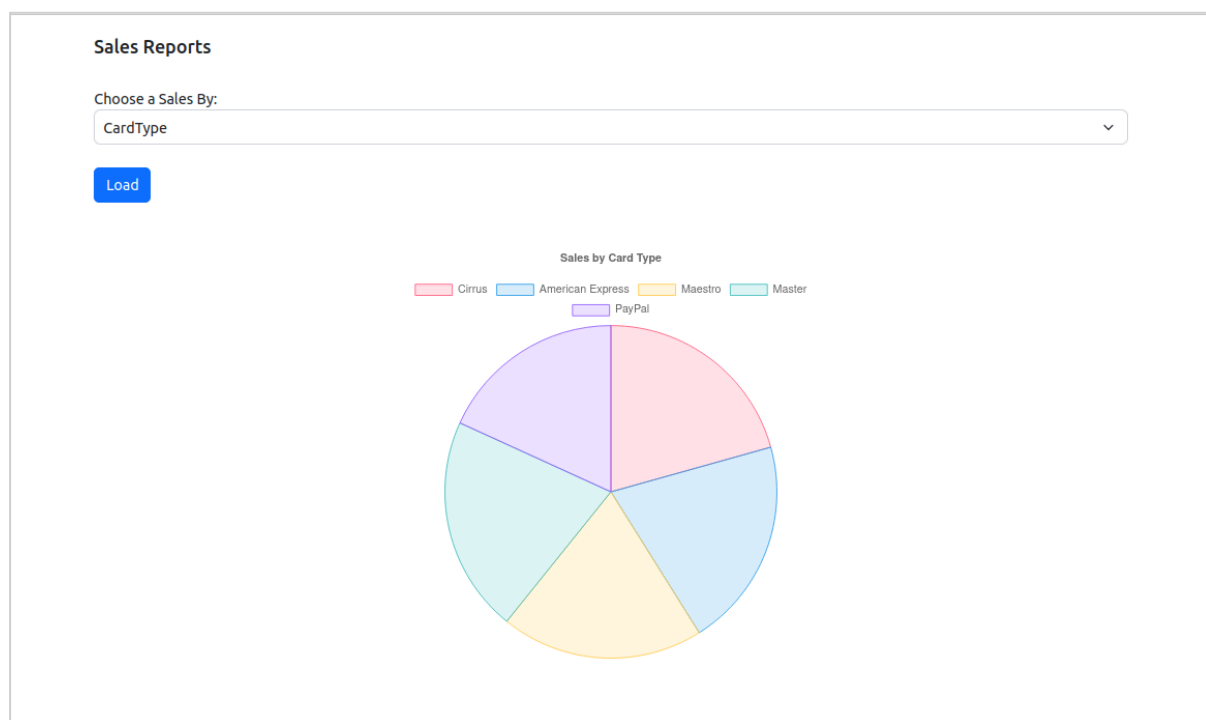


Figure 31 : Consumer 2

CONCLUSION GENERALE

Ce projet, intitulé **"Un système intelligent d'assistance pour la préparation des machines et l'accès aux informations techniques"**, a permis de développer une solution complète et innovante. Celle-ci intègre des techniques avancées de web scraping, de traitement et de modélisation des données, ainsi que le fine-tuning de modèles d'intelligence artificielle. À travers les différentes étapes du projet, depuis le benchmarking des outils d'IA générative jusqu'à la collecte, la structuration des données, et le déploiement des modèles, nous avons réalisé des avancées significatives.

Nous avons mené une étude approfondie des différentes solutions d'IA disponibles sur le marché. Après avoir établi des critères de comparaison en mettant l'accent sur l'open-source, nous avons exploré ces outils afin de mieux comprendre l'architecture générale des IA génératives pour le texte et de réaliser un benchmarking.

Un processus rigoureux de collecte de données via le web scraping a été mis en place, ciblant des documents techniques contenant des informations sur les machines. Les données recueillies ont ensuite été nettoyées et structurées pour garantir leur qualité et leur adéquation avec les étapes de modélisation.

En utilisant des modèles d'IA générative tels que GPT-2, LLaMA 3, et Gemma, nous avons effectué un fine-tuning sur des données spécifiques à ce projet de stage. Cette démarche nous a permis de personnaliser les modèles et d'améliorer leur capacité à générer des recommandations précises et adaptées aux problèmes techniques rencontrés sur les machines.

Un pipeline a été développé pour alimenter un tableau de bord en temps réel, utilisant des technologies complexes comme Apache Spark pour le traitement en temps réel et Apache Kafka comme outil d'intégration, facilitant l'injection des données dans des topics dédiés.

Tout au long du projet, nous avons rencontré plusieurs contraintes, notamment liées aux ressources GPU nécessaires pour le déploiement des modèles fine-tunés. La taille et la complexité des modèles ont parfois dépassé les capacités des ressources disponibles, rendant l'entraînement et le déploiement moins efficaces.

Ce projet ouvre des perspectives d'améliorations et d'extensions futures :

- Extension de la base de données : Enrichir la base de données avec de nouvelles sources permettra d'améliorer encore la précision et la couverture du système
- Amélioration des modèles : En explorant d'autres architectures d'IA et en poursuivant le fine-tuning avec des techniques avancées, nous pourrons optimiser les performances des modèles pour obtenir des résultats plus pertinents et précis.
- Optimisation des ressources : Explorer des solutions pour optimiser l'utilisation des ressources GPU, telles que des techniques de compression de modèles ou des infrastructures cloud performantes, afin de surmonter les limitations actuelles et d'améliorer l'efficacité du déploiement.

WEBOGRAPHIE

- ✓ <https://openai.com/chatgpt> (Consulté le 27 Juillet 2024)
- ✓ <https://en.wikipedia.org/wiki/BERT> (language model) (Consulté le 29 Juillet 2024)
- ✓ <https://arxiv.org/abs/1906.08237> (Consulté le 29 Juillet 2024)
- ✓ <https://github.com/google-research/text-to-text-transfer-transformer> (Consulté le 29 Juillet 2024)
- ✓ <https://research.facebook.com/publications/bart-denoising-sequence-to-sequence-pre-training-for-natural-language-generation-translation-and-comprehension/> (Consulté le 29 Juillet 2024)
- ✓ <https://deepmind.google/technologies/gemini/#gemini-1.0> (Consulté le 29 Juillet 2024)
- ✓ <https://www.microsoft.com/en-us/bing/do-more-with-ai/bing-ai-features?form=MA13KP> (Consulté le 29 Juillet 2024)
- ✓ <https://ai.meta.com/blog/large-language-model-llama-meta-ai/> (Consulté le 29 Juillet 2024)
- ✓ <https://www.techopedia.com/definition/jasper-ai> (Consulté le 31 Juillet 2024)
- ✓ <https://medium.com/geekculture/what-does-cohere-do-cdadf6d70435> (Consulté le 31 Juillet 2024)
- ✓ <https://analyticsindiamag.com/discovering-cohere-ai-and-how-its-different-from-openai/> (Consulté le 31 Juillet 2024)

- ✓ <https://www.blogdumoderateur.com/etude-50-outils-ia-generative-plus-utilises-2023/> (Consulté le 31 Juillet 2024)
- ✓ <https://www.aixploria.com/claude/> (Consulté le 31 Juillet 2024)
- ✓ <https://em360tech.com/tech-article/large-language-model> (Consulté le 1 août 2024)
- ✓ <https://llmmodels.org/> (Consulté le 1 août 2024)
- ✓ <https://www.impli.fr/outil/copy-ai> (Consulté le 1 août 2024)
- ✓ <https://support.anthropic.com/fr/articles/7989434-qu-est-ce-que-claude> (Consulté le 1 août 2024)
- ✓ <https://www.blogdumoderateur.com/tools/claude/> (Consulté le 1 août 2024)
- ✓ <https://www.techopedia.com/definition/lamda-google-lamda> (Consulté le 1 août 2024)
- ✓ LaMDA (Consulté le 1 août 2024)
- ✓ <https://www.ibm.com/topics/api> (Consulté le 1 août 2024)
- ✓ <https://arxiv.org> (Consulté le 1 août 2024)
- ✓ <https://arxiv.org/abs/2305.14314> (Consulté le 2 septembre 2024)
- ✓ <https://www.datacamp.com/tutorial/fine-tuning-google-gemma> (Consulté le 4 septembre 2024)

- ✓ <https://huggingface.co/google/gemma-7b> (Consulté le 4 septembre 2024))
- ✓ https://huggingface.co/datasets/OumaimaABJAOU/Disease_food_interaction
- ✓ <https://huggingface.co/datasets/OumaimaGHAZOUAN/data.jsonl>