RÉPUBLIQUE ALGERIENNE DÉMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

École Nationale Polytechnique

Département du Génie Industriel

Mémoire de Projet de Fin d'Études
En vue de l'obtention du diplôme d'Ingénieur d'État en Génie Industriel
Option : Data Science et Intelligence Artificielle

Création d'un modèle d'Intelligence Artificielle de génération de textes pour la description technique de projets.

Application : MB inc

Réalisé par :
Sarah BOUARABA
Yousra LAIB

Encadré par :
Mr Oussama ARKI
Mr Khaled BOUAZIZ

Présenté et soutenu publiquement le (04/07/2023)

Composition du Jury

Président	Mr. Iskander ZOUAGHI	MCA	ENP
Examinateur	Mr. Hakim FOURAR-LAIDI	MCA	ENP
Promoteur	Mr. Oussama ARKI	MCB	ENP
Invité	Mr. Khaled BOUAZIZ	Consultant	MB inc

ENP 2023

RÉPUBLIQUE ALGERIENNE DÉMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

École Nationale Polytechnique

Département du Génie Industriel

Mémoire de Projet de Fin d'Études
En vue de l'obtention du diplôme d'Ingénieur d'État en Génie Industriel
Option : Data Science et Intelligence Artificielle

Création d'un modèle d'Intelligence Artificielle de génération de textes pour la description technique de projets.

Application : MB inc

Réalisé par :
Sarah BOUARABA
Yousra LAIB

Encadré par :
Mr Oussama ARKI
Mr Khaled BOUAZIZ

Présenté et soutenu publiquement le (04/07/2023)

Composition du Jury

Président	Mr. Iskander ZOUAGHI	MCA	ENP
Examinateur	Mr. Hakim FOURAR-LAIDI	MCA	ENP
Promoteur	Mr. Oussama ARKI	MCB	ENP
Invité	Mr. Khaled BOUAZIZ	Consultant	MB inc

ENP 2023

Dédicaces

C'est avec une immense gratitude et une profonde humilité que je prends un moment pour reconnaître ceux qui ont joué un rôle essentiel dans mon parcours académique et personnel. Alors que je réfléchis aux années passées, je me rends compte à quel point chaque interaction, chaque geste de soutien et chaque mot d'encouragement ont été déterminants dans l'achèvement de cette importante étape de ma vie.

Ce travail est le résultat non seulement de mes efforts, mais aussi de l'influence, du soutien et de la confiance de nombreuses personnes. Aujourd'hui, je souhaite prendre un instant pour leur exprimer ma reconnaissance et leur dédier ce travail. Ils sont l'inspiration silencieuse derrière chaque page, chaque mot et chaque idée qui ont trouvé leur place dans ce mémoire.

Je dédie ce travail en premier lieu à mes parents, les personnes les plus dignes d'admiration à mes yeux, qui ont été ma source de soutien inébranlable tout au long de cette aventure.

Mon père, un homme exceptionnel, a toujours été là pour moi. Il a été mon roc, ma boussole, et mon guide. Sa sagesse et sa bienveillance m'ont permis de surmonter tous les obstacles et de grandir en tant qu'individu. Il m'a encouragé à poursuivre mes rêves, à travailler dur et à croire en moi-même. Son amour inconditionnel et sa présence constante ont été un véritable moteur dans ma vie.

Mes deux mamans, chacune à leur manière, ont joué un rôle essentiel dans mon parcours. L'une d'entre elles m'a accompagné tout au long de ma croissance, insufflant des valeurs essentielles telles que la persévérance, l'empathie et l'ouverture d'esprit. Elle a été ma confidente, mon modèle et ma meilleure amie. Grâce à elle, j'ai appris à apprécier chaque moment de la vie et à toujours chercher le positif.

L'autre maman, malheureusement partie trop tôt, continue à veiller sur moi depuis les cieux. Son amour éternel et son souvenir restent gravés dans mon cœur. Elle m'a transmis sa force intérieure et son courage, me rappelant que rien n'est impossible lorsque l'on croit en soi. Je sens sa présence dans les moments de doute et sa voix résonne dans ma tête lorsqu'il s'agit de prendre des décisions importantes.

À vous qui vous êtes sacrifiés pour élever vos cinq enfants et faire d'eux des personnes qui ont réussi à partir de rien, votre amour inconditionnel, votre soutien indéfectible et votre dévouement sans faille ont été les fondements de notre réussite. Je ne vis que pour vous rendre heureux et fiers, et j'espère sincèrement y parvenir. Je vous aime plus que tout et je vous suis éternellement reconnaissante.

À mes deux grands frères, Yacine et Haroune, qui ont grandement influencé ma vie avec leurs sages conseils et leur soutien immuable. À mon cadet, Younes, un rayon de soleil dans nos vies, dont la curiosité et la joie de vivre ne cessent de nous émerveiller. Et enfin, à Ines, ma précieuse et unique sœur, l'étoile scintillante dans mon ciel. Sa présence apporte toujours de la douceur et du réconfort dans les moments les plus sombres. Je rends hommage à votre présence dans ma vie, et je chéris chaque moment passé avec vous. Vous êtes des pierres précieuses dans le jardin de mon existence, et je suis infiniment chanceuse de vous avoir dans ma vie.

À mes amies de longue date qui ont toujours montré la beauté de l'amitié sincère et indéfectible. Sarah, mon amie depuis le lycée, l'amie la plus unique et inoubliable que j'aie jamais connue. Feriel et Selma, mes amies d'enfance, nos souvenirs partagés, notre enfance, nos rires et nos larmes ont formé le tissu de ma vie. Je vous chéris au plus profond de mon cœur.

À Khadidja et Faten, mes compagnes d'armes durant la prépa. Ensemble, nous avons partagé des moments inoubliables. Nos échecs transformés en leçons, nos succès célébrés comme des triomphes majeurs, nos nuits blanches passées à réviser, nos éclats de rire qui ont égayé les longues heures d'étude, et nos conversations infinies qui ont tissé des liens d'amitié indéfectibles. Mes années de prépa ont été marquées par votre présence, vous avez fait de ces années, souvent redoutées, une expérience enrichissante et pleine de vie.

À IEC, le joyau parmi les clubs estudiantins, celui qui a été pour moi un formidable terrain d'apprentissage et de développement personnel, je dédie ces mots. IEC demeurera toujours une source lumineuse d'inspiration et un symbole de fierté pour moi. Ensemble, espérons continuer à briller, en guidant et en éclairant le chemin des futures générations d'Indus.

À Anti-Ghoumma, le meilleur des groupes, et ses membres. Chacun de vous a joué un rôle déterminant dans la création de moments mémorables, peuplés de rires sincères et de divertissements indélébiles. Vous êtes une source constante d'amusement et de bonne humeur, vous êtes une famille choisie, la meilleure qui soit.

À nos précieux alumni Indus - Souad, Hynd, Thafat, Zineb, Maya, Lynda et Anis - dont le soutien, l'orientation et le partage ont été d'une immense valeur, j'adresse mes sincères remerciements. À la communauté polytechnicienne, toujours à l'écoute et prête à aider, j'exprime ma reconnaissance. Notre réseau, unique en son genre, est un formidable maillage de solidarité, d'entraide et de dévouement.

À toutes les promotions futures de la spécialité DSIA, les bébés DAÏTA qui suivront nos pas, cette dédicace vous est dédiée. Que ce travail serve de pierre angulaire à votre édification, et que vous y trouviez l'inspiration pour continuer à faire évoluer cette spécialité. Faites preuve de curiosité, d'innovation et de persévérance, et surtout, n'oubliez jamais de partager vos connaissances avec les autres. Rendez cette spécialité renommée non seulement par la qualité de votre travail, mais aussi par votre engagement envers l'éthique, la collaboration et l'apprentissage continu. Bonne chance à vous toutes et tous, et rendez-nous fiers.

Et le meilleur pour la fin.. À celle qui a été plus qu'une binôme pour moi, ma complice et confidente. Nous avons partagé bien plus que des lignes de code, des analyses de données et des heures de travail acharné. Nous avons partagé des rires, des moments de doute, des victoires, des échecs transformés en leçons, et surtout, une amitié exceptionnelle. Nos discussions interminables, nos débats passionnés et nos moments de complicité ont été une source de réconfort et de joie. C'est à toi Yousra que je dédie le meilleur de cette expérience, car sans toi, cette aventure n'aurait pas été la même.

Malheureusement, je suis limitée en espace pour mentionner tout le monde, mais de nombreuses personnes ont joué un rôle important dans ma vie au cours de ces cinq dernières années, et cette liste est loin d'être exhaustive. Je tiens à remercier toutes les personnes qui, de près ou de loin, ont contribué à ma réussite.

Sarah

Dédicaces

C'est à mes parents, les piliers de ma vie et mes sources d'inspiration inébranlables, que je consacre ce travail. Ils ont toujours été là pour moi, même face à la difficulté de quitter le cocon familial pour poursuivre mes études à l'École Nationale Polytechnique. Leur amour, leur soutien et leur fierté ont été les moteurs qui m'ont permis d'atteindre cette étape.

Ma mère, cette personne inébranlable, a été particulièrement touchée par mon départ. Son soutien inconditionnel m'a encouragé à aller de l'avant, à poursuivre mes rêves, même lorsque ceux-ci divergeaient des siens. Quant à mon père, il a toujours été une source de guidance et d'inspiration pour moi, guidant mes pas avec une détermination inébranlable, souvent au détriment de ses propres aspirations. Je dédie donc ce travail avec une immense gratitude à ces deux figures importantes de ma vie.

Ma douce sœur Lydia est une présence constante, semblable à une étoile scintillante dans ma vie. Sa sagesse résonne comme une mélodie apaisante qui guide mon cœur. Son amour inconditionnel et sa précieuse présence sont des cadeaux que je valorise chaque jour. Je suis infiniment reconnaissante d'avoir une sœur aussi merveilleuse que Lydia. Merci d'être toi, ma chère sœur.

Mes petites sœurs Sidra, Darine, Ranime, Mayar et Sadim, ainsi que mon petit frère Bader, ont également joué un rôle essentiel dans ce parcours. Leur affection inconditionnelle, leur émerveillement face au monde et leur résilience face aux défis ont été pour moi une source constante d'inspiration et de motivation. Leur joie et leur fierté envers leur grande sœur ont illuminé les moments les plus difficiles. Je dédie également ce travail à eux avec un amour immense et une profonde gratitude.

À mes chères amies Samah, Sihem, Amira et Bouthaina, vous êtes mon cœur qui bat à l'unisson, bien plus que de simples compagnons de route.

Samah, ma petite boule d'énergie, ton enthousiasme contagieux et ton sourire lumineux remplissent ma mémoire de moments de joie et de rires inoubliables.

Sihem, ma douce lumière dans l'obscurité, ta sagesse et ta patience m'ont aidée dans les moments difficiles. Tu es mon phare, mon guide, et je te suis éternellement reconnaissante.

Amira, mon rocher de loyauté, ton soutien inébranlable m'a apporté un réconfort immense, me rappelant combien la constance et la confiance sont précieuses.

Bouthaina, mon doux calme dans la tempête, ta sérénité et ta franchise m'ont ouvert de nouvelles voies dans mon voyage à travers la vie.

Votre amitié est un trésor inestimable

Et comment pourrais-je oublier mes précieuses amies, mes "besties" Vous occupez une place spéciale dans mon cœur et dans mon histoire.

Kenza, ma petite étoile filante, ton esprit pétillant et ton courage énorme m'inspirent tous les jours. Ta force intérieure et ton amour sincère pour nos liens d'amitié me font te voir comme un véritable trésor. Je t'admire beaucoup.

Besma, mon doux rayon de soleil, ton empathie et ta gentillesse ont créé une petite bulle de confiance et de compréhension entre nous. Ton soutien indéfectible est comme un câlin chaleureux lors des jours de pluie, apaisant les moments difficiles.

Maissa, ma petite luciole, ton optimisme contagieux et ta capacité à trouver la lumière dans les jours sombres sont comme une lanterne dans la nuit. Ton sourire illumine mon parcours, rappelant que même lors des jours les plus sombres, il y a toujours une lueur d'espoir.

Rania, ma douce source de paix, tu es comme une oasis calme au milieu de la tempête de la vie, toujours là avec de bons conseils. Ta tranquillité et ta sagesse sont comme un rocher solide sur lequel je peux m'appuyer quand je suis incertaine et perdue.

Sofia, ma petite étoile pétillante, ton énergie vibrante donne des couleurs à mes journées. Ton cœur, aussi vaste et généreux que l'océan, est une source inépuisable de joie et d'inspiration. Ta présence est comme une douce brise d'été, me rappelant combien la bienveillance et la simplicité peuvent embellir le monde autour de nous.

Ikram, ma douce échappatoire, ta tranquillité et ta constance sont comme un abri sûr dans le tumulte du quotidien. Tu es une amie fidèle et vraie qui célèbre la vie avec une grâce et une dignité qui forcent l'admiration. Ton soutien et ton amitié sont des trésors inestimables qui rendent ma vie plus riche.

Votre présence a transformé les moments simples en souvenirs précieux, et je suis honorée d'avoir votre amitié qui représente l'une des plus belles parts de mon parcours.

En hommage à mes trois balises lumineuses :

Islem, tel un phare constant dans l'obscurité, tu éclaires mon chemin avec ton courage et ta persévérance. Ta présence, toujours là et réconfortante, est une source constante de force dans ma vie, guidant mes pas avec une détermination qui ne faiblit jamais."

Rached, ton cœur généreux et gentil est comme un rocher solide sur lequel je peux toujours m'appuyer. Tu es non seulement un ami fidèle, mais aussi une source constante de réconfort et de soutien. Ta bienveillance et ta disponibilité pour les autres te rendent vraiment spécial dans ma vie, toujours là avec une main tendue.

Djamel, mon cher frère de cœur, ton aide et tes conseils ont été comme un ancre dans ma vie. Ta présence constante à mes côtés est un véritable trésor. J'aime beaucoup ta façon unique de voir les choses. Merci d'être toujours là pour moi.

Mon passage au club IEC a été un chapitre mémorable de ma vie, où j'ai eu l'honneur de rencontrer des personnes extraordinaires telles que Souad, Hynd, Thafat, Anya et Lynda. Leur présence bienveillante a favorisé mon épanouissement et m'a permis de tisser des liens profonds et précieux. J'exprime ma gratitude à mes Rhs, dont les membres ont joué un rôle exceptionnel dans ma croissance personnelle et professionnelle. Enfin, je tiens à remercier tous les membres du club IEC. Chacun à sa manière a créé un environnement stimulant et a laissé une empreinte indélébile sur mon parcours. Je chéris chaque instant passé en leur compagnie et leur suis profondément reconnaissante.

En guise de conclusion, je tiens à mettre à l'honneur à ma précieuse binôme, Sarah, dont la personnalité éblouissante a enrichi mon parcours. Notre partenariat fructueux a tissé l'une des plus belles amitiés de ma vie, symbolisée par ta tendresse et ta sincérité. Cette réalisation est le fruit de notre travail commun, portant ton empreinte de joie, soutien et force. Tu es bien plus qu'une binôme, tu es une amie inestimable.

Yousra

Dédicaces Spéciales DAÏTA

C'est avec une immense joie et une profonde émotion que nous dédions ces mots à tous les membres de notre classe, la première promotion exceptionnelle de DSIA à Polytech. Au cours de ces trois dernières années, nous avons tissé des liens indélébiles, partageant des moments inoubliables, des rires, des réussites et des défis. Cette dédicace célèbre plus qu'une simple promotion, elle honore une famille choisie, riche de sa diversité et de ses talents, une véritable deuxième maison pour nous tous.

À Lina, source inépuisable d'humour, tes blagues et tes fameuses demandes pour des cours en ligne chaque fois que la pluie menaçait ne manquaient jamais de nous faire rire. Les moments de stress étaient plus légers grâce à tes mémos qui nous faisaient rire à gorges déployées.

À Insaf, notre "Kastamar" ou "Rojo", l'une des oreilles les plus attentives, tu as toujours su être sérieuse quand il le fallait et drôle au moment opportun, faisant preuve d'une grande sagesse en toutes circonstances.

À Nawel, toujours serviable, toujours prête à donner un coup de main. Ton grand cœur a fait de notre classe un lieu plus chaleureux.

À Nour, notre major de classe. Ton calme et ton intelligence nous ont souvent impressionnés, et qui pourrait oublier les révisions que tu nous prodiguais sur le tableau ? Tu étais notre enseignante "incognito".

À Amel, notre chanteuse talentueuse, ton énergie positive et ton esprit de vivre ont toujours apporté une note joyeuse à nos journées.

À Amira, si calme et douce, ta présence apaisante a souvent été un baume sur nos journées.

À Chaimaa, l'english woman de la classe, ta maîtrise de la langue de Shakespeare nous a toujours amusés et impressionnés.

À Anes, ton cri de guerre "Metaliiiiii !" a souvent brisé la monotonie de nos journées. Même dans la colère et le mécontentement, tu nous faisais rire.

À Smail, notre autre major et "meilleur Doudou au monde". Ton sens de l'humour et ta serviabilité étaient toujours là pour nous réconforter. Nous nous souviendrons toujours de tes menaces hilarantes avec de la poudre à canon.

À Somnef, ton talent et tes compétences nous ont souvent laissés stupéfaits. Tu étais une fierté pour nous tous.

À Rayane, le grand Kimo, toujours prêt à éclater de rire depuis le lycée, surtout lors des contacts visuels (les yeux dans les yeux). Ta générosité et ta volonté constante d'aider ont fait de toi une source de joie et de soutien pour nous tous.

À Stiffou, tu as su nous baptiser en tant que les 'DAÏTA'. Tes réactions et ton fameux "l'Coooooo" nous ont souvent fait rire aux éclats.

À Sofiane, notre sentimental oriental, malgré ton calme apparent, tu as toujours su nous surprendre avec ton énergie et tes réactions drôles et inattendues.

À Ramy, malgré ton apparence sérieuse et ton air souvent énervé, tu caches un cœur d'or. Ton talent et tes capacités exceptionnelles ne sont égalés que par ton dévouement et ton travail acharné.

À Anis, toujours avec un sourire aux lèvres et un rire dans la voix. Même si tu n'as pas continué avec nous pour cette dernière année, ton esprit joyeux et ton humour contagieux nous ont accompagnés tout au long.

À Mounaim, toujours sage et motivé. Ton rôle de maître des sondages dans le groupe pour régler les choses ne sera jamais oublié.

À Hachem, même si nous te voyions rarement, ta présence était toujours ressentie. Tu es une véritable force, dotée d'une intelligence remarquable.

Et enfin, à Mondhir, toujours en retard avec une tasse de café à la main, ton pc et ton chargeur. Ta personnalité unique et ton humour sec ne manquaient jamais de nous amuser.

Tout ce qu'on peut dire pour conclure, c'est merci. Merci pour ces trois merveilleuses années. Vous avez tous été exceptionnels à votre manière, chacun a apporté une couleur unique à notre palette collective. Voici à notre avenir brillant et à l'amitié qui nous unit pour toujours !

Sarah & Yousra

Remerciements

Nous tenons à exprimer notre profonde gratitude et nos sincères remerciements à tous ceux qui ont contribué à la réalisation de ce mémoire. Votre soutien inestimable, vos conseils précieux et votre encouragement constant ont été essentiels pour mener à bien ce projet.

Tout d'abord, nous souhaitons remercier notre promoteur, Mr ARKI, pour sa guidance experte et ses précieux conseils tout au long de notre travail. Votre expertise et votre implication ont été d'une importance capitale pour nous guider dans notre réflexion et nous aider à mener à bien nos recherches.

Nous adressons également nos remerciements à notre promoteur, Mr BOUAZIZ, pour son soutien et sa contribution significative à notre mémoire. Votre expertise et votre expérience dans le domaine ont enrichi notre travail et lui ont donné une dimension concrète et appliquée.

Nous tenons à exprimer notre reconnaissance envers l'ensemble de l'équipe pédagogique du département du Génie Industriel, et en particulier Messieurs BOUKABOUS, HAMRI, BOUBAKEUR, FOURAR et ZOUAGHI, ainsi que Mesdames BOUCHAFAA et NAHILI. Vos enseignements, vos conseils et votre encadrement ont été essentiels dans notre parcours académique et ont joué un rôle déterminant dans la réalisation de ce mémoire.

Nous souhaitons également remercier chaleureusement l'équipe BRINIAC, et tout particulièrement Hind et Mounaim, pour avoir constamment été à notre écoute, nous avoir généreusement offert leur aide et partagé leurs connaissances. Votre collaboration et votre soutien ont été d'une valeur inestimable et ont contribué à l'enrichissement de notre travail.

Enfin, nous tenons à exprimer notre gratitude envers Yamina, notre alumni, dont la contribution a été précieuse. Tes conseils et ta disponibilité ont été d'une grande aide dans notre démarche de recherche et nous sommes profondément reconnaissantes.

Merci à tous !

ملخص

الهدف من هذا العمل هو استكشاف استخدام الذكاء الاصطناعي للتوليد التلقائي للنصوص في الوصف التقني للمشاريع. ركزنا على شركة MB inc **وقمنا بتطوير نموذج ذكاء اصطناعي مبتكر يستخدم التعلم العميق ومعالجة اللغة الطبيعية. على الرغم من التحديات مثل الإفراط في التعلم وضمان جودة النصوص المنشأة ، أظهر نظامنا نتائج واعدة في تحسين الإنتاجية والاتصال التقني.**

الكلمات المفتاحية : الذكاء الاصطناعي، التوليد التلقائي للنصوص، الوصف التقني للمشاريع، التعلم العميق، معالجة اللغة الطبيعية، التعلم بالنقل، التقارير التقنية، MB inc.

Abstract

The objective of this work is to explore the use of Artificial Intelligence for the automatic generation of texts in the technical description of projects. We focused on the MB inc organization and developed an innovative AI model that uses deep learning and natural language processing. Despite challenges such as overfitting and ensuring the quality of generated texts, our system has shown promising results in improving productivity and technical communication.

Keywords : Artificial Intelligence (AI), Automatic text generation, Technical description of projects, Deep Learning, Natural Language Processing (NLP), Transfer Learning, Technical reports, MB inc.

Résumé

L'objectif de ce travail est d'explorer l'utilisation de l'intelligence artificielle pour la génération automatique de textes dans la description technique des projets. Nous avons mis l'accent sur l'organisme MB inc et développé un modèle d'IA innovant qui utilise l'apprentissage profond et le traitement du langage naturel. Malgré les défis, tels que l'overfitting et la garantie de la qualité des textes générés, notre système a montré des résultats prometteurs dans l'amélioration de la productivité et de la communication technique.

Mots-Clés : Intelligence Artificielle (IA), Génération automatique de textes, Description technique des projets, Apprentissage Profond, Traitement Automatique du Langage Naturel (TALN), Apprentissage par Transfert, Rapports techniques, MB inc.

Table des matières

Liste des tableaux

Table des figures

Liste des abréviations

2D : 2 Dimensions
3D : 3 Dimensions
AF : Année fiscale
AI : Artificial Intelligence
API : Application Programming Interface
ARC : Accounts Receivable Conversion
AR : Augmented Reality
AWS : Amazon Web Services
BART : Bidirectional and Auto-Regressive Transformers
BERT : Bidirectional Encoder Representations from Transformers
BNC : Bayonet Neill-Concelman
BOW : Bag Of Words
BPE : Byte-Pair Encoding
CAH : Classification Ascendante Hiérarchique
CBOW : Continuous Bag Of Words
CDAE : Développement Des Affaires Electroniques
CMS : Content Management System
CNN : Convolutional Neural Network
CNRC PARI : Conseil National de Recherches Canada Programme d'Aide à la Recherche Industrielle
CPU : Central Processing Unit
CTMM : Production de Titres MultiMédia
D3 : Data-Driven Documents
DL : Deep Learning
Doc2Vec : Document to Vector
DOS : Disk Operating System
DS : Data Science
EC2 : Elastic Compute Cloud
FAF : Fin d'année fiscale
Glove : Global Vectors for Word Representation
GP : Génération Procédurale
GPT : Generative Pre-training Transformer
GPU : Graphics Processing Unit
GRU : Gated Recurrent Unit
HTTP : Hypertext Transfer Protocol
IA : Intelligence Artificielle
IBM : International Business Machines

IQ : Intelligence Quotient
JSON : JavaScript Object Notation
LM : Language Model
LSTM : Long Short-Term Memory
MATLAB : Matrix Laboratory
MIT : Massachusetts Institute of Technology
ML : Machine Learning
MRQ : Ministère du Revenu du Québec
NER : Named Entity Recognition
NLTK : Natural Language Toolkit
NLP : Natural Language Processing
NPC : Non playable caracters
NumPy : Numerical Python
PARI : Programme d'Aide à la Recherche Industrielle
PFE : Projet de Fin d'Etudes
PLM : Pre-trained Language Model
PPL : Perplexity
POS : Part Of Speech
POST : Post Office Protocol
RBC : Royal Bank of Canada
RD : Research and Development
Regex : Regular Expression
RL : Reinforcement Learning
RNN : Recurrent Neural Network
RS&DE : Recherche Scientifique et Développement Expérimental
SARSA : State-Action-Reward-State-Action
SCAND : Scan and Deliver
Sklearn : Scikit-learn
SSL : Secure Sockets Layer
T5 : Text-to-Text Transfer Transformer
TALN : Traitement Automatique du Langage Naturel
TF-IDF : Term Frequency-Inverse Document Frequency
URL : Uniform Resource Locator
VR : Virtual Reality
Word2Vec : Word to Vector
XLNET : eXtreme Language Understanding

Introduction générale

L'intelligence artificielle (IA) et son potentiel disruptif ont fait l'objet d'un nombre croissant d'études et de discussions ces dernières années. Le domaine de la génération de textes ne fait pas exception à cette tendance, avec l'apparition de technologies qui permettent de créer des textes automatiquement et en grande quantité. Ces technologies ont des implications considérables pour une multitude de secteurs, notamment celui des rapports techniques, qui génèrent un volume énorme de données non structurées. Ce mémoire vise à développer un modèle d'IA innovant pour la génération de textes pour la description technique des projets, en ciblant trois aspects essentiels : le contexte de l'étude, la problématique et les objectifs. Nous établirons aussi la structure du rapport pour la réalisation de notre projet de fin d'études (PFE).

Contexte

Les progrès récents en IA ont apporté une solution pour faciliter la structuration et l'organisation de données non structurées dans les rapports techniques. Ces rapports sont essentiels dans divers secteurs, mais leur volume, leur nombre et leur complexité croissants ont rendu leur traitement de plus en plus ardu. Notre travail se concentre sur la création d'un système de génération de textes basé sur l'IA qui peut produire des descriptions techniques précises, concises et pertinentes de projets. Ce système se base sur l'efficacité de la prise de notes et de la structuration des données pour améliorer la productivité et la communication technique dans diverses industries.

Problématique

La génération automatique de textes pour la description technique des projets se heurte à plusieurs problèmes significatifs. Les défis à relever comprennent le volume conséquent et la complexité des rapports techniques, qui rendent leur interprétation coûteuse et chronophage. Ces rapports sont souvent décentralisés, stockés dans différents dossiers de Dropbox ou d'autres services de stockage cloud, compliquant leur gestion et leur exploitation. La barrière linguistique pose également un problème, les rapports étant généralement rédigés en anglais et en français, nécessitant un modèle capable de traiter efficacement ces deux langues. En outre, le risque d'overfitting, un phénomène où le modèle apprend trop bien les données d'apprentissage au détriment de nouvelles données, présente un défi majeur. Enfin, la garantie de la qualité et de la pertinence des textes générés est une préoccupation primordiale, puisque les modèles peuvent parfois produire des textes grammaticalement corrects mais manquant de sens ou de pertinence contextuelle.

Objectifs

Notre travail vise à répondre à ces problématiques en atteignant les objectifs suivants :

- Développer un modèle d'intelligence artificielle performant pour la génération de textes techniques.
- Structurer et organiser les données de manière efficace grâce à un système de prise de notes adapté.

- Améliorer l'efficacité et la qualité de la communication technique dans les projets en utilisant une technologie avancée.
- Réduire les coûts et les délais d'interprétation des documents textuels grâce à la génération automatique de textes précis.

Organisation du mémoire

Notre mémoire s'articulera autour de trois grandes sections, chacune consacrée à un aspect fondamental de notre recherche et présentant une synthèse approfondie du travail effectué. Chaque section vise à répondre à notre problématique et à atteindre les objectifs fixés.

Première Partie : État de l'art

Cette première section est dédiée à l'examen approfondi des concepts clés qui sous-tendent notre étude.

- **Le premier chapitre** présente les rapports techniques, une entité centrale de notre recherche. Il explorera leur domaine d'application, leur définition, leurs types et les conventions de rédaction généralement adoptées.
- **Le deuxième chapitre** se penche sur le traitement automatique des données textuelles. Nous y abordons l'apprentissage automatique, en soulignant ses limites pour résoudre notre problématique, avant d'explorer l'apprentissage profond et les variétés de problèmes qu'il aborde. Un accent particulier est mis sur le traitement automatique du langage naturel (NLP), une sous-discipline cruciale pour notre étude. Nous concluons le chapitre en introduisant la notion d'apprentissage par transfert, une méthode prometteuse pour notre objectif.
- **Le troisième chapitre** est consacré à la génération automatique des textes. Nous y définissons ce concept, examinons les différentes approches existantes et décrivons en détail le processus de génération. L'utilisation des modèles de transformation dans le cadre de l'apprentissage par transfert, notre approche choisie, y est particulièrement examinée.

Deuxième Partie : État des lieux

Cette section donne un aperçu de l'environnement dans lequel notre étude est menée.

- **Le quatrième chapitre** présente brièvement l'organisme d'accueil, "MB inc".
- **Le cinquième chapitre**, quant à lui, décrit le système global existant et l'état actuel de son fonctionnement.

Troisième Partie : Conception et réalisation de la solution

Cette section finale détaille la conception et la mise en œuvre de notre solution.

- **Le sixième chapitre** détaille la conception de la solution proposée pour la génération de texte basée sur l'apprentissage par transfert. Il justifie le choix de cette approche et en décrit les étapes.

- **Le septième et dernier chapitre** offre une description exhaustive de l'architecture technique du système réalisé, énumérant les outils et technologies utilisés. Il trace en détail toutes les étapes de mise en place de notre solution, de la création de la base de données à la modélisation, jusqu'au déploiement de la solution.

Cet agencement vise à fournir une lecture fluide et logique, en guidant le lecteur à travers la complexité de notre recherche et de nos solutions.

Première partie
État de l'art

Chapitre 1

Les rapports techniques

1.1 Introduction

Afin de mieux comprendre le contexte de notre étude, de répondre à la problématique posée et d'atteindre nos objectifs, ce premier chapitre vise à présenter la RS&DE en décrivant son objectif et les différents travaux admissibles au titre de ce domaine. Ensuite, nous allons explorer les rapports techniques, en expliquant leur nature et en donnant un aperçu de leurs différents types. Enfin, nous conclurons ce chapitre en abordant les meilleures pratiques pour la rédaction de ces rapports techniques.

1.2 La RS&DE

1.2.1 Qu'est ce que la RS&DE ?

La RS&DE est une activité visant à acquérir ou améliorer les connaissances scientifiques et technologiques par une approche expérimentale et systématique. Elle peut être menée dans des laboratoires universitaires, des centres de recherche publics ou privés, ou dans les entreprises elles-mêmes. Au Canada, elle est encouragée par un régime fiscal avantageux pour les entreprises, à condition de répondre à certaines conditions, notamment d'être effectuée de manière systématique et d'avoir pour objectif de faire avancer les connaissances dans un domaine donné. [8]

1.2.2 Le but de la RS&DE

La RS&DE a pour objectif l'avancement des connaissances scientifiques et technologiques en utilisant une approche expérimentale et systématique. Elle consiste à acquérir ou améliorer les connaissances dans différents domaines, que ce soit par le biais de la recherche pure ou appliquée, ou encore par le développement expérimental, avec pour finalité des avancées technologiques significatives. Les entreprises qui mènent ces activités peuvent bénéficier d'un régime fiscal avantageux, à condition de respecter certaines conditions.

1.2.3 Les travaux admissibles au titre de la RS&DE

Les travaux admissibles au titre de la RS&DE peuvent être de nature diverses et doivent répondre à certains critères pour être éligible. Ils peuvent inclure des travaux de **recherche pure**,

qui visent à faire progresser les connaissances scientifiques dans un domaine sans avoir d'application pratique immédiate en vue. Les travaux de **recherche appliquée**, quant à eux, sont orientés vers des applications pratiques, avec pour objectif de résoudre des problèmes concrets ou de développer de nouvelles technologies. Enfin, les travaux de **développement expérimental** ont pour but d'améliorer ou de créer de nouveaux produits, dispositifs ou procédés en utilisant une approche expérimentale et systématique, avec pour finalité le progrès technologique. Quel que soit le type de travaux, ils doivent être menés dans le cadre d'une démarche scientifique rigoureuse et systématique, avec la documentation et la traçabilité adéquate pour prouver leur éligibilité au crédit d'impôt pour RS&DE.

1.2.4 RS&DE - Aperçu technique

Les projets de recherche et développement expérimentent souvent des incertitudes technologiques, empêchant les équipes de prévoir précisément les résultats finaux ou la méthode pour les atteindre. De même, la limite de la base de connaissances peut se présenter comme un obstacle pour les chercheurs et les ingénieurs. Pour surmonter ces défis, un personnel compétent et une investigation systématique sont nécessaires. Les travaux effectués doivent inclure la formulation d'hypothèses, la vérification par des expérimentations, des analyses et des conclusions, ainsi que la documentation des résultats d'efforts. Pour mener à bien ces projets, une variété d'outils peut être utilisée, tels que la conception, la recherche opérationnelle, l'analyse mathématique, la recherche psychologique, les essais, la collecte de données, la programmation informatique, etc. Les résultats obtenus peuvent être utilisés pour une variété de fins, telles que l'étude du marché ou la promotion des ventes, le contrôle de la qualité ou la mise à l'essai normale des matériaux, des dispositifs, des produits ou des procédés, la recherche en sciences sociales ou en sciences humaines, la prospection, l'exploration et le forage pour la découverte de minéraux, de pétrole ou de gaz naturel et leur production, la production commerciale d'un matériau, d'un dispositif ou d'un produit nouveau ou amélioré, et l'utilisation commerciale d'un procédé nouveau ou amélioré. Les avancements technologiques sont souvent le résultat de la compréhension de relations scientifiques, repoussant l'état des connaissances pour créer ou améliorer des produits ou des procédés existants. Les travaux effectués nécessitent une grande créativité pour structurer et raconter une histoire qui met en valeur les progrès technologiques réalisés.

1.3 Les rapports techniques

1.3.1 Définition

Un rapport technique est un document rédigé dans le but de décrire un projet, une étude ou un travail réalisé dans un domaine technique. Ce rapport fournit une analyse détaillée de la méthodologie employée, des résultats obtenus, des conclusions et des recommandations. Le rapport technique peut être utilisé pour partager des informations avec d'autres professionnels du même domaine ou pour présenter des résultats à un public plus large.

1.3.2 Les différents types de rapports techniques

Les rapports techniques peuvent prendre différentes formes, selon leur objectif et leur contenu. Les principaux types de rapports techniques sont :

- **Les rapports d'analyse :** Ce type de rapport présente une analyse détaillée d'un sujet spécifique, en utilisant des données, des graphiques et des tableaux pour appuyer les conclusions. Les rapports d'analyse sont souvent utilisés dans les domaines scientifiques et techniques pour évaluer la performance d'un système ou d'un processus.
- **Les rapports de recherche :** Ce type de rapport présente les résultats d'une étude scientifique ou d'une enquête de recherche. Les rapports de recherche peuvent inclure une analyse des données, des résultats statistiques, des conclusions et des recommandations pour des futures études.
- **Les rapports d'expertise :** Ce type de rapport est rédigé par un expert pour fournir une évaluation professionnelle d'un sujet spécifique. Les rapports d'expertise peuvent être utilisés dans des domaines tels que la médecine, la finance ou la technologie pour fournir des conseils ou des opinions d'experts.
- **Les rapports de diagnostic :** Ce type de rapport est utilisé pour identifier et résoudre les problèmes techniques. Les rapports de diagnostic peuvent inclure des recommandations pour des solutions et des stratégies de mise en œuvre pour résoudre les problèmes identifiés.

1.4 La rédaction des rapports techniques

1.4.1 Les compétences nécessaires pour rédiger un rapport technique

Les compétences nécessaires pour rédiger un bon rapport technique comprennent :

- **Une capacité rédactionnelle :** Un bon rapport technique doit être clair, concis et bien structuré. Il est important de savoir comment communiquer des idées complexes de manière simple et facilement compréhensible.
- **Une solide compréhension de plusieurs domaines de la technologie et de l'ingénierie :** Il est essentiel d'avoir une connaissance approfondie du domaine technique ou de l'ingénierie pour lequel le rapport est rédigé. Cela inclut une compréhension des concepts clés, des normes et des procédures.

1.4.2 Les outils et les technologies pour la rédaction des rapports techniques

Pour faciliter le processus de rédaction et garantir des résultats de qualité, divers outils et technologies sont disponibles. Ces outils offrent des fonctionnalités spécifiques pour la création, la structuration et la présentation des rapports techniques.

- **Les logiciels de traitement du texte :** Les logiciels de traitement de texte tels que Microsoft Word, Google Docs et LibreOffice Writer sont des outils essentiels pour la rédaction de rapports techniques. Ils permettent de créer et de formater facilement le contenu du rapport, d'organiser les sections, de gérer les références bibliographiques et d'insérer des images ou des tableaux. Ces logiciels offrent également des fonctionnalités de vérification orthographique et grammaticale pour garantir la qualité du texte.
- **Les logiciels de présentation :** Les logiciels de présentation, comme Microsoft PowerPoint, Google Slides et Apple Keynote, sont utiles pour créer des diapositives et des présentations visuelles dans les rapports techniques. Ils permettent de présenter les informations de manière

claire et concise à l'aide de graphiques, de schémas, de diagrammes et d'autres éléments visuels. Les logiciels de présentation offrent également des fonctionnalités d'animation et de transition pour rendre les présentations plus dynamiques et attrayantes.

- **Les logiciels de modélisation :** Les logiciels de modélisation sont utilisés pour représenter graphiquement des concepts, des processus ou des systèmes complexes. Dans les rapports techniques, ces outils sont particulièrement utiles pour visualiser des données, créer des diagrammes, des schémas ou des représentations en 3D. Des logiciels de modélisation tels que AutoCAD, MATLAB, SketchUp, ou SolidWorks sont couramment utilisés dans différents domaines techniques.
- **Les logiciels de gestion de projet :** La rédaction de rapports techniques peut souvent impliquer une collaboration entre plusieurs personnes et nécessiter une gestion efficace du projet. Les logiciels de gestion de projet, tels que Trello, Asana, Jira ou Microsoft Project, aident à organiser les tâches, à attribuer des responsabilités, à suivre les délais et à faciliter la communication au sein de l'équipe de rédaction du rapport.
- **Les logiciels de traitement des données :** Dans certains rapports techniques, il est nécessaire de collecter, analyser et présenter des données. Les logiciels de traitement de données, tels que Microsoft Excel, Google Sheets, R ou Python, sont utilisés pour manipuler et analyser des données, créer des graphiques, des tableaux croisés dynamiques et des représentations visuelles. Ces outils permettent de présenter les résultats de manière claire et compréhensible.

En utilisant ces différents outils et technologies, les rédacteurs de rapports techniques peuvent améliorer leur efficacité, la qualité de leurs documents et leur capacité à communiquer efficacement les informations techniques. Chaque outil offre des fonctionnalités spécifiques qui peuvent être adaptées en fonction des besoins et des exigences du rapport technique en cours de rédaction.

1.4.3 Les enjeux et les défis de la rédaction de rapports techniques

La rédaction de rapports techniques présente certains enjeux et défis qui doivent être pris en compte pour produire des documents de qualité. Ces défis sont liés à la complexité des sujets techniques, à la diversité des publics cibles et aux enjeux liés à la qualité de la communication. Examinons ces points plus en détail :

- **Les défis liés à la complexité des sujets techniques :**
  Les rapports techniques traitent souvent de sujets complexes et spécialisés, tels que l'ingénierie, la science, l'informatique ou d'autres domaines techniques. La principale difficulté réside dans la capacité à traduire des concepts techniques complexes en un langage clair et accessible pour les lecteurs non spécialisés. Il est essentiel de simplifier les informations techniques tout en maintenant leur exactitude et leur précision, afin de faciliter la compréhension du rapport par un large public.
- **Les défis liés à la diversité des publics cibles :**
  Les rapports techniques peuvent être destinés à des publics variés, allant des experts techniques aux décideurs non spécialisés. Chaque public a des besoins, des connaissances et des attentes différents. Il est donc important de prendre en compte cette diversité et d'adapter le contenu, le ton et le niveau de détail du rapport en fonction du public cible. Cela peut nécessiter l'utilisation de termes techniques précis pour les experts, ainsi que des explications et des exemples plus accessibles pour les non-spécialistes.

- **Les enjeux liés à la qualité de la communication :**
  La qualité de la communication est un enjeu crucial dans la rédaction de rapports techniques. Il est essentiel de transmettre clairement les informations, les résultats et les analyses de manière cohérente et compréhensible. Cela implique d'adopter une structure logique, d'utiliser un langage précis et concis, d'éviter les jargons techniques excessifs et d'expliquer les termes spécialisés lorsque cela est nécessaire. La qualité de la communication est également liée à la présentation visuelle des informations, en utilisant des graphiques, des tableaux et des schémas pour renforcer la compréhension.

D'autres défis et enjeux peuvent également se présenter, tels que la gestion du temps, la collecte et l'analyse des données, ainsi que la cohérence et la fiabilité des sources d'information. Pour relever ces défis, il est recommandé de travailler en étroite collaboration avec des experts du domaine, de faire preuve de rigueur dans la recherche et l'analyse des informations, et de procéder à des révisions et des relectures attentives pour garantir la qualité et l'exactitude du rapport technique final.

1.4.4 Les bonnes pratiques pour la rédaction de rapports techniques

La rédaction de rapports techniques efficaces nécessite l'application de bonnes pratiques pour garantir la clarté, la précision et la compréhension du document.

Les conseils pour la rédaction d'un rapport technique efficace :

- Définir clairement les objectifs du rapport et l'audience cible.
- Adoption d'une structure logique avec une introduction, des sections clairement définies et une conclusion.
- Utilisation d'un langage clair, précis et concis en évitant les jargons techniques excessifs.
- Explication des termes spécialisés et fournir des définitions si nécessaire.
- Utilisation des exemples, des illustrations ou des cas concrets pour clarifier les concepts techniques.
- Utilisation des phrases courtes et bien structurées pour faciliter la lecture et la compréhension.
- S'assurer de la cohérence et de la précision des informations tout au long du rapport.
- Faire des révisions et des relectures attentives pour éliminer les erreurs grammaticales et les fautes de frappe.

Les conseils pour la présentation visuelle des données :

- Utilisation des graphiques, des tableaux, des diagrammes ou des schémas pour présenter visuellement les données.
- Choix du type de visualisation approprié en fonction de la nature des données et de l'objectif de la présentation.
- S'assurer que les visuels sont clairs, lisibles et bien étiquetés.
- Utilisation des couleurs appropriées pour améliorer la compréhension et la distinction des données.
- Éviter les visuels encombrés ou confus qui pourraient rendre la lecture difficile.

- Utilisation des titres, des légendes et des annotations pour expliquer les visuels et fournir des informations supplémentaires.

Les conseils pour la gestion de la communication avec le public cible :

- Identifier clairement le public cible et adapter le contenu et le niveau de détail en conséquence.
- Détermination des connaissances préalables du public cible et fournir des explications et des clarifications si nécessaire.
- Utilisation d'un ton approprié et engageant pour maintenir l'intérêt du public.
- Être attentif aux réactions et aux retours du public et répondre aux questions ou aux préoccupations.
- Présentation des informations de manière organisée et logique pour faciliter la compréhension.
- Anticipation des questions potentielles du public et fournir des réponses ou des explications claires.
- Utilisation des exemples ou des illustrations pour rendre les concepts techniques plus concrets et accessibles.

L'application de ces bonnes pratiques contribuera à la rédaction de rapports techniques plus efficaces, améliorant ainsi la qualité de la communication et la compréhension des informations présentées.

1.5 Conclusion

En conclusion, ce chapitre sur les rapports techniques a couvert plusieurs aspects importants liés à la recherche scientifique et au développement expérimental (RS&DE) ainsi qu'à la rédaction de rapports techniques.

La rédaction de rapports techniques joue un rôle essentiel dans la transmission des résultats de la RS&DE et la communication efficace des informations techniques. En appliquant les compétences nécessaires, en utilisant les outils appropriés, en faisant face aux défis et en suivant les bonnes pratiques, les professionnels de la RS&DE peuvent produire des rapports techniques de qualité, favorisant ainsi la diffusion des connaissances et le progrès scientifique.

Chapitre 2

Le traitement automatique des données textuelles

2.1 Introduction

Le traitement automatique des données textuelles est un domaine en pleine expansion qui fait appel à des techniques d'intelligence artificielle pour analyser et extraire des informations à partir de des documents textuels. Ce chapitre explore différentes approches et méthodes utilisées dans ce domaine. Nous commencerons par examiner l'apprentissage automatique, en nous intéressant aux types d'apprentissage et à ses limites. Ensuite, nous nous concentrerons sur l'apprentissage approfondi, en soulignant son importance et les types de problèmes auxquels il peut être appliqué. Nous aborderons également un aspect spécifique de l'apprentissage approfondi, le traitement automatique du langage naturel, en explorant son sens, son lien avec l'intelligence artificielle et l'apprentissage approfondi, ainsi que son évolution historique. Enfin, nous aborderons l'apprentissage par transfert, en définissant cette approche, en expliquant son principe, son utilité et les différentes stratégies qui lui sont associées.

2.2 L'apprentissage automatique (Machine Learning)

2.2.1 Définition

L'apprentissage automatique, également appelé "Machine Learning" en anglais, est un domaine de l'informatique qui permet aux ordinateurs d'apprendre à partir de données sans être explicitement programmés. En utilisant des algorithmes et des modèles mathématiques, l'apprentissage automatique permet aux ordinateurs de reconnaître des modèles et de prendre des décisions autonomes en se basant sur ces modèles. Les applications de l'apprentissage automatique sont nombreuses, allant de la reconnaissance de formes à la prédiction de résultats, en passant par la recommandation de produits et la détection de fraudes.

Où se situe la Data Science ?

Elle consiste à découvrir et à communiquer des informations à partir des données. L'apprentissage automatique est souvent un outil important pour les travaux de la Data Science, en particulier pour faire des prédictions à partir des données. Le schéma suivant illustre la relation entre le ML et la DS :

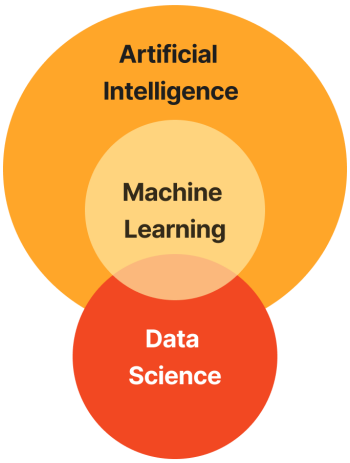


FIGURE 2.1 – La relation entre le ML et la DS.

2.2.2 Les types d'apprentissage automatique

Il existe essentiellement quatre types d'apprentissage automatique, chacun ayant son propre principe, type de données, méthodes, modèles et exemples d'application :

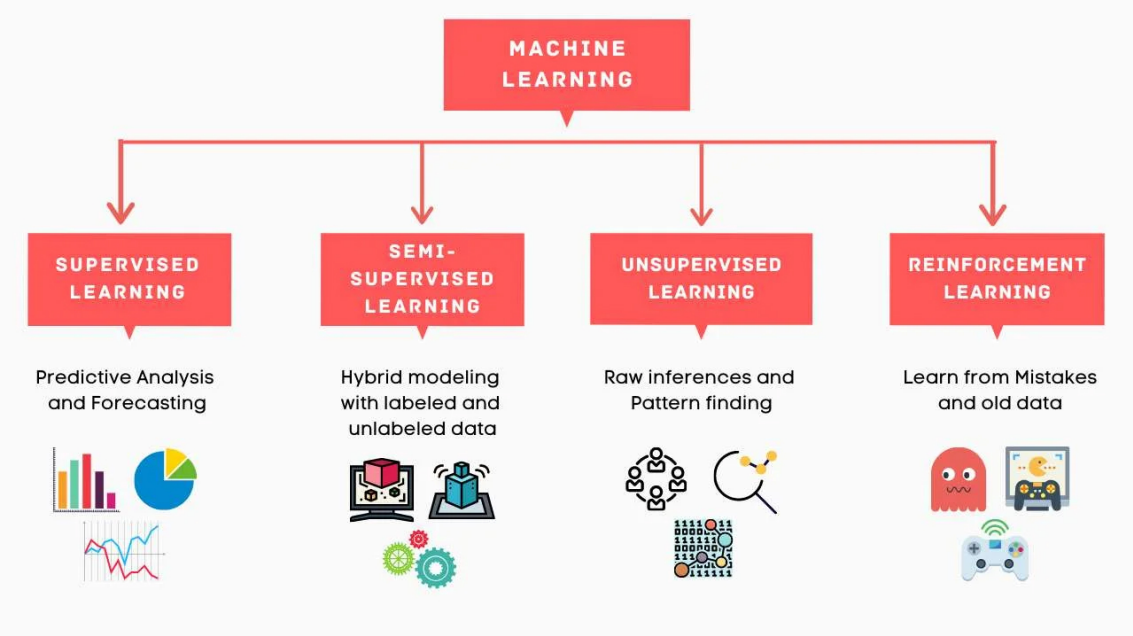


FIGURE 2.2 – Les types d'apprentissage automatique.

Voici une description détaillée de chaque type :

Type ML	Supervised Learning	Unsupervised Learning	Semi-supervised Learning	Reinforcement Learning
Principe	Le modèle est entraîné sur un ensemble de données annotées, où chaque instance de données est associée à une étiquette ou une sortie connue.	Le modèle est entraîné sur un ensemble de données non annotées, sans étiquettes connues.	Le modèle est entraîné sur un ensemble de données contenant à la fois des données annotées et non annotées.	Le modèle apprend à prendre des décisions en interagissant avec un environnement.
But	Prédire la sortie pour les nouvelles instances de données.	Découvrir des structures intéressantes dans les données, telles que des clusters, des motifs ou des associations.	Utiliser les données non annotées pour améliorer la précision des prédictions pour les données annotées.	Maximiser une récompense ou une fonction de performance.
Données utilisées	De type numérique, catégorique ou textuel.	De type numérique, catégorique ou textuel.	De type numérique, catégorique ou textuel.	De type numérique, catégorique ou textuel.
Méthodes courantes	Les régressions linéaires et logistiques, les arbres de décision, les forêts aléatoires, les machines à vecteurs de support (SVM) et les réseaux de neurones.	La classification ascendante hiérarchique (CAH), la réduction de dimensionnalité, la détection d'anomalies et les algorithmes d'association.	Les modèles probabilistes et les réseaux de neurones.	Les algorithmes Q-learning, SARSA et Monte Carlo.
Exemples d'application	La reconnaissance de la parole, la classification d'images et la prédiction des prix de l'immobilier.	La segmentation de marché, la détection de fraude et l'analyse de sentiments.	La classification d'images, la détection de spams et la recommandation de produits.	La robotique, les jeux vidéo et la gestion de portefeuille financier.

TABLE 2.1 – Description détaillée des types d'apprentissages automatique.

2.2.3 Les limites de l'apprentissage automatique

Bien que l'apprentissage automatique soit une technique puissante pour extraire des connaissances à partir de données, il a également certaines limites qu'il est important de prendre en compte :

- **Dépendance aux données :** L'apprentissage automatique dépend fortement de la qualité et de la quantité des données d'entraînement. Si les données sont mal étiquetées ou non représentatives, cela peut entraîner des erreurs dans les prédictions.
- **Surajustement :** Lorsque le modèle est trop complexe par rapport aux données d'entraînement, il peut surapprendre et ne pas généraliser bien pour les données de test. Cela peut également entraîner une mauvaise performance pour de nouvelles données.
- **Explicabilité :** Les modèles d'apprentissage automatique peuvent être très complexes et difficiles à comprendre. Il peut être difficile de savoir comment le modèle prend des décisions et d'expliquer ces décisions aux utilisateurs.
- **Données biaisées :** Les données d'entraînement peuvent être biaisées en raison de facteurs tels que la sélection des données, les préjugés humains ou les erreurs de mesure. Cela peut entraîner des prédictions biaisées pour les données de test.
- **Manque de diversité :** Les modèles d'apprentissage automatique peuvent manquer de diversité dans les types de données qu'ils peuvent traiter. Par exemple, les modèles d'apprentissage automatique peuvent avoir des difficultés à traiter des données non structurées telles que les images, les sons et les textes.
- **Coût de calcul :** Les algorithmes d'apprentissage automatique peuvent nécessiter une puissance de calcul élevée et des ressources de stockage importantes pour traiter de grandes quantités de données. Cela peut être coûteux et prendre beaucoup de temps pour entraîner et mettre en œuvre les modèles.

Il est important de prendre en compte ces limites lors de l'utilisation de l'apprentissage automatique pour garantir des résultats fiables et éviter des erreurs coûteuses.

2.3 L'apprentissage profond (Deep Learning)

2.3.1 Définition

Le Deep Learning, ou l'apprentissage profond, est un type d'apprentissage machine, qui est lui-même un sous-domaine de l'intelligence artificielle. Il utilise des réseaux de neurones artificiels inspirés du cerveau humain pour apprendre, prévoir et décider de manière autonome à partir de grandes quantités de données.

Le terme "Deep" fait référence à la profondeur de ces réseaux, qui sont constitués de plusieurs couches de neurones interconnectés pour extraire des caractéristiques et des niveaux d'abstraction des données. Grâce à cette architecture complexe, le deep learning est capable d'effectuer des tâches de reconnaissance de formes, de traitement de la parole, de la vision et du langage naturel, en apprenant de manière autonome à partir de grandes quantités de données. [9]

2.3.2 Pourquoi le DL ?

Le Deep Learning est une technique puissante pour résoudre des problèmes complexes dans divers domaines tels que la reconnaissance de formes, le traitement de la parole, de la vision et du langage naturel. Il peut être utilisé lorsque les approches classiques d'apprentissage automatique ne sont pas suffisantes pour traiter des données de grande dimensionnalité et de complexité élevée.

Les avantages du Deep Learning sont nombreux : il est capable d'apprendre des représentations de caractéristiques de manière automatique, il peut gérer des données de grande dimensionnalité,

il peut être utilisé pour la classification, la reconnaissance d'images, le traitement de la parole et du langage naturel, et il est capable d'améliorer ses performances à mesure que les données d'entraînement deviennent plus abondantes.

En outre, le Deep Learning est particulièrement bien adapté aux problèmes où les données sont non linéaires et peuvent présenter des relations complexes entre les variables, ce qui en fait un outil privilégié pour la modélisation et la prédiction de phénomènes complexes tels que le changement climatique, la médecine personnalisée et la détection de fraudes.

2.3.3 Les types de problèmes traités par DL

- **Computer vision :**
  La vision par ordinateur (computer vision en anglais) est un domaine de l'intelligence artificielle qui se concentre sur la manière dont les ordinateurs peuvent interpréter et comprendre les images et les vidéos. Plus précisément, la vision par ordinateur vise à permettre aux ordinateurs de "voir" le monde comme le font les humains, en analysant des images et en en extrayant des informations utiles.

  Les tâches de la vision par ordinateur incluent la reconnaissance d'objets, la classification d'images, la détection de visages, la segmentation d'images, la reconnaissance de gestes, la surveillance vidéo et la réalité augmentée, entre autres. Pour accomplir ces tâches, les algorithmes de vision par ordinateur utilisent des techniques de traitement d'images, de reconnaissance de formes et d'apprentissage automatique, notamment le deep learning.

  En utilisant des réseaux de neurones profonds, les ordinateurs peuvent apprendre à détecter et à identifier des objets, des visages, des émotions, des scènes, des mouvements, etc. dans des images et des vidéos. Le deep learning permet également de résoudre des problèmes de segmentation d'image, de reconnaissance de texte et de génération d'images.

- **Natural Language Processing (NLP) :**
  Le Deep Learning, en particulier les réseaux de neurones profonds, est devenu une méthode très populaire pour résoudre les problèmes de NLP en raison de leur capacité à apprendre des représentations de haute qualité à partir de données textuelles complexes. Les réseaux de neurones profonds peuvent être utilisés pour résoudre une variété de tâches de NLP telles que la classification de texte, la génération de texte, la traduction automatique et l'analyse de sentiment.

  Le Deep Learning a permis de grandes avancées dans les domaines du traitement automatique du langage naturel, tels que la reconnaissance automatique de la parole, la compréhension automatique du langage naturel et la réponse automatique. Il a également permis de traiter des données textuelles plus complexes telles que des documents entiers plutôt que de simples phrases ou des mots isolés.

  En résumé, le Deep Learning représente un outil puissant pour résoudre des problèmes de NLP en raison de sa capacité à apprendre des représentations de haute qualité à partir de données textuelles complexes, ce qui permet d'obtenir de meilleurs résultats dans la résolution de tâches de NLP.

2.4 Le traitement automatique du langage naturel (Natural Language Processing - NLP)

2.4.1 Le NLP - C'est quoi ?

Le traitement automatique du langage naturel (TALN) est un sous-domaine de la linguistique, de l'informatique et de l'intelligence artificielle qui se concentre sur les interactions entre les ordinateurs et le langage humain, en particulier sur la façon de programmer les ordinateurs pour comprendre, traiter, produire et analyser de grandes quantités de données de langage naturel utilisé par les êtres humains dans leur communication verbale ou écrite.

Le NLP est un domaine de recherche en constante évolution, avec de nouvelles techniques et de nouveaux modèles informatiques développés régulièrement pour améliorer les capacités de traitement du langage naturel.

2.4.2 La relation entre AI, NLP et DL

L'AI (Intelligence Artificielle) est un domaine de l'informatique qui vise à créer des systèmes capables d'effectuer des tâches qui nécessitent normalement l'intelligence humaine. Le NLP (Traitement du Langage Naturel) est une branche de l'AI qui se concentre sur la compréhension et la production du langage humain par les ordinateurs. Le DL (Apprentissage Profond) est une technique de l'AI qui utilise des réseaux de neurones artificiels pour apprendre à partir de données non structurées, telles que des images et du texte, en trouvant des modèles hiérarchiques de représentation des données.

Ainsi, le DL est utilisé pour entraîner des modèles de NLP afin qu'ils puissent comprendre et produire du langage naturel de manière plus efficace. Les techniques de DL telles que les réseaux de neurones convolutifs (CNN) et les réseaux de neurones récurrents (RNN) ont été appliquées avec succès à des tâches de NLP telles que la classification de texte, la traduction automatique, la génération de texte, etc.

En résumé, le NLP est un domaine de l'AI qui utilise des techniques de DL pour résoudre des problèmes liés à la compréhension et la production du langage naturel.

2.4.3 L'histoire du NLP

Les racines historiques du traitement du langage naturel remontent à :

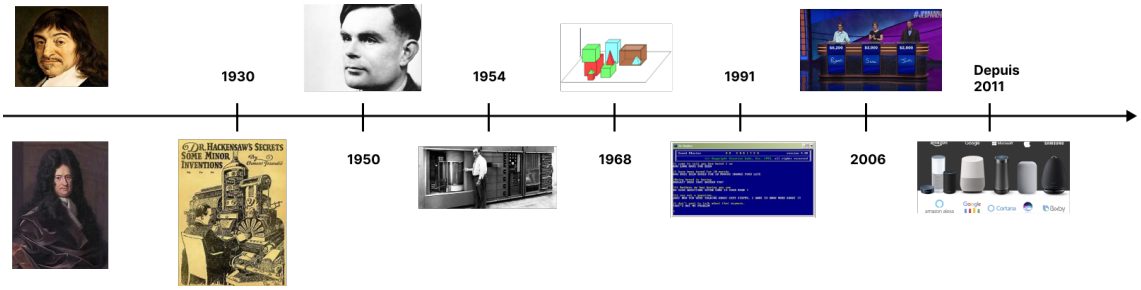


FIGURE 2.3 – Timeline : Histoire du NLP.

- Les philosophes tels que Leibniz et Descartes ont proposé des symboles qui pourraient lier les mots entre les langues. Toutes ces propositions sont restées théoriques, sans qu'aucune machine réelle ne soit construite.
- Depuis les années 30, il y a eu des idées et des conceptions pour une machine de traduction, mais il n'y en avait pas une réellement efficace.
- En 1950, Alan Turing a publié son célèbre article "Computing Machinery and Intelligence" qui a présenté ce qui est maintenant appelé le test de Turing pour servir de norme pour l'intelligence artificielle.

  Ce critère repose sur la capacité du programme informatique à simuler la caractéristique humaine dans une conversation écrite avec un humain en temps réel, de manière suffisamment convaincante pour que l'interlocuteur ne puisse pas faire la distinction de manière fiable, sur la base du contenu de la conversation seulement, entre le programme et un être humain réel.
- "The Georgetown experiment" a été inventé en collaboration avec IBM en 1954 pour traduire automatiquement plus de soixante phrases en russe en anglais. Les auteurs du programme ont prétendu qu'en trois ou cinq ans, le problème de la traduction automatique serait résolu.
- En 1968, un programme appelé SHRDLU a été développé par l'ingénieur américain Terry Winograd au MIT, qui a réussi à établir un dialogue plutôt fluide avec l'ordinateur :

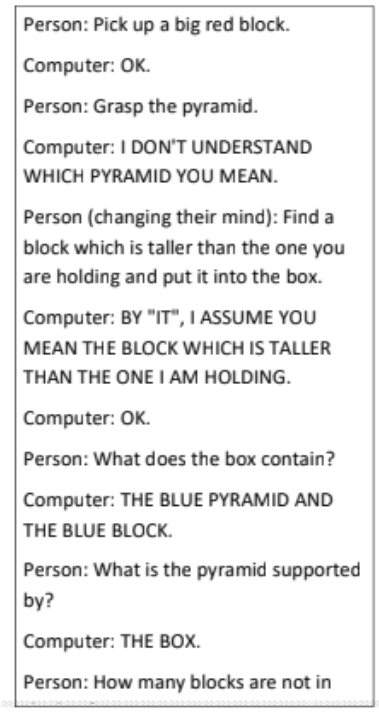


FIGURE 2.4 – Résultat du programme SHRDLU.

- En 1991, un modèle de "processeur psychologique intelligent" a été créé, basé sur l'idée d'un chatbot et fonctionnant sur DOS, commençant par la phrase :

"HELLO [UserName], MY NAME IS DOCTOR SBAITSO.

I AM HERE TO HELP YOU. SAY WHATEVER IS IN YOUR MIND FREELY, OUR CONVERSATION WILL BE KEPT IN STRICT CONFIDENCE. MEMORY

- En 2006, la première version du programme géant Watson d'IBM a été publiée. Il a réussi à surpasser les humains dans la réponse aux questions dans le célèbre programme américain, Jeopardy !
- L'assistant personnel Siri a été lancé en 2011, suivis par Alexa et Cortana en 2014, Google Assistant en 2016 et Samsung Bixby en 2017.

2.4.4 Le Text Mining

Définition

Le text mining, également appelé fouille de textes ou exploration de données textuelles, est une méthode de traitement automatique du langage naturel qui permet d'extraire des informations pertinentes à partir de textes non structurés. Elle permet de trouver des modèles et des tendances dans les données, même lorsque ces données ne sont pas organisées de manière cohérente. Le text mining est utilisé dans une variété de domaines, tels que la recherche scientifique, l'analyse de données, la veille stratégique ou technologique selon des pistes de recherches prédéfinies.

Le text mining est un domaine interdisciplinaire qui combine des connaissances en linguistique, informatique, mathématiques et statistiques.

$$\textit{Text Mining} = \textit{Linguistique} + \textit{Data Mining}$$

Le processus du Texte Mining

Le text mining est un processus qui implique plusieurs étapes pour extraire des informations utiles à partir des textes non structurés. Les principales étapes du processus de text mining sont les suivantes :

FIGURE 2.5 – Le processus du Text Mining.

Détaillons chaque étape de ce processus :

- **Collecte des données :** La première étape consiste à collecter les données textuelles pertinentes pour l'analyse. Ces données peuvent être des articles de presse, des rapports d'entreprise, des tweets, des commentaires de blog, des publications scientifiques, etc.
- **Prétraitement des données :** Les données collectées doivent être nettoyées et prétraitées pour éliminer les erreurs, les doublons et les informations inutiles. Le prétraitement comprend également la suppression des termes techniques inutiles et l'utilisation de la lemmatisation et de la racinisation pour normaliser les termes techniques.
- **Analyse des données :** Les données textuelles sont ensuite traitées en utilisant des techniques de traitement du langage naturel (NLP) pour extraire des informations pertinentes. Les techniques de NLP comprennent l'analyse syntaxique, l'analyse sémantique, l'analyse de sentiment, la reconnaissance d'entités nommées, etc.
- **Extraction de connaissances :** Une fois que les informations pertinentes sont identifiées, les techniques d'extraction de connaissances sont utilisées pour extraire des connaissances à partir des données. Les techniques d'extraction de connaissances comprennent la fouille de données, la fouille de textes, la classification, le regroupement, l'analyse de tendances, etc.
- **Vérification et validation :** Les connaissances extraites sont vérifiées et validées par des experts pour s'assurer de leur exactitude et de leur qualité.
- **Utilisation des résultats :** Les connaissances extraites peuvent être utilisées pour aider à la prise de décision, pour la recherche, pour la surveillance de l'opinion publique, pour la détection de fraudes, pour la surveillance de la qualité, etc.

Il est important de noter que le processus d'extraction de connaissances à partir des textes est un processus itératif. Cela signifie qu'il peut être nécessaire de répéter certaines étapes plusieurs fois pour affiner les résultats et améliorer la qualité des connaissances extraites.

2.4.5 Obstacles et défis du NLP

Le traitement automatique du langage naturel (NLP) est une discipline complexe qui nécessite la compréhension et la modélisation des caractéristiques uniques et subtiles du langage humain. Bien qu'il y ait eu des avancées significatives dans le domaine du NLP, il existe encore plusieurs obstacles et défis à surmonter pour atteindre une véritable compréhension du langage naturel.

Citons quelques-uns des principaux obstacles et défis du NLP :

- **Ambiguïté :** Les phrases et les mots peuvent avoir des significations multiples en fonction du contexte et de l'intention de l'auteur ou du locuteur. Il est difficile pour les systèmes NLP de comprendre ces nuances subtiles. L'exemple suivant présente un diction d'Al-Mutanabbi illustrant la difficulté de traiter le langage en général :
- **Variabilité linguistique :** Les langues sont très variées et peuvent avoir de nombreuses variations régionales, sociales et culturelles. Les systèmes NLP doivent être capables de gérer cette variabilité pour pouvoir comprendre et générer des textes dans différentes langues.
- **Complexité syntaxique :** La syntaxe des langues naturelles est complexe et peut inclure des phrases longues et complexes, des constructions de phrases inversées, des ellipses et des références implicites. Les systèmes NLP doivent être en mesure de comprendre et d'analyser ces constructions syntaxiques complexes.

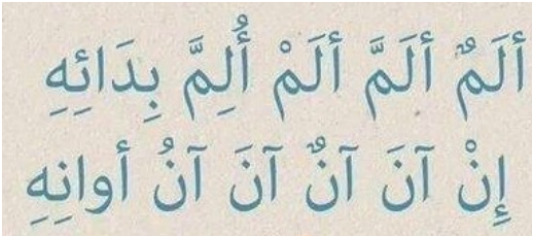


FIGURE 2.6 – Exemple illustrant la difficulté de traiter le langage en général.

- **Manque de données :** Les systèmes NLP nécessitent des ensembles de données volumineux et variés pour s'entraîner efficacement. Cependant, il peut être difficile d'obtenir des données de haute qualité et en quantité suffisante pour certaines tâches NLP.
- **Biais :** Les modèles NLP peuvent être biaisés en raison des données d'entraînement qui ne représentent pas la diversité des utilisateurs et des situations. Cela peut conduire à des résultats injustes ou inexactes.
- **Confidentialité des données :** Les données nécessaires pour entraîner les modèles NLP peuvent contenir des informations sensibles sur les utilisateurs. Les développeurs doivent être en mesure de protéger la confidentialité de ces données tout en garantissant la qualité des modèles.
- **Éthique :** Les systèmes NLP peuvent être utilisés pour des applications qui soulèvent des questions éthiques et morales, telles que la surveillance et le profilage des utilisateurs. Les développeurs doivent être conscients de ces implications éthiques et prendre des mesures pour garantir que leurs systèmes sont utilisés de manière responsable et éthique.

2.4.6 Applications du NLP

Le NLP a de nombreuses applications dans différents domaines, notamment :

- **Traduction automatique :** Les systèmes de traduction automatique utilisent des techniques de NLP pour traduire des textes d'une langue à une autre.
- **Chatbots et assistants virtuels :** Les chatbots et les assistants virtuels utilisent le NLP pour comprendre le langage naturel de l'utilisateur et fournir des réponses appropriées.
- **Résumé automatique :** Les algorithmes de NLP peuvent être utilisés pour résumer de longs documents en quelques phrases.
- **Analyse des sentiments :** Le NLP est utilisé pour analyser les sentiments exprimés dans un texte, ce qui peut être utile pour les entreprises pour évaluer la satisfaction des clients.
- **Extraction d'informations :** Les systèmes de NLP peuvent extraire des informations importantes telles que les noms, les lieux et les dates à partir de textes.
- **Reconnaissance vocale :** Les systèmes de reconnaissance vocale utilisent des techniques de NLP pour convertir la parole en texte.
- **Correction automatique :** Les algorithmes de NLP sont utilisés dans les programmes de correction automatique pour suggérer des corrections grammaticales et orthographiques.
- **Analyse de texte :** Le NLP est utilisé pour analyser de grandes quantités de texte pour détecter des tendances, des thèmes et des modèles.

- **Génération automatique de textes :** Le TALN permet de générer automatiquement du texte pour diverses applications, comme la rédaction de rapports ou la création de contenu.

Notre travail se concentre précisément sur cette dernière application, **la génération automatique de textes**, qui est le sujet principal de notre mémoire.

2.5 L'apprentissage par transfert (Transfer Learning)

2.5.1 Le Transfer Learning, c'est quoi ?

Le transfer learning, également appelé apprentissage par transfert, est une technique d'apprentissage automatique qui consiste à utiliser les connaissances acquises lors de l'apprentissage d'une tâche pour améliorer les performances d'un modèle sur une autre tâche. Au lieu de partir de zéro pour entraîner un modèle pour une nouvelle tâche, le modèle pré-entraîné est utilisé comme point de départ et est ajusté ou fine-tuné pour la nouvelle tâche. Le transfert learning permet de gagner du temps et de l'argent en évitant de devoir entraîner un modèle à partir de zéro pour chaque tâche et en utilisant des modèles pré-entraînés sur de grands ensembles de données pour améliorer la performance du modèle sur des tâches spécifiques.

L'idée principale du transfer learning est donc de transférer la connaissance apprise à partir d'un modèle entraîné sur une tâche à un autre modèle pour aider à résoudre une tâche différente ou similaire. Le transfert de connaissances peut être effectué de différentes manières, comme par exemple en utilisant les couches cachées d'un modèle pré-entraîné pour extraire des caractéristiques pertinentes d'un nouveau jeu de données.

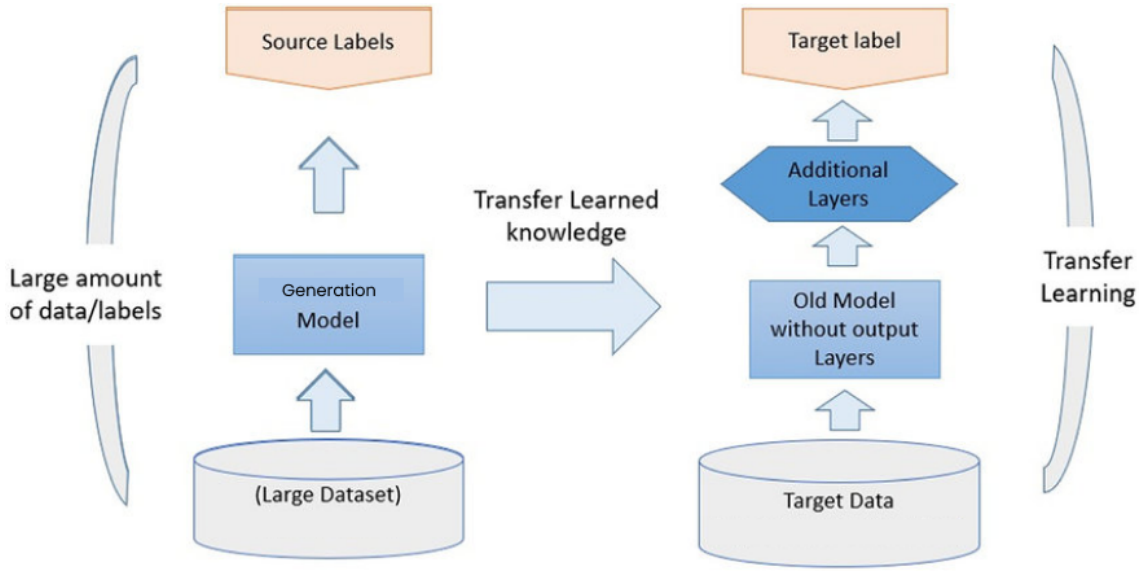


FIGURE 2.7 – Principe du Transfer Learning. [2]

2.5.2 Pourquoi le Transfer Learning ?

Le transfer learning est une approche qui permet de résoudre les problèmes de sur-apprentissage (overfitting) et d'améliorer la performance des modèles en utilisant des connaissances préalablement

acquises. En effet, les réseaux de neurones profonds ont besoin de grandes quantités de données pour être entraînés efficacement. Cependant, dans de nombreux cas, les données disponibles sont limitées ou coûteuses à acquérir. Le transfer learning permet de transférer les connaissances acquises sur des tâches similaires et/ou des données plus abondantes pour entraîner un nouveau modèle sur une tâche spécifique. Cela permet d'améliorer la généralisation et la précision du modèle, tout en réduisant le temps et les coûts nécessaires pour l'entraîner. Le Transfer learning est largement utilisé dans de nombreux domaines, notamment en vision par ordinateur et en traitement du langage naturel.

2.5.3 Les types du Transfer Learning

Il existe différentes catégories de transfer learning :

- **L'apprentissage par transfert inductif (inductive transfer learning) :**
  les données d'apprentissage sont labélisées et relèvent du même domaine, et les tâches à réaliser sont proches (exemple : reconnaitre un chat et un chien).
- **L'apprentissage par transfert non supervisé (unsupervised transfer learning) :**
  les données d'apprentissage ne sont pas étiquetées mais relèvent du même domaine. Quant à la tâche à réaliser, elle est différente.
- **L'apprentissage par transfert transductif (transductive transfer learning) :**
  les tâches sont les mêmes, mais les domaines sont différents. C'est souvent le cas dans le traitement automatique du langage avec des réseaux de neurones spécialisés dans des thématiques différentes.

2.5.4 Les approches du Transfer Learning

Il existe plusieurs approches de transfert learning en apprentissage automatique, notamment :

- **Transfert de tâches (task transfer) :**
  Cette approche consiste à entraîner un modèle sur une tâche source, puis à l'adapter pour une tâche cible. Par exemple, un modèle pré-entraîné pour la reconnaissance d'objets peut être adapté pour la classification d'images médicales.
- **Transfert de connaissances (knowledge transfer) :**
  Cette approche consiste à transférer les connaissances apprises d'une tâche source à une tâche cible, sans nécessairement adapter le modèle. Par exemple, un modèle pré-entraîné pour la reconnaissance de la parole peut être utilisé pour l'identification des locuteurs.
- **Transfert de représentations (representation transfer) :**
  Cette approche consiste à transférer les représentations apprises d'une tâche source à une tâche cible. Les représentations peuvent être utilisées pour initialiser les poids du modèle, ou comme entrée pour un nouveau modèle. Par exemple, un modèle pré-entraîné pour la détection d'objets peut être utilisé pour extraire des caractéristiques d'images qui peuvent être utilisées pour une tâche de classification d'images.

Ces approches peuvent être utilisées de manière combinée ou séparée selon le contexte de la tâche à accomplir.

2.5.5 Les stratégies du Transfer Learning

Il existe plusieurs stratégies de transfer learning, notamment :

- **Fine-tuning :**
  Le fine-tuning est une stratégie de transfert learning qui consiste à prendre un modèle pré-entraîné sur une tâche similaire à la tâche cible, et à poursuivre l'entraînement de ce modèle sur des données spécifiques à la tâche cible.

  Plus précisément, le fine-tuning consiste à réinitialiser les dernières couches du modèle pré-entraîné et à remplacer ces couches par de nouvelles couches adaptées à la tâche cible, tout en conservant les poids des autres couches. Ensuite, le modèle est entraîné sur les données spécifiques à la tâche cible, avec des taux d'apprentissage plus faibles que lors de l'entraînement initial du modèle pré-entraîné.

  Cette approche est souvent utilisée lorsque les données spécifiques à la tâche cible sont limitées, car elle permet de bénéficier de l'apprentissage préalable sur une tâche similaire, tout en adaptant le modèle aux nouvelles données. Cela peut conduire à de meilleures performances sur la tâche cible, en particulier lorsque les données d'entraînement sont rares.

- **Domain adaptation :**
  Le domain adaptation est une des stratégies du transfer learning qui consiste à appliquer un modèle entraîné sur un domaine source à un domaine cible différent mais proche du domaine source. Cela permet de transférer les connaissances acquises sur le domaine source vers le domaine cible et ainsi d'améliorer les performances du modèle sur le domaine cible.

  Dans le cas de l'apprentissage automatique, cela peut se faire en adaptant les paramètres du modèle déjà entraîné sur le domaine source pour qu'il soit plus adapté au domaine cible. Cette adaptation peut se faire en utilisant des techniques comme la réduction de dimensionnalité, la sélection de caractéristiques, ou encore l'ajout de couches spécifiques pour mieux capturer les spécificités du domaine cible.

  En résumé, le domain adaptation en transfer learning permet de tirer parti des connaissances acquises sur un domaine pour améliorer les performances d'un modèle sur un domaine proche mais différent.

2.6 Conclusion

Le traitement automatique des données textuelles repose sur des avancées majeures dans le domaine de l'intelligence artificielle, notamment l'apprentissage profond et le traitement automatique du langage naturel. Ce chapitre a exploré ces concepts clés, en fournissant des définitions approfondies, en mettant en évidence leur importance et en discutant des différentes méthodes et des applications associées. L'apprentissage profond offre des approches puissantes pour analyser les documents textuelles, en permettant l'extraction d'informations précieuses et en aidant à résoudre des problèmes complexes. Le traitement automatique du langage naturel permet de comprendre et d'analyser les données textuelles, ce qui facilite la recherche, la classification et l'analyse des informations. Enfin, le transfer learning se révèle être une approche prometteuse pour appliquer

les connaissances apprises d'une tâche à une autre, offrant des avantages en termes d'efficacité et de performance. Le traitement automatique des données textuelles ouvre de nouvelles perspectives dans de nombreux domaines, tels que l'ingénierie, les sciences et la recherche, en améliorant la capacité à exploiter les connaissances contenues dans ces documents spécialisés et à prendre des décisions éclairées.

Chapitre 3

La génération automatique de textes

3.1 Introduction

Ces dernières années, la recherche en génération automatique de texte a connu un fort développement, offrant des possibilités de générer des textes de manière automatique et personnalisée. Dans ce chapitre, nous allons présenter les différentes approches et techniques de génération automatique de textes. Nous aborderons également les différents algorithmes utilisés pour cette tâche ainsi que les métriques et les méthodes d'évaluation employées pour mesurer l'efficacité de ces algorithmes.

3.2 Définition de la génération automatique de textes

La génération automatique de textes est un sous-domaine du traitement automatique du langage naturel qui se concentre sur l'utilisation de l'informatique pour produire du texte en langage naturel à partir de données ou d'informations structurées. Elle peut être définie comme le processus par lequel les ordinateurs produisent du texte en langage naturel à partir de données ou d'informations structurées. Cela peut inclure la création de rapports, d'articles, de contenus web, de courriels, et plus encore. Le texte généré peut être aussi simple qu'une phrase ou aussi complexe qu'un rapport complet.

Formellement, la génération automatique de textes est un processus algorithmique qui transforme les données structurées en texte en langage naturel. Elle utilise des techniques d'intelligence artificielle, notamment l'apprentissage automatique et l'apprentissage profond, pour apprendre à partir de grands ensembles de données textuelles et générer du texte qui répond à des critères spécifiques. Les systèmes de génération de texte peuvent être basés sur des règles et des modèles prédéfinis, ou ils peuvent utiliser des techniques d'apprentissage pour s'adapter et améliorer leur performance au fil du temps.

3.3 Les approches de génération automatique

Il existe plusieurs approches pour la génération automatique de textes, les principales sont :

- **Approche basée sur les règles :** cette approche repose sur la définition de règles linguistiques et de templates pour la génération de texte. Elle est généralement utilisée pour

des tâches spécifiques et limitées en termes de domaines de connaissance. Exemple : La génération de réponses automatiques dans un chatbot. Les règles peuvent être définies pour répondre à des questions courantes en utilisant des templates prédéfinis. Par exemple, si l'utilisateur demande "Quel temps fait-il aujourd'hui ?", le chatbot peut répondre avec une règle prédéfinie telle que "Il fait ensoleillé et chaud aujourd'hui."

- **Approche basée sur les modèles statistiques :** cette approche utilise des modèles probabilistes pour générer du texte. Elle se base sur des corpus de données d'entraînement pour apprendre les probabilités d'occurrence des différents mots et phrases, et utilise ces probabilités pour générer des phrases cohérentes. Exemple : La génération automatique de légendes pour des images. Un modèle statistique peut être entraîné sur un grand corpus de légendes d'images pour apprendre la probabilité d'occurrence des mots et des phrases. Ensuite, lorsqu'une nouvelle image est donnée en entrée, le modèle peut générer une légende cohérente en utilisant les probabilités apprises. Par exemple, pour une image d'un chien jouant dans un parc, le modèle peut générer la légende "Un chien joyeux profite d'une journée ensoleillée dans le parc."
- **Approche hybride :** cette approche combine les deux précédentes en utilisant des règles et des modèles statistiques pour générer du texte. Elle permet de bénéficier à la fois de la flexibilité des règles et de la précision des modèles statistiques. Exemple : La génération automatique de résumés de textes. Cette approche peut combiner des règles pour identifier les informations clés à inclure dans le résumé, ainsi que des modèles statistiques pour générer des phrases grammaticalement correctes et cohérentes. Par exemple, en utilisant des règles, le système peut déterminer les phrases les plus importantes d'un article et ensuite utiliser un modèle statistique pour générer un résumé concis et informatif.
- **Approche basée sur l'apprentissage profond :** cette approche utilise des réseaux de neurones profonds pour apprendre à générer du texte de manière automatique. Elle permet de générer des textes de qualité supérieure à partir de données d'entraînement de grande taille. Exemple : La génération automatique de textes créatifs. Cette approche utilise des réseaux de neurones profonds, tels que les réseaux de neurones récurrents (RNN) ou les transformers, pour apprendre à générer du texte de manière automatique. Le modèle est entraîné sur de grandes quantités de données textuelles et peut générer des histoires, des poèmes ou d'autres formes de texte créatif. Par exemple, le modèle peut générer une nouvelle histoire fantastique avec des personnages imaginaires et des péripéties captivantes.

3.4 Les différentes étapes de génération automatique

La génération automatique de textes suit des étapes clés. Chaque étape est essentielle pour produire des textes précis, pertinents et de haute qualité en utilisant les avancées de l'intelligence artificielle et du traitement automatique des données textuelles. Le processus se résume dans les étapes suivantes :

3.4.1 Le prétraitement des données (Preprocessing)

Le prétraitement est une étape cruciale dans le processus de génération de textes, car une mauvaise gestion des informations peut entraîner un échec de traitement. Il vise à normaliser les différents éléments du texte tels que les phrases et les mots, en les mettant dans un format standard. Cette normalisation permet de créer des caractéristiques significatives et de réduire le bruit et les

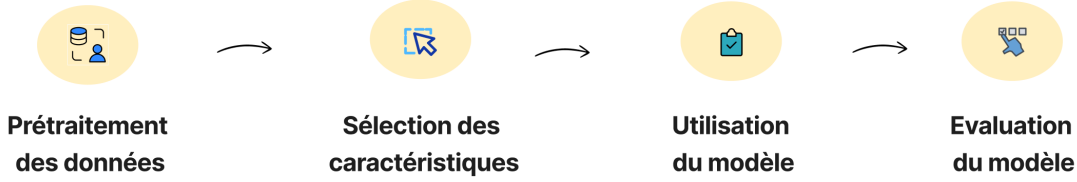


FIGURE 3.1 – Les différentes étapes de la génération automatique.

parasites, qui peuvent être introduits par des facteurs tels que des symboles non pertinents, des caractères spéciaux, etc.

Un prétraitement approprié permet de s'assurer que les données sont propres et prêtes pour l'analyse ultérieure, et contribue ainsi à la qualité globale du processus de génération automatique des textes.

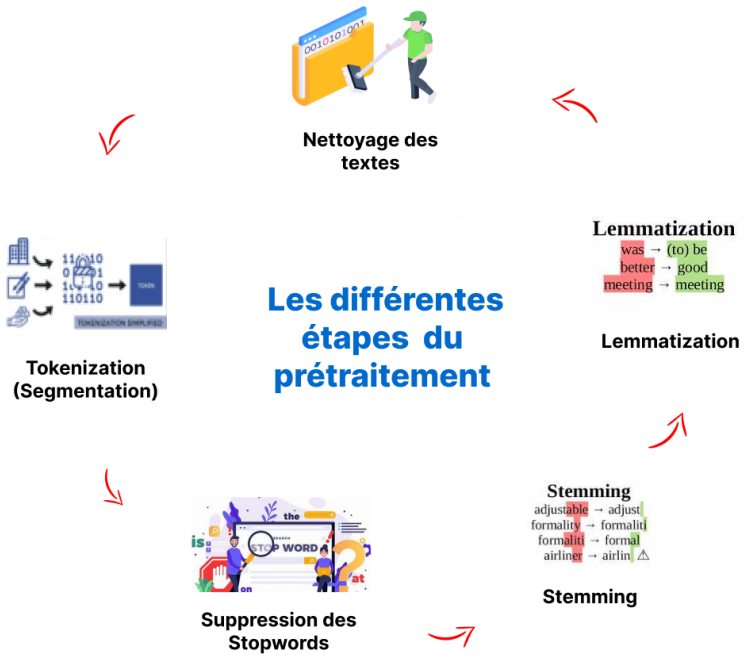


FIGURE 3.2 – Les différentes étapes du prétraitement.

Détaillons chacune de ces étapes :

- **Nettoyage des textes :**
  Il s'agit de supprimer toutes les informations inutiles ou redondantes, ainsi que les erreurs et les incohérences. L'objectif étant d'obtenir un corpus de données propre et cohérent, qui peut être utilisé pour la génération de textes de qualité.

- **Tokenization (Segmentation) :**
  La tokenization est une étape cruciale dans le processus de génération automatique. Elle consiste à diviser le texte en unités plus petites appelées "jetons" ou "tokens". Ces tokens peuvent être des mots, des phrases, des paragraphes ou même des caractères individuels, selon les besoins de la tâche de génération.

  Elle permet de représenter le texte sous une forme plus structurée et plus facile à manipuler pour les modèles de génération automatique. Elle peut également aider à réduire la complexité du texte en éliminant les éléments superflus tels que la ponctuation ou les espaces en trop.

  En général, elle est réalisée en utilisant des règles de découpage prédéfinies, ou bien en utilisant des modèles de machine learning pour apprendre à identifier les différents types de tokens dans le texte.

- **Suppression des Stopwords :**
  Les stop words sont des mots courants de la langue qui ne contiennent pas de signification significative et qui peuvent être éliminés sans perte d'information. Les exemples de stop words incluent des mots tels que "le", "la", "les", "de", "et", "à", "pour", etc.

  La suppression des stop words peut aider à réduire la taille du vocabulaire, à améliorer la performance de la modélisation et à éliminer le bruit inutile. Cette étape est généralement effectuée avant l'étape de la vectorisation, où les mots sont convertis en vecteurs numériques pour l'analyse et la modélisation ultérieures.

  Cependant, il est important de noter que la suppression des stop words peut ne pas être toujours appropriée pour certains types de textes, en particulier pour les domaines techniques spécialisés où des mots courants peuvent avoir une signification technique importante. Dans ces cas, une liste de stop words personnalisée peut être utilisée pour garantir que les mots pertinents ne sont pas supprimés.

- **Stemming**
  Technique qui consiste à réduire les mots à leur racine ou à leur forme de base, appelée "stem". Cette technique est utilisée pour normaliser les mots et améliorer la précision des recherches et des analyses.

  Dans le contexte de la génération automatique des descriptions techniques, le stemming peut être utilisé pour réduire la complexité des termes techniques et rendre le texte plus compréhensible pour un public non technique. Par exemple, le mot "programmation" pourrait être réduit à "programm" pour simplifier le texte tout en conservant l'essentiel de l'information.

Cependant, il convient de noter que le stemming peut également introduire des erreurs et des ambiguïtés dans le texte, en raison de la réduction des mots à leur forme de base. Il est donc important d'utiliser cette technique avec prudence et de la combiner avec d'autres techniques de traitement du langage naturel pour améliorer la qualité globale de la génération de textes.

- **Lemmatization :**
  Technique qui consiste à transformer les mots en leur forme de base ou lemmes. Contrairement au stemming, qui se contente de retirer les suffixes et préfixes pour ramener les mots à leur racine, la lemmatisation utilise des dictionnaires de connaissances linguistiques pour identifier la forme canonique ou la racine des mots.

  Par exemple, pour le mot "marchait", la lemmatisation va identifier que sa forme canonique est "marcher". Cette technique est utile pour normaliser les mots dans un texte et réduire le nombre de variantes d'un même mot, ce qui peut faciliter l'analyse sémantique et la recherche d'informations dans le texte.

  Dans le contexte de la génération automatique des descriptions techniques, la lemmatisation peut être utilisée pour normaliser les termes techniques et rendre le texte plus cohérent et compréhensible. Cependant, la lemmatisation peut être plus complexe que le stemming car elle nécessite l'utilisation de ressources linguistiques plus sophistiquées pour identifier la forme canonique des mots.

3.4.2 La sélection des caractéristiques (Feature extraction)

Dans cette étape, des caractéristiques pertinentes sont extraites à partir des données prétraitées. Les caractéristiques peuvent inclure des informations telles que le sujet, le contexte, les entités nommées, la syntaxe, etc. Ces caractéristiques sont utilisées pour entraîner le modèle de génération et pour générer des textes précis et pertinents.

Les caractéristiques peuvent être extraites à partir de différents niveaux de granularité du texte, tels que les mots, les phrases ou les paragraphes. L'objectif étant de créer un ensemble de caractéristiques représentant le texte de manière significative pour l'algorithme de génération automatique. Ces caractéristiques doivent être suffisamment discriminantes pour permettre à l'algorithme de distinguer les différents types de documents et de générer un texte pertinent et cohérent.

Les techniques de l'extraction des caractéristiques :

BOW (Bag of Words)

Une méthode courante pour la sélection des caractéristiques en génération automatique de textes. Elle consiste à représenter un document sous forme d'un sac de mots, en ignorant l'ordre des mots et leur contexte. Chaque mot unique du document est considéré comme une caractéristique, et un vecteur de fréquence de chaque mot est créé pour représenter le document.

Cette méthode est simple à mettre en œuvre et efficace pour les tâches de classification de texte, mais elle peut ne pas tenir compte de la sémantique ou de la structure du texte. De plus, elle ne prend pas en compte les synonymes et les mots ayant des sens similaires, ce qui peut affecter

la qualité de la génération de texte. Ces limitations ont conduit au développement de techniques plus avancées telles que **Word2Vec** et **GloVe**.

Voici un exemple de représentation en utilisant le model de BOW :

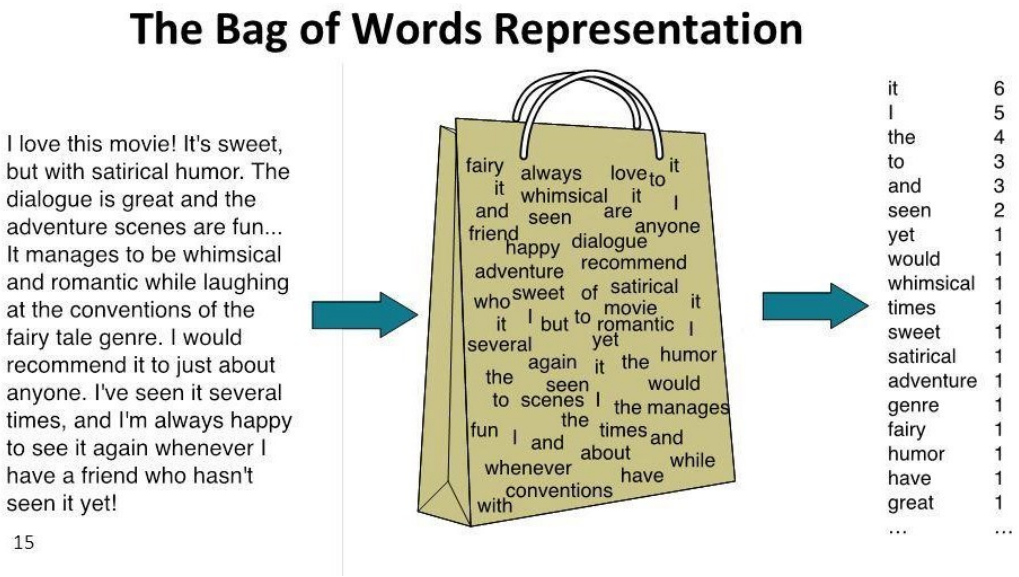


FIGURE 3.3 – Exemple de représentation en utilisant le Bag of Word model. [3]

TF-IDF (Term Frequency-Inverse Document Frequency)

Une méthode de pondération souvent utilisée en traitement de texte pour mesurer l'importance relative d'un terme dans un document ou une collection de documents.

L'idée derrière la méthode TF-IDF est de donner une pondération élevée aux termes qui apparaissent fréquemment dans un document, mais pas dans l'ensemble de la collection de documents. Cela permet de mettre en avant les termes qui sont caractéristiques d'un document par rapport aux autres documents.

La pondération TF-IDF d'un terme t dans un document d est calculée comme suit :

$$TFIDF(t, d) = TF(t, d) * IDF(\text{t}) \tag{3.1}$$

où :

— TF(t,d) : la fréquence du terme t dans le document d (nombre de fois où le terme apparaît dans le document).

— IDF(t) : la mesure d'inverse de la fréquence de document du terme t, calculée comme suit :

$$IDF(t) = log(N/df(t)) \tag{3.2}$$

  — N : le nombre total de documents dans la collection.

  — df(t) : le nombre de documents dans la collection qui contiennent le terme t.

Le résultat final de la pondération TF-IDF pour un terme t dans un document d est le produit de la fréquence du terme dans le document et son inverse de document fréquence.

Cela signifie que les mots qui apparaissent fréquemment dans un document, mais qui sont également fréquents dans l'ensemble des documents, ont un score TF-IDF plus faible, tandis que les mots qui sont rares dans l'ensemble des documents mais fréquents dans un document particulier ont un score TF-IDF plus élevé. Cette pondération est utile pour identifier les mots clés importants dans un document et pour les comparer avec d'autres documents.

Word embedding (encodage de mots)

Le word embedding est une méthode de représentation des mots sous forme de vecteurs dans un espace vectoriel continu. Cette méthode est utilisée pour capturer la similarité sémantique et syntaxique entre les mots. Elle est largement utilisée dans les tâches de traitement automatique du langage naturel, y compris la génération automatique de textes.

L'idée derrière les word embeddings est de représenter chaque mot comme un vecteur dense dans un espace vectoriel continu, où chaque dimension représente une caractéristique du mot. Les embeddings sont construits de manière à ce que les mots similaires aient des vecteurs similaires, c'est-à-dire qu'ils soient proches les uns des autres dans l'espace vectoriel.

Il existe plusieurs techniques de word embeddings, telles que **Word2Vec**, **GloVe** et **FastText**. Ces techniques permettent de représenter les mots sous forme de vecteurs denses, en utilisant différentes approches pour apprendre la similarité sémantique et syntaxique entre les mots à partir d'un corpus de texte. Les embeddings ainsi obtenus peuvent ensuite être utilisés dans des modèles de génération de textes pour capturer la signification des mots dans le contexte de la génération..

a. Word2Vec

Un modèle d'apprentissage automatique pour la représentation vectorielle des mots. Il permet de créer des représentations vectorielles d'un vocabulaire à partir d'un corpus de textes, où chaque mot est représenté par un vecteur de nombres réels. Ces vecteurs sont construits de telle manière que les mots qui ont un contexte similaire dans le corpus ont des représentations vectorielles similaires.

Dans la génération automatique de textes, Word2Vec peut être utilisé pour la représentation des mots clés, l'identification des relations sémantiques entre les termes techniques, la détection de similarité entre les phrases, etc. Cela permet d'améliorer la qualité de la génération en utilisant des vecteurs sémantiques plutôt que simplement des fréquences de mots.

Il existe deux méthodes principales pour créer des représentations vectorielles de mots avec le Word2Vec : **CBOW** et **Skip-Gram**. L'image suivante illustre le différence entre les deux méthodes :

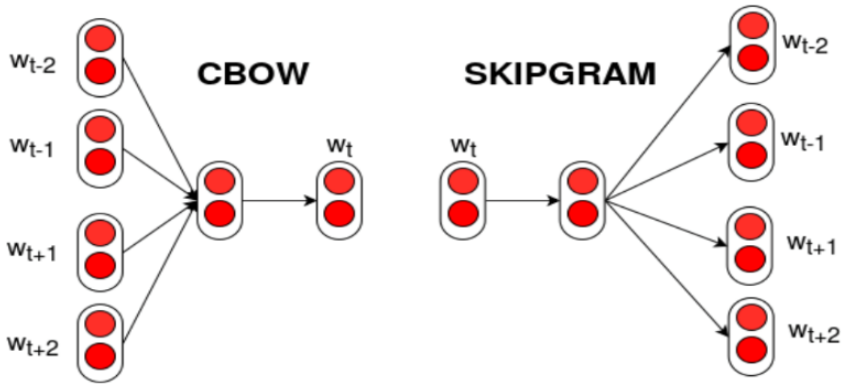


FIGURE 3.4 – La différence entre CBOW et Skip-Gram. [4]

- **CBOW (Continuous Bag of Words)**

  Un mélange entre les 2 techniques "Bag of Words" et "NGram" . Elle vise à prédire un mot donné en fonction de ses mots environnants.

  L'idée de BOW dépend de l'utilisation d'un certain nombre de mots dans le texte et de faire des nombres 1 et 0 en fonction de la présence de chaque mot, et l'idée de NGram dépend de l'utilisation d'un ou plusieurs des mots précédents pour en déduire le mot suivant, et donc en utilisant CBOW nous pouvons utiliser plus d'un mot dans la même phrase pour déduire un mot spécifique.

  Il peut être défini comme un réseau de neurones, mais il est utilisé pour prédire quel est le mot manquant dans une phrase particulière (souvent le dernier mot), en calculant l'intégration matricielle des mots d'entrée, et à partir de laquelle il est traité pour atteindre le couche suivante dans le réseau, et enfin la couche finale avec une fonction d'activation *softmax* pour sélectionner le mot manquant.

- **Skip-Gram**

  Contrairement à l'algorithme CBOW, qui utilise le contexte pour prédire le mot cible, l'algorithme Skip-Gram utilise le mot cible pour prédire le contexte.

L'algorithme Skip-Gram consiste à créer un modèle neuronal qui utilise une couche cachée pour apprendre les représentations vectorielles des mots. Chaque mot est représenté par un vecteur dense de nombres réels, et les mots qui ont des significations similaires sont situés dans des espaces vectoriels similaires.

L'entraînement de l'algorithme Skip-Gram se fait en maximisant la probabilité que les mots du contexte soient prédits à partir du mot cible. Pour cela, il utilise une fonction de coût appelée Negative Sampling (échantillonnage négatif) qui permet de réduire le temps de calcul nécessaire à l'apprentissage.

b. GloVe

Vise à capturer les relations sémantiques et syntaxiques entre les mots en attribuant à chaque mot un vecteur numérique dense.

Le principe fondamental de GloVe repose sur l'idée que les mots qui apparaissent fréquemment dans des contextes similaires partagent une signification sémantique similaire. Le modèle GloVe utilise une matrice de co-occurrence qui enregistre la fréquence à laquelle chaque paire de mots apparaît ensemble dans un corpus de texte. Cette matrice est ensuite utilisée pour estimer les représentations vectorielles des mots.

Contrairement à d'autres approches, comme Word2Vec, GloVe combine à la fois des informations locales (fréquence des co-occurrences) et des informations globales (fréquence des mots dans l'ensemble du corpus) pour générer des représentations de mots plus robustes. Il utilise une fonction d'optimisation pour apprendre ces représentations de manière à minimiser la différence entre les produits scalaires des vecteurs de co-occurrence et les logarithmes des fréquences de co-occurrence.

Les vecteurs GloVe résultants peuvent être utilisés dans diverses tâches de NLP, telles que la recherche d'informations, la classification de textes, la traduction automatique et la génération de texte. Les vecteurs GloVe permettent de capturer des relations lexicales et sémantiques, ce qui facilite la comparaison et l'analyse de similitude entre les mots.

GloVe a été largement adopté dans la communauté de la recherche en NLP en raison de sa simplicité, de sa performance et de sa capacité à produire des représentations vectorielles riches en informations sémantiques.

c. FastText

FastText est une bibliothèque d'apprentissage automatique open source développée par Facebook AI Research, utilisée principalement dans le domaine du traitement automatique du langage naturel (NLP). Elle est conçue pour l'apprentissage de représentations vectorielles de mots et de phrases en utilisant des méthodes de classification et de regroupement de texte.

L'une des caractéristiques distinctives de FastText est sa capacité à représenter les mots en tenant compte des informations de sous-mots. Contrairement aux méthodes traditionnelles de représentation de mots, qui considèrent les mots comme des unités discrètes, FastText décompose les mots en sous-mots appelés "n-grammes". Cela permet à FastText de capturer les informations de structure interne des mots, ce qui est particulièrement utile pour traiter les mots peu fréquents ou inconnus.

L'apprentissage des représentations de mots avec FastText se fait via des réseaux de neurones artificiels. Il utilise une architecture basée sur des sacs de mots (bag-of-words) et utilise des techniques d'optimisation efficaces pour entraîner rapidement les modèles sur de grands ensembles de données textuelles.

POS (Part of Speech)

Désigne la catégorie grammaticale d'un mot dans une phrase. Les catégories grammaticales courantes sont le nom, le verbe, l'adjectif, l'adverbe, la préposition, la conjonction, le pronom et l'interjection.

L'utilisation de POS dans la génération automatique de textes permet de mieux comprendre la structure grammaticale des phrases et donc de générer des textes plus cohérents et naturels. Cela peut également aider à résoudre des problèmes de construction de phrases ambigües ou incorrectes. En outre, l'utilisation de POS peut aider à identifier les relations sémantiques entre les mots, ce qui peut être utile pour la génération de textes plus précis et pertinents.

NER (Named-Entity Recognition)

La reconnaissance d'entités nommées (Named-Entity Recognition ou NER) est une technique de traitement de langage naturel qui consiste à identifier et extraire les noms propres et autres entités nommées (tels que les noms d'organisations, de lieux, de dates, de quantités, etc.) à partir de textes non structurés. Voici un exemple d'application de la technique :

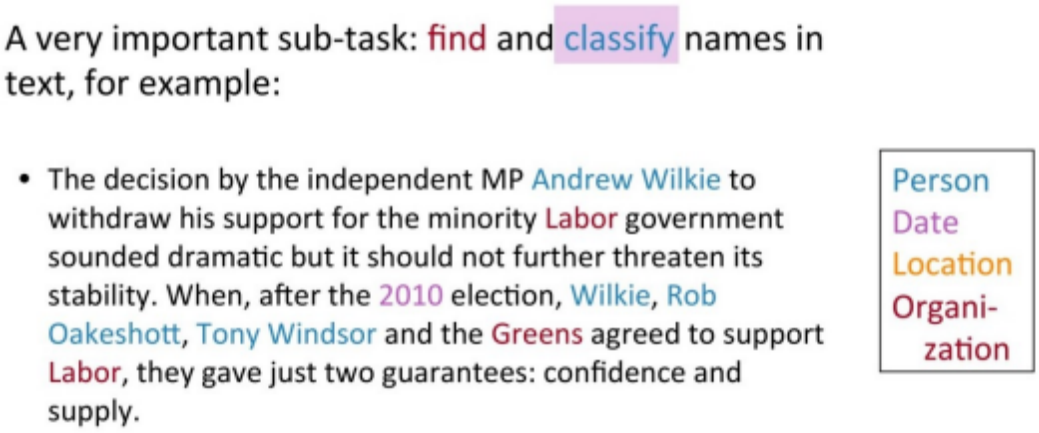


FIGURE 3.5 – Exemple d'application du NER.

Matchers

Les matchers, également appelés analyseurs de règles, sont des outils d'extraction d'informations qui permettent de chercher des motifs spécifiques dans un texte et de les marquer en fonction de règles prédéfinies. Par exemple, un matcher pourrait être utilisé pour chercher des expressions de temps dans un texte et les marquer comme des dates. Les matchers peuvent être utilisés en combinaison avec d'autres outils d'analyse de texte tels que le POS tagging et la NER pour extraire des informations plus complexes et plus précises.

Structure syntaxique (Syntactic Structure)

La "Syntactic Structure" fait référence à la structure grammaticale d'un texte. L'analyse de la structure syntaxique peut aider à comprendre la relation entre les différents éléments du texte, ce qui peut être utilisé pour générer des textes plus cohérents et bien structurés.

Par exemple, en utilisant des outils d'analyse syntaxique, il est possible de détecter les phrases principales et les phrases secondaires dans un texte, ainsi que les relations entre ces phrases. Cela peut être utilisé pour générer des textes bien structurés avec des paragraphes clairement définis et des transitions logiques entre eux.

Il existe deux modèles de cette structure :

- **Le modèle circulaire (constituency)**

  Il s'agit d'un arbre syntaxique qui représente les relations hiérarchiques entre les différents éléments constitutifs de la phrase, en organisant ces éléments en groupes appelés constituants. Les constituants sont des sous-parties de la phrase qui sont identifiées en fonction de leur rôle syntaxique dans la phrase, tels que le sujet, le verbe, l'objet, etc. Chaque constituant peut être lui-même divisé en sous-constituants plus petits, jusqu'à ce que l'on atteigne les mots individuels de la phrase. L'image suivante montre un exemple illustrant ce modèle :

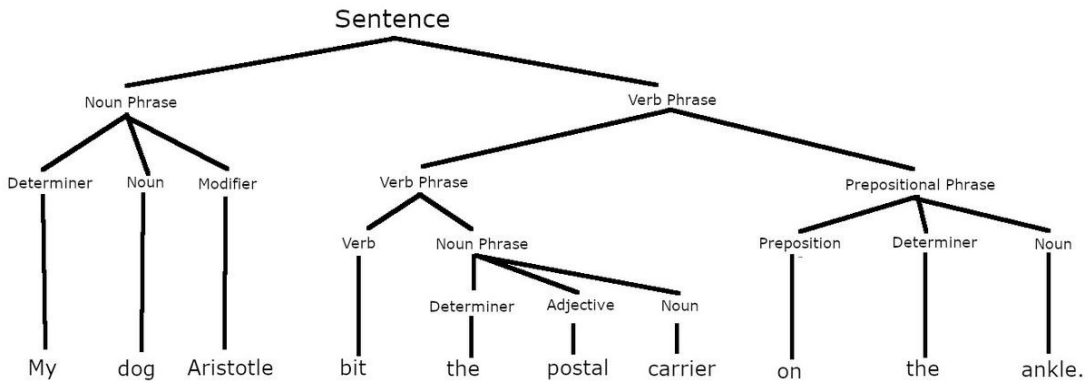


FIGURE 3.6 – Exemple illustrant le modèle circulaire.

- **La structure de dépendance**

  La structure de dépendance, appelée également arbre de dépendance, est une méthode pour représenter la structure syntaxique d'une phrase en utilisant des arcs qui relient chaque mot à son mot de tête ou à un autre mot dépendant. Chaque arc est étiqueté avec le type de relation syntaxique qui existe entre les deux mots. L'image suivante montre un exemple illustrant cette méthode :

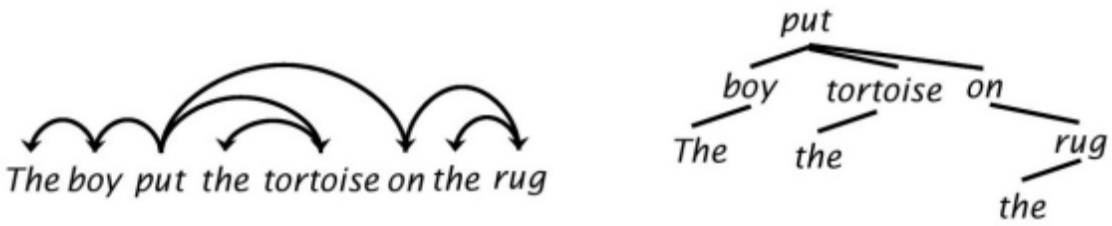


FIGURE 3.7 – Exemple illustrant la structure de dépendance.

Visualisation de textes (Text Vizualization)

La visualisation de textes est une méthode pour représenter visuellement des données textuelles complexes de manière à faciliter leur compréhension et leur analyse. Elle peut être utilisée pour découvrir des modèles, des tendances ou des relations dans les données textuelles, ainsi que pour communiquer les résultats de l'analyse à un public non technique.

Les techniques de visualisation de textes incluent :

- **Les nuages de mots (Word clouds) :** Une représentation visuelle des mots les plus fréquents dans un texte, où la taille de chaque mot est proportionnelle à sa fréquence.
- **Les Heat maps :** Une représentation visuelle des valeurs numériques associées à chaque mot ou groupe de mots dans un texte.
- **Les graphiques de réseau :** montrent les relations entre les mots, les entités ou les thèmes dans un corpus de texte, où chaque nœud représente un mot ou une entité et chaque lien représente une relation entre eux.
- **Les cartes thermiques :** montrent la fréquence des mots ou des entités dans un corpus de texte, où les zones les plus foncées représentent les éléments les plus fréquents.
- **Les graphiques de dispersion :** montrent la distribution des mots ou des entités dans un corpus de texte en fonction de leur fréquence et de leur position dans le texte.
- **Les diagrammes de Venn :** montrent les relations entre les ensembles de mots ou d'entités qui se chevauchent dans un corpus de texte.
- **Graphes de similarité :** Des représentations graphiques qui montrent la similarité entre les mots ou les groupes de mots dans un texte.
- **Graphes de co-occurrence :** Des représentations graphiques qui montrent comment les mots ou les groupes de mots se produisent.

En utilisant ces techniques de visualisation, les analystes de textes peuvent explorer les données de manière interactive, découvrir des modèles et des tendances, et communiquer les résultats de l'analyse à un public non technique de manière efficace et claire.

3.4.3 L'utilisation d'un modèle de génération

Cette étape consiste à choisir et à entraîner un modèle de génération approprié pour le problème spécifique de la génération de textes.

Les modèles de langage ont considérablement évolué ces dernières années avec l'utilisation de réseaux de neurones profonds, tels que les **réseaux de neurones récurrents (RNNs)** et les **Transformers**. Ces modèles peuvent prendre en compte des contextes plus larges et produire des textes plus fluides et plus naturels. Dans ce qui suit nous allons présenter quelques modèles jugés les plus pertinents pour la résolution de notre problématique.

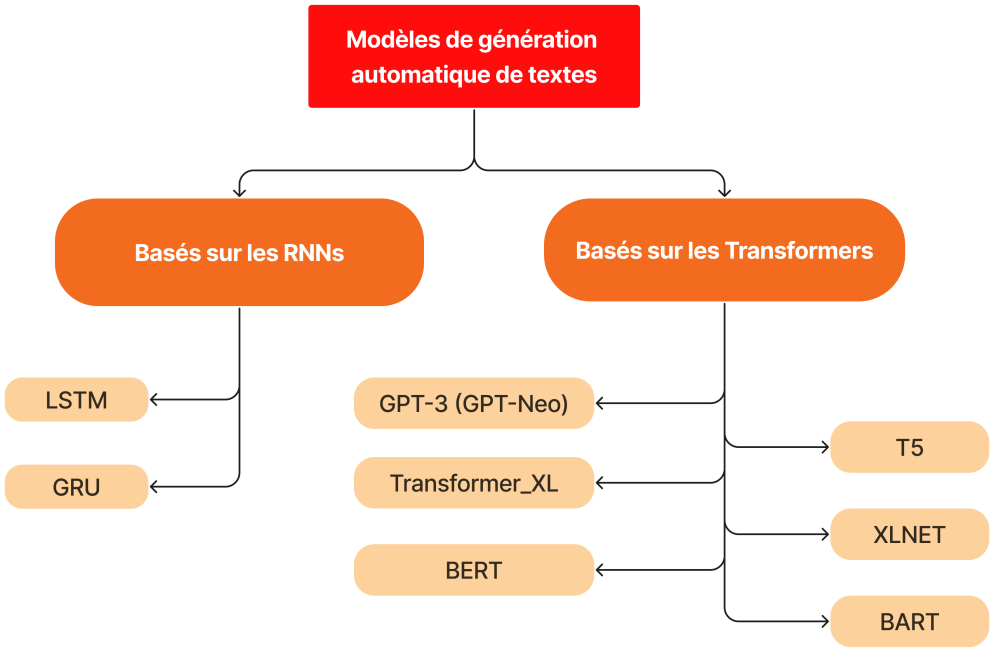


FIGURE 3.8 – Les différents modèles de génération.

I. Les modèles basés sur les RNNs

Les modèles de traitement automatique du langage naturel basés sur l'architecture des réseaux de neurones récurrents (RNNs) ont été parmi les premiers modèles utilisés pour la génération de textes. Ils ont été conçus pour travailler sur des séquences de données, ce qui les rendent particulièrement adaptés pour traiter des séquences de textes.

Principe du fonctionnement des RNNs :

Un RNN est très similaire à un réseau de neurones non bouclé classique, dans lequel le flux des activations se dirige dans un sens unique, depuis la couche d'entrée vers la couche de sortie. Son architecture est composée d'une couche d'entrée, de couches cachées et d'une couche de sortie.

En revanche, le RNN se distingue par la capacité de certaines connexions à revenir en arrière dans le réseau. Expliquons le principe de fonctionnement du RNN à partir d'un RNN constitué par un seul neurone :

Ce neurone reçoit des entrées et produit une sortie. Dans le détail, à chaque étape temporelle t, ce neurone récurrent reçoit le vecteur d'entrée x(t) ainsi que sa propre sortie produite à l'étape temporelle précédente o(t-1). Nous pouvons représenter ce réseau le long d'un axe du temps. On dit alors qu'on a «déplié le réseau dans le temps».

À chaque étape temporelle t, chaque neurone récurrent reçoit à la fois le vecteur d'entrée x(t) et le vecteur de sortie de l'étape temporelle précédente o(t-1).

Chaque neurone récurrent possède 2 types de poids :

- des poids (notés U sur le schéma ci-dessous) reliant les entrées à la sortie (comme pour un réseau de neurones classique).
- des poids (notés V) entre la sortie et l'entrée de la couche, qui sont les connexions récurrentes.

Le schéma suivant montre la structure d'un réseau de neurones récurrents :

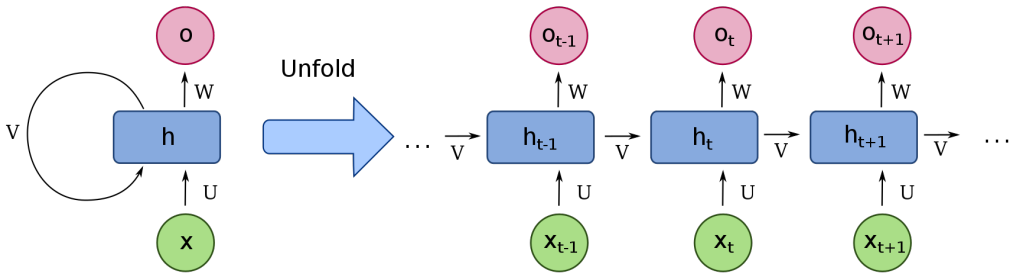


FIGURE 3.9 – Schéma d'un réseau de neurones récurrents à une unité reliant l'entrée et la sortie du réseau. À droite, la version dépliée de la structure. [5]

En fait, la sortie d'un neurone récurrent étant une fonction de toutes les entrées des étapes précédentes, on considère que ce neurone possède une forme de **mémoire**.

On appelle cellule de mémoire, la partie du réseau de neurones récurrent qui conserve un état entre plusieurs étapes temporelles.

L'état d'une cellule à l'étape t, noté h(t) (h pour « hidden » layer qui signifie couche cachée), est une fonction de certaines entrées à cette étape temporelle et de son état à l'étape temporelle précédente : $h(t) = f(h(t\text{-}1), x(t))$. La sortie est donc fonction de l'état précédent et des entrées courantes.

Les différents types de RNNs :

Les réseaux de neurones récurrents sont capables de gérer plusieurs types de données en entrée comme en sortie.

- **One-to-one (Un à un) :** ce type de RNN prend une entrée et produit une sortie. C'est l'utilisation la plus simple de l'architecture RNN et est similaire à un réseau de neurones classique.

- **One-to-many (Un à plusieurs) :** ce type de RNN prend une seule entrée et produit plusieurs sorties. Par exemple, il peut générer une séquence de mots à partir d'une image.
- **Many-to-one (Plusieurs à un) :** ce type de RNN prend plusieurs entrées et produit une seule sortie. Par exemple, il peut être utilisé pour classer une séquence de mots en une catégorie donnée (sentiment classification)
- **Many-to-many (Plusieurs à plusieurs) :** ce type de RNN prend plusieurs entrées et produit plusieurs sorties. Il peut être utilisé pour prédire une séquence de mots à partir d'une autre séquence de mots, comme la traduction automatique.

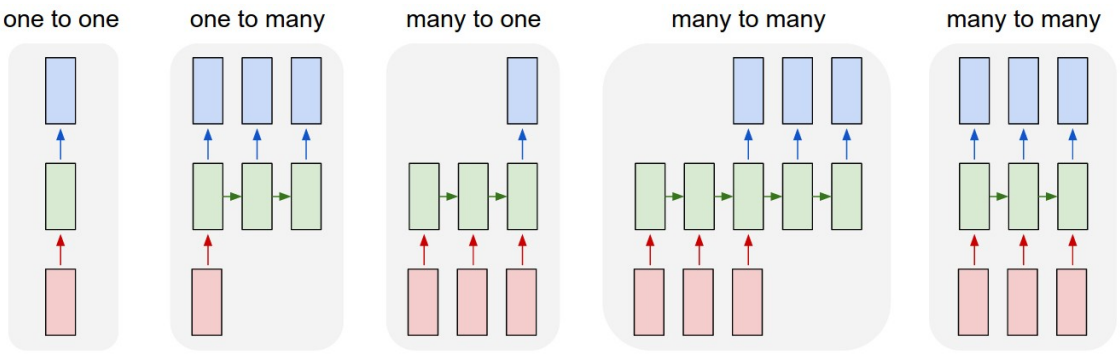


FIGURE 3.10 – Les principaux types des RNNs. [6]

Les variantes des RNNs :

Deux des RNNs les plus connus et largement utilisés dans la génération de texte sont **LSTM** et **GRU**. LSTM a été introduit en 1997 par Sepp Hochreiter et Jurgen Schmidhuber, tandis que GRU a été introduit plus récemment en 2014 par Junyoung Chung et son équipe.

a. LSTM(Long Short-Term Memory)

Le modèle LSTM, cellule de longue mémoire à court terme est un RNN particulier qui a été conçu pour surmonter les problèmes de la vanishing gradient (disparition/explosion du gradient) et de la dépendance à long terme qui se posent avec les RNNs classiques lorsqu'ils sont utilisés pour traiter des séquences de données de longueur importante.

Les LSTM disposent d'une structure de cellule de mémoire qui leur permet de stocker des informations pour une durée prolongée et de décider quand il est nécessaire de mettre à jour ces informations ou de les oublier. Cette structure de cellule est composée de plusieurs portes, notamment la porte d'entrée **(input gate)**, la porte d'oubli **(forget gate)** et la porte de sortie **(output gate)**, qui régulent le flux d'informations entrantes et sortantes de la cellule de mémoire.

Le fonctionnement des LSTM peut être divisé en trois étapes principales : la première étape est la décision de la mise à jour des informations stockées dans la cellule de mémoire en utilisant la porte d'entrée, la deuxième étape est la décision de ce qu'il faut oublier en utilisant la porte d'oubli, et la troisième étape est la décision de la sortie en utilisant la porte de sortie.

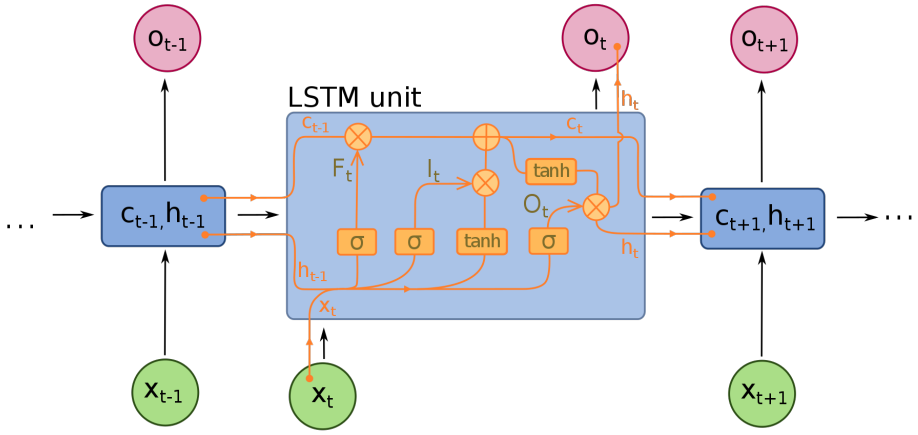


FIGURE 3.11 – Schéma d'un réseau LSTM à une unité. Le graphe des opérations est détaillé pour l'étape t. Les poids ne sont pas indiqués. [5]

b. GRU (Gated Recurrent Units)

Ils ont été développés pour résoudre le problème de la disparition du gradient qui se produit lors de la rétropropagation dans les RNNs. Les GRU utilisent des portes pour contrôler le flux d'information dans le réseau. Contrairement aux LSTM, les GRU n'ont que deux portes : une porte de mise à jour **(update gate)** et une porte de réinitialisation **(reset gate)**. La porte de mise à jour contrôle la quantité d'information nouvelle qui est mise à jour dans l'état caché, tandis que la porte de réinitialisation détermine la quantité d'information qui est conservée de l'état précédent. Les GRU sont connus pour être plus rapides à entraîner que les LSTM et peuvent fournir de bonnes performances sur des tâches de traitement du langage naturel telles que la génération de textes et la traduction automatique.

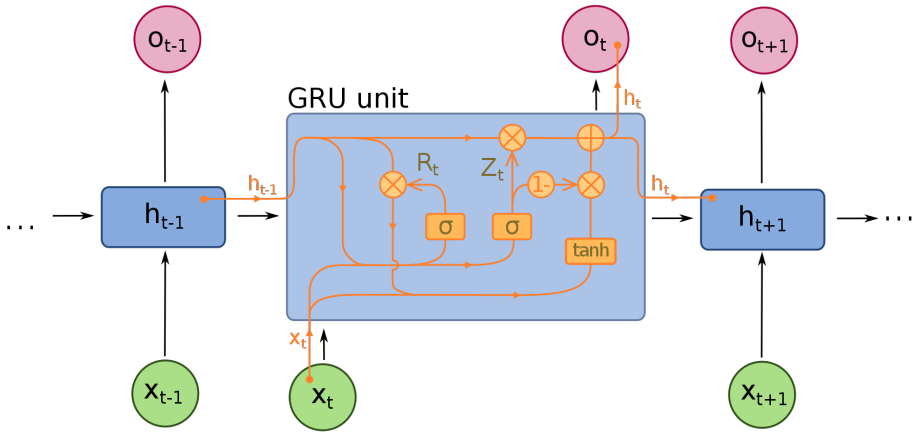


FIGURE 3.12 – Schéma d'un réseau GRU à une unité. [5]

Limites des RNNs :

Les RNNs sont très utiles pour la génération de textes, mais ils présentent certains inconvénients qui peuvent limiter leur performance. L'un des principaux inconvénients des RNNs est la difficulté à conserver des informations à long terme. Cela peut entraîner des problèmes de vanishing gradient ou de saturation, qui peuvent affecter la qualité de la génération de textes.

Une solution proposée à ce problème est l'utilisation de modèles basés sur des architectures plus avancées comme les **Transformers**, qui peuvent capturer des informations à long terme de manière plus efficace. Les Transformers ont également l'avantage de permettre une parallélisation plus efficace, ce qui permet un entraînement plus rapide et plus efficace du modèle.

En résumé, bien que les RNNs soient encore largement utilisés pour la génération de textes, les modèles basés sur les Transformers ont montré une performance améliorée dans de nombreux cas, grâce à leur capacité à capturer des informations à long terme de manière plus efficace.

II. Les modèles pré-entraînés - PLM (Basés sur les Transformers)

Pourquoi les Transformers ?

Les Transformers sont une famille de modèles de traitement de langage naturel qui ont été introduits en 2017 par Vaswani et al. Ils se distinguent des RNNs en utilisant des couches d'attention pour calculer les représentations des mots dans une phrase plutôt que de parcourir séquentiellement chaque mot. Les Transformers ont connu un grand succès dans diverses tâches de NLP, notamment la traduction automatique, la génération de textes et les question-réponse.

En ce qui concerne la génération de textes, les modèles basés sur l'architecture des Transformers ont montré de très bons résultats. En effet, ces modèles ont une capacité importante à capturer des relations non locales entre les mots d'une phrase, ce qui est particulièrement important pour cette tâche.

En outre, les Transformers ont permis une amélioration significative de la vitesse de calcul par rapport aux RNNs, ce qui a permis l'entraînement de modèles plus grands sur des corpus de données massifs.

L'architecture des Transformers

L'architecture des transformers repose sur deux concepts clés : **l'attention** et **les couches d'encodeurs/décodeurs**.

- **Attention :**
  L'attention est une technique qui permet à un modèle de donner davantage de poids à certaines parties d'une séquence d'entrée lors de la prise de décision. Dans le contexte du traitement du langage naturel, cela signifie que le modèle peut se concentrer sur des mots ou des phrases spécifiques lorsqu'il génère ou comprend du texte. L'attention est basée sur des vecteurs d'attention qui sont calculés en fonction de la similarité entre les mots ou les positions dans la séquence d'entrée.
- **Couches d'encodeurs/décodeurs :**
  L'architecture des Transformers est composée de couches d'encodeurs et de décodeurs empilées. Chaque couche est constituée de sous-modules qui effectuent des opérations spécifiques sur les données d'entrée.
  - **Encodeurs (Input Embedding) :** Les couches d'encodeurs prennent en entrée une séquence de mots ou de symboles et les transforment en une représentation vectorielle de haute qualité. Chaque couche d'encodeur comprend deux sous-modules principaux : le mécanisme d'attention multi-têtes et le réseau de neurones positionnel.

- **Mécanisme d'attention multi-têtes (Multi-Head Attention) :** Il permet au modèle de capturer les relations à longue distance entre les mots dans la séquence d'entrée en calculant les vecteurs d'attention pour chaque mot. Ces vecteurs d'attention pondèrent l'importance des autres mots dans la séquence lors de la génération de la représentation vectorielle.
- **Réseau de neurones positionnel (Positional Encoding) :** Il encode les informations de position dans la séquence d'entrée, permettant au modèle de comprendre l'ordre et la structure des mots.

— **Décodeurs (Output Embedding) :** Les couches de décodeurs prennent la représentation vectorielle générée par les encodeurs et génèrent une séquence de sortie, souvent mot par mot. Chaque couche de décodeur comprend également un mécanisme d'attention multi-têtes, mais il est légèrement modifié pour se concentrer sur les parties pertinentes de la représentation vectorielle générée par les encodeurs.

Les couches d'encodeurs et de décodeurs sont empilées les unes sur les autres, généralement plusieurs fois, pour augmenter la capacité du modèle. Les informations sont transmises de couche en couche lors de la génération de la sortie.

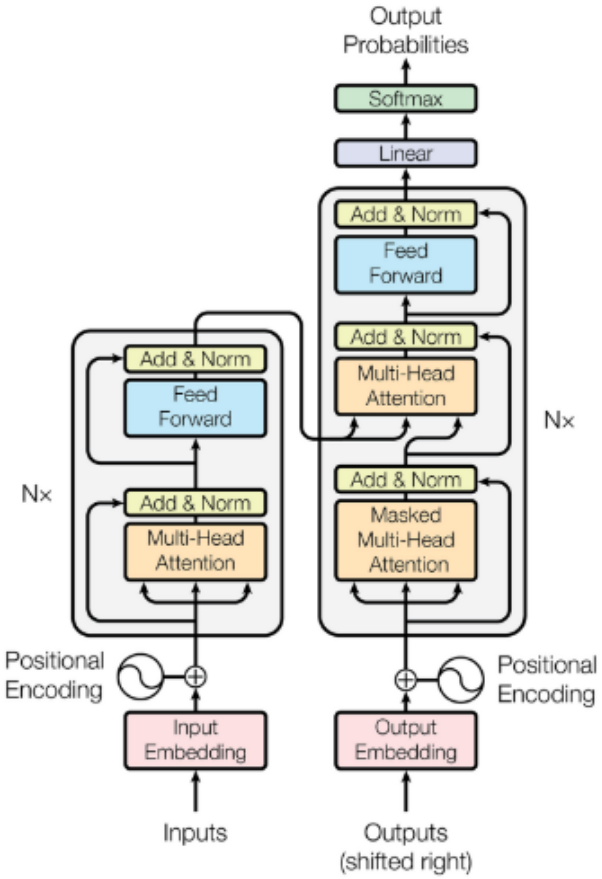


FIGURE 3.13 – Architecture du modèle Transformer. [7]

Ces modèles sont conçus pour gérer des séquences de données (particulièrement utiles en NLP). Contrairement aux modèles Seq2seq utilisés en RNNs, ils ne contiennent ni récurrence ni convolution, ils ne nécessitent donc pas de traiter les séquences dans l'ordre. Ce fait nous permet de paralléliser (beaucoup plus que les RNNs) et réduit le temps de formation.

Les modèles

Il existe divers modèles utilisant cette technique puissante, connus sous le nom de **"Pretrained Language Models - PLM"**. Dans ce qui suit, nous allons présenter certains de ces modèles à savoir le modèle **GPT-3 (GPT-Neo)** et **BART**, en détaillant comment ils peuvent être appliqués à la génération de textes.

L'image suivante montre le processus illustratif d'application des PLMs à la génération de textes. Nous divisons le processus en trois étapes principales : l'apprentissage de la représentation des entrées, la conception et la sélection de l'architecture du modèle, l'optimisation des paramètres du modèle.

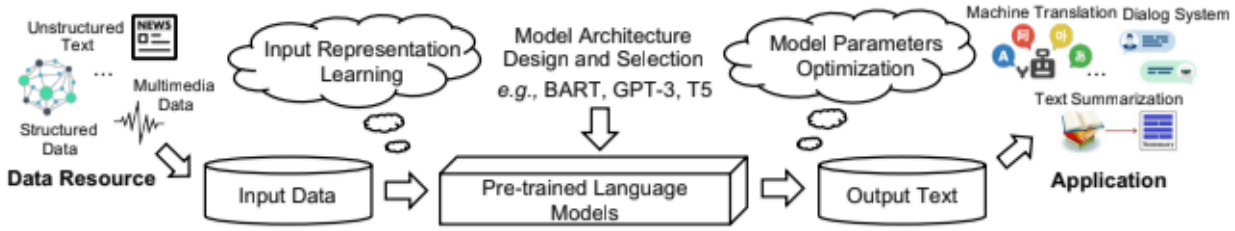


FIGURE 3.14 – Le processus d'application des PLMs à la génération de textes.

a. GPT-3 (GPT-Neo)

Description

GPT-3 (Generative Pretrained Transformer 3), dont une version open-source est connue sous le nom de GPT-Neo, est un modèle de langage basé sur le transformer, développé par OpenAI. Il s'agit d'un modèle d'apprentissage automatique non supervisé qui utilise le machine learning pour produire du texte humain. GPT-3 est le successeur de GPT-2 et est actuellement l'un des plus grands et des plus puissants modèles de langage disponible.

Architecture

GPT-3 utilise une architecture de transformer avec 175 milliards de paramètres. Il est basé sur une architecture de réseau de neurones transformer, qui utilise l'attention multi-têtes pour pondérer l'importance relative des mots dans l'entrée lors de la génération de texte.

Caractéristiques

GPT-3 est capable de générer du texte qui est presque indiscernable de celui écrit par un humain. Il peut répondre aux questions, écrire des essais, résumer des textes, traduire des langues, et même générer du code informatique. Il est également capable de tâches d'apprentissage à quelques

exemples, ce qui signifie qu'il peut apprendre à partir d'un petit nombre d'exemples de tâches.

Représentation des inputs

Les inputs pour GPT-3 sont généralement des séquences de texte, qui sont encodées en utilisant un tokenizer basé sur BPE. Le modèle prédit ensuite le prochain mot de la séquence à chaque étape, en utilisant le contexte des mots précédents dans la séquence.

Fine-tuning

Le fine-tuning de GPT-3 implique l'entraînement du modèle sur une tâche spécifique en utilisant un petit ensemble de données d'entraînement. Cela permet au modèle d'adapter ses poids pré-entraînés à la tâche spécifique, améliorant ainsi ses performances sur cette tâche.

b. BART

Description

BART (Bidirectional and Auto-Regressive Transformers) est un modèle de langage développé par Facebook AI. Il est basé sur l'architecture du transformer et est conçu pour être bidirectionnel, ce qui signifie qu'il peut comprendre le contexte à partir des deux directions dans une séquence de texte.

Architecture

BART utilise une architecture de transformer qui est similaire à celle de GPT et de BERT, mais avec une différence clé : BART est pré-entraîné en supprimant arbitrairement du texte et en apprenant à le reconstruire. Cela lui permet de comprendre le contexte à partir des deux directions dans une séquence de texte.

Caractéristiques

BART est capable de tâches de génération de texte et de compréhension de texte, y compris la réponse aux questions, la traduction de langues, et la rédaction de résumés. Il est également capable de tâches de classification de texte et d'extraction d'information.

Représentation des inputs

Comme GPT-3, BART prend en entrée des séquences de texte, qui sont encodées en utilisant un tokenizer. Cependant, BART est pré-entraîné en supprimant arbitrairement du texte de l'entrée et en apprenant à le reconstruire, ce qui lui permet de comprendre le contexte à partir des deux directions dans une séquence de texte.

Fine-tuning

Le fine-tuning de BART implique l'entraînement du modèle sur une tâche spécifique en utilisant un petit ensemble de données d'entraînement. Cela permet au modèle d'adapter ses poids pré-entraînés à la tâche spécifique, améliorant ainsi ses performances sur cette tâche. Le fine-tuning

de BART peut être effectué pour une variété de tâches, y compris la génération de texte.

3.4.4 L'évaluation des résultats du modèle

Cette étape permet de mesurer la qualité et la pertinence des résultats produits par le modèle. Quel que soit le type de modèle utilisé, il est important d'évaluer la qualité du texte généré. L'évaluation peut être effectuée de manière subjective en demandant à des experts de noter la qualité des rapports ou de manière objective en utilisant des mesures telles que la précision, la cohérence, la lisibilité, la pertinence, la perplexité, la robustesse, la variabilité, etc. Ces résultats sont utilisés pour ajuster le modèle et améliorer sa performance.

Types d'évaluation

- **Évaluation humaine :** Dans cette technique, un groupe de personnes évalue la qualité du texte généré par le modèle. Les évaluateurs humains peuvent être des experts du domaine ou des personnes ordinaires. Les évaluations sont généralement basées sur des critères tels que la clarté, la cohérence, la pertinence et la fluidité du texte.
- **Évaluation automatique :** Les métriques automatiques peuvent être utilisées pour évaluer la qualité du texte généré. Les métriques d'évaluation automatique les plus courantes incluent la précision, le rappel, et la perplexité. Ces métriques sont souvent utilisées pour évaluer des aspects spécifiques de la génération de textes, tels que la précision de la grammaire ou la cohérence thématique.

Métriques d'évaluation automatique

- **Perplexity (PPL) :**

  La perplexité est une mesure d'évaluation qui est souvent utilisée pour évaluer la qualité des modèles de langue, y compris ceux utilisés pour la génération de textse. Elle est définie comme l'exponentielle de l'entropie croisée moyenne, qui mesure la quantité d'incertitude ou de surprise dans les prédictions du modèle. En d'autres termes, plus la perplexité est faible, mieux le modèle peut prédire la séquence de mots suivante dans un texte donné.

  La perplexité est calculée en comparant les distributions de probabilité prédites par le modèle avec les distributions de probabilité réelles des données de test.

$$PPL = exp(NLL/N) \tag{3.3}$$

  où :

  — NNL : la somme des log-vraisemblances négatives pour toutes les phrases du corpus de test

  — N : le nombre total de mots dans le corpus de test.

En général, pour les modèles de langue, une perplexité inférieure à 100 est considérée comme excellente, tandis qu'une perplexité entre 100 et 200 est considérée comme bonne. Cependant, il convient de noter que la perplexité peut varier considérablement en fonction du domaine de la langue et de la complexité de la tâche de génération de textes.

- **La similarité de cosinus :**

  La similarité de cosinus est une méthode courante d'évaluation de la qualité de la génération de textes. Elle mesure la similitude entre deux vecteurs de mots en calculant l'angle entre eux dans un espace vectoriel. Plus les deux vecteurs sont proches, plus leur similarité de cosinus est élevée, ce qui signifie que le texte généré est plus similaire au texte de référence.

  La similarité de cosinus est souvent utilisée pour comparer les vecteurs de représentation des phrases, qui sont des représentations vectorielles de haute dimension des phrases dans un espace vectoriel. Les vecteurs de représentation des phrases sont souvent générés à l'aide de techniques d'apprentissage non supervisées, telles que Word2Vec ou GloVe.

  Pour évaluer la qualité de la génération de texte à l'aide de la similarité de cosinus, les vecteurs de représentation des phrases sont générés pour le texte de référence et le texte généré, puis la similarité de cosinus entre les deux est calculée. Un score élevé indique que le texte généré est similaire au texte de référence, tandis qu'un score faible indique qu'il y a des différences significatives entre les deux textes.

- **Distinct :**

  Le métrique Distinct (ou Diversité Lexicale) est une métrique d'évaluation utilisée pour mesurer la diversité lexicale des phrases générées par un modèle de langage. Elle évalue le nombre de mots uniques dans les phrases générées par rapport au nombre total de mots dans ces phrases.

$$D = \frac{\text{Nombre de mots distincts générés}}{\text{Nombre total de mots générés}}$$

  Cette formule mesure le pourcentage de mots uniques dans la génération, ce qui permet d'évaluer la variété et la diversité de la génération. Un score Distinct élevé indique une génération plus diversifiée.

3.5 Conclusion

Au cours de ce troisième chapitre, nous avons exposé la méthodologie à suivre pour la génération de données textuelles en présentant le processus, les techniques et modèles de génération automatique utilisés, allant des modèles basés sur les réseaux de neurones récurrents aux modèles pré-entraînés basés sur les Transformers, qui représentent une avancée majeure dans le domaine du NLP. Nous avons également introduit les différents moyens d'évaluation qui seront appliqués pour évaluer les résultats obtenus.

Deuxième partie
État des lieux

Chapitre 4

Présentation de l'organisme d'accueil

4.1 Introduction

Suite à l'exploration des différents concepts théoriques clés liés à notre problématique dans le premier chapitre, nous allons désormais établir le cadre d'application de cette étude en présentant l'organisme d'accueil où s'est déroulé notre stage, à savoir MB inc.

4.2 Présentation de MB inc

MB inc est un cabinet de conseil de renom, situé au **Canada**, spécialisé dans la gestion de réclamations fiscales de crédits d'impôts, qui a été fondé en **2009**. Forte de plus de 11 années d'expérience, l'entreprise offre des services de conseil stratégique de haute qualité aux entreprises innovantes de toutes tailles et de tous secteurs d'activités, notamment dans les domaines de la santé, de l'intelligence artificielle et du secteur manufacturier, pour n'en citer que quelques-uns. Avec un professionnalisme et une expertise inégalés, MB inc est reconnue pour son approche personnalisée, sa connaissance approfondie des réglementations fiscales les plus complexes, ainsi que pour son engagement indéfectible en faveur de l'excellence.

FIGURE 4.1 – Logo de MB inc.

4.3 La fiche technique de MB inc.

FIGURE 4.2 – La fiche technique de MB inc.

4.4 Termes et Jargon

Avant de nous plonger dans notre problématique, nous allons d'abord présenter quelques concepts clés afin de nous familiariser avec le vocabulaire utilisé dans le monde professionnel.

Terme	**Description**
Crédit d'impôt	Contrat qui confère à celle-ci un droit à une somme d'argent, soumis à des conditions de remboursement, plutôt qu'un don gratuit.
Subvention	Aide financière accordée à une entreprise sélectionnée suite à un appel à candidature.
Client	L'entreprise pour laquelle MB inc agit en tant que mandataire pour la réclamation des crédits d'impôts. Le contrat avec le client prend fin après une période déterminée.
Réclamation	L'ensemble d'informations et documents transmis au gouvernement pour une Année Fiscale (AF).

TABLE 4.1 – Tableau récapitulatif des termes et du jargon utilisés.

Différence entre : un crédit d'impôt et une subvention

Crédit d'impôt	Subvention
Crédit accordé à une entreprise si elle respecte les conditions et les cas d'application	Bourse accordée à une entreprise suite à un appel de candidature
Il est un droit et est accordé même si le gouvernement en faillite	Elle est accordée à l'intérieur d'une enveloppe et dans la limite du budget
Il fait partie du rapport d'impôt de l'entreprise produit à chaque fin d'année fiscale	L'appel de projet/candidature la concernant possède une date limite avant laquelle il faut soumettre la candidature
Plus ou moins garanti selon les conditions	Aucune garantie d'obtention même si le projet s'aligne parfaitement avec l'appel de projet
Information soumise réglementée et standardisée avec très peu de changement	Les documents demandés varient beaucoup et ne suivent pas forcément un format précis
Implique un travail sur la fiscalité de l'entreprise et son rapport d'impôt	Demande parfois une prévision financière sur le futur et les retombés du projet en question
Un remboursement d'une partie des dépenses déjà engagées	Un investissement avant le début du projet et ne nécessite pas que des dépenses soient engagées

TABLE 4.2 – La différence entre un **crédit d'impôt** et une **subvention**.

4.5 Mission de MB inc

Les entreprises ne sont pas forcément qualifiées pour monter un dossier de subvention ou remplir des annexes pour un crédit d'impôt. Elles préfèrent, donc, faire appel à des spécialistes pour les aider à garantir l'obtention des crédits et surtout de la maximisation des remboursements qui leur correspondent. C'est là qu'intervient MB inc. Sa mission principale est d'accompagner ces entreprises stratégiquement dans leurs recherches de financement d'argent public, en vue d'assurer la pérennité de leurs activités.

4.5.1 Les objectifs visés

Les objectifs visés sont notamment de :

- Contrôler stratégiquement les coûts et risques financiers associés aux projets des entreprises de développement en assurant la pérennité des remboursements d'argent public auxquels elles ont droit.
- Transformer l'expérience de réclamation de crédits d'impôts en une opération simplifiée et optimisée qui se scindera au modèle d'affaire de l'entreprise comme une source de financement extrêmement rentable qu'on ne peut contourner.

Pour y arriver, MB inc propose un accompagnement stratégique adapté aux objectifs et aux besoins financiers du moment de l'entreprise, basé sur une vision et une approche de partenariat d'affaires, qui leur assure un processus fluide, peu contraignant, et financièrement maximisé. [1].

4.6 Les services de MB inc

MB inc offre un modèle d'encadrement basé sur un processus de cinq étapes [1] :

- Établir la stratégie d'accompagnement optimal en fonction des besoins ponctuels de l'entreprise en financement
- Identifier les dépenses admissibles aux différents crédits d'impôts en fonction des critères établis par l'ARC, du MRQ et d'autres organismes.
- Analyser vos projets et activités admissibles, optimiser stratégiquement les montants réclamés pour vos projets.
- Soumettre une demande conforme aux règles des différents programmes.
- Suivre la progression de votre dossier et contrôler le processus en situation de demande d'Audit de la part du gouvernement.

Ses principaux champs d'expertise sont :

- **Crédit RS&DE (Recherche scientifique et développement expérimental) :** 100% de taux de réussite.
- **Crédit CDAE (Développement des affaires électroniques) :** MB inc maîtrise parfaitement le processus.
- **Crédit CTMM (Production de titres multimédias) :** MB inc soumet près de 25% des demandes totales produites annuellement à IQ.
- **PARI (Aide à la recherche industrielle) :** L'expertise de MB inc ouvre aux entreprises des portes comme une PME innovante.

Champs	Objectif du programme	Avantages	Les travaux de recherche sont-ils admissibles ?
Crédit RS&DE	Inciter les entreprises canadiennes de toutes tailles et de tous secteurs à mener des activités d'innovations et de développement économique.	Ce programme est partie intégrante de la loi de l'impôt et se traduit en un droit à : • un remboursement d'impôts. • une déduction d'impôt sur le revenu • un crédit d'impôt à l'investissement (CII).	• **De la recherche pure :** Le projet apporte de nouvelles connaissances scientifiques sans vision d'application concrète. • **De la recherche appliquée :** Le projet apporte de nouvelles connaissances scientifiques mais en vue d'une application concrète. • **Le développement expérimental :** Le projet a comme objectif une avancée technologique.
Crédit CDAE	C'est un programme provincial. Il encourage les entreprises dont les activités sont concentrées autour de la conception de systèmes informatiques ou de l'édition de logiciels. L'objectif de Revenu Québec est de stimuler l'emplois dans le domaine des technologies de l'information.	Si l'entreprise rencontre les critères d'éligibilité, elle pourra bénéficier d'un remboursement du salaire de la main-d'œuvre admissible. Ce remboursement sera maintenu tant et aussi longtemps que votre société rencontre les critères d'éligibilité.	• **Premier critère :** L'entreprise se situe au Québec et y pratique ses activités. • **Deuxième critère :** L'entreprise doit engendrer 75% et plus de ses activités dans le secteur des technologies de l'information. • **Troisième critère :** L'entreprise doit obtenir une attestation d'admissibilité émise par Investissement Québec. • **Quatrième critère :** Votre entreprise doit employer des salariés admissibles.

Crédit CTMM	Les titres multimédias qui favorisent l'interactivité. Afin d'être désigné comme tel, un titre multimédia doit présenter au moins trois des quatre critères suivants : un texte, du son, des images fixes et des images animées.	L'entreprise en étant admissible au crédit d'impôt pour la production de titres multimédias pourra profiter d'un crédit d'impôt remboursable sur une partie des salaires et paiements de sous-traitants ayant travaillé sur le projet.	• **Premier critère :** Votre entreprise doit exercer ses activités au Québec et verser les salaires admissibles à des employés œuvrant au Québec. • **Deuxième critère :** L'entreprise doit détenir une attestation d'admissibilité émise par Investissement Québec pour l'année de réclamation.
PARI	Le programme d'aide à la recherche industrielle consiste en des subventions directes. Ce programme proposé par le Conseil national de recherches Canada (CNRC) concerne les petites et moyennes entreprises.	Si Le projet de recherche et développement est admissible, l'entreprise peut donc demander des subventions sur les salaires de vos employés et sur vos factures de sous-traitants engagés pour le projet.	• **Premier critère :** L'entreprise doit être située au Canada, avoir moins de 500 employés à temps plein et être à but lucratif. • **Deuxième critère :** L'entreprise doit élaborer un projet innovant qui tend à répondre à certaines incertitudes technologiques. • **Troisième critère :** L'entreprise doit démontrer qu'un profit est possible via cette innovation technologique.

TABLE 4.3 – Les champs d'expertise de MB inc [1].

4.7 Ses Valeurs

MB inc ne se contente pas uniquement d'assister les entreprises dans leur quête de financement public, elle est également guidée par un ensemble de valeurs qui orientent ses actions et la distinguent de ses concurrents, notamment :

- **L'expertise :** Grâce à son savoir-faire en matière de financement, MB inc a su se faire un nom auprès des autorités gouvernementales en charge de l'examen de ses dossiers. De plus, l'entreprise a su tisser des liens solides avec diverses institutions de financement, telles que le CNRC PARI, D3 de Concordia, la RBC, la BNC, RD Capital, ainsi que de nombreux autres organismes de financement dédiés aux entrepreneurs.
- **Le pouvoir de propulser les ambitions des entreprises :** Cela implique de fournir un accompagnement personnalisé, des conseils avisés et un soutien constant pour aider les entreprises à identifier et à poursuivre les opportunités de croissance et de développement qui leur permettront de réaliser leurs ambitions à long terme. En d'autres termes, MB inc

s'engage à travailler en étroite collaboration avec ses clients pour les aider à atteindre leurs objectifs de manière efficace et efficiente.

- **Soutenir stratégiquement l'économie locale innovante :** Ceci reflète son engagement à aider les entreprises locales à se développer et à prospérer grâce à l'innovation. En offrant un soutien stratégique aux entreprises locales, MB inc contribue à l'économie locale en stimulant la croissance et la création d'emplois. Cette valeur montre également que l'entreprise est fière de sa communauté et de son rôle dans son développement économique. En fin de compte, cette valeur montre l'engagement de MB inc envers la durabilité à long terme de l'économie locale en favorisant l'innovation et en soutenant les entreprises locales.
- **Partager avec les entreprises les risques et bienfaits liés au financement public :** Ceci implique que MB inc considère ses clients comme des partenaires et s'engage à travailler en étroite collaboration avec eux. Cela signifie que l'entreprise est prête à partager les risques et les bénéfices avec ses clients lorsqu'elle les aide à obtenir un financement public. MB inc s'efforce de garantir que ses clients bénéficient des avantages financiers tout en minimisant les risques associés à la recherche de financements publics. Cette approche renforce la confiance entre MB inc et ses clients et permet une collaboration fructueuse et durable.
- **Agir comme leur partenaire :** Cette valeur reflète leur engagement à travailler en étroite collaboration avec les entreprises clientes, en agissant comme un véritable partenaire pour atteindre leurs objectifs. Cela implique d'écouter attentivement les besoins des clients, de comprendre leur vision et leur stratégie, et de fournir des solutions adaptées à leur situation spécifique. MB inc s'engage également à établir une relation de confiance à long terme avec ses clients, en agissant de manière éthique et transparente à chaque étape de leur collaboration. En agissant comme leur partenaire, MB inc est déterminée à aider les entreprises à réussir et à réaliser leur potentiel de croissance.

4.8 Clients & Partenaires

FIGURE 4.3 – Clients et Partenaires de MB inc.

4.9 Structure organisationnelle et Hiérarchie

L'équipe de MB inc est constituée d'experts techniques et scientifiques ainsi que d'experts fiscaux, qui possèdent une compréhension approfondie des défis de financement rencontrés par les entrepreneurs.

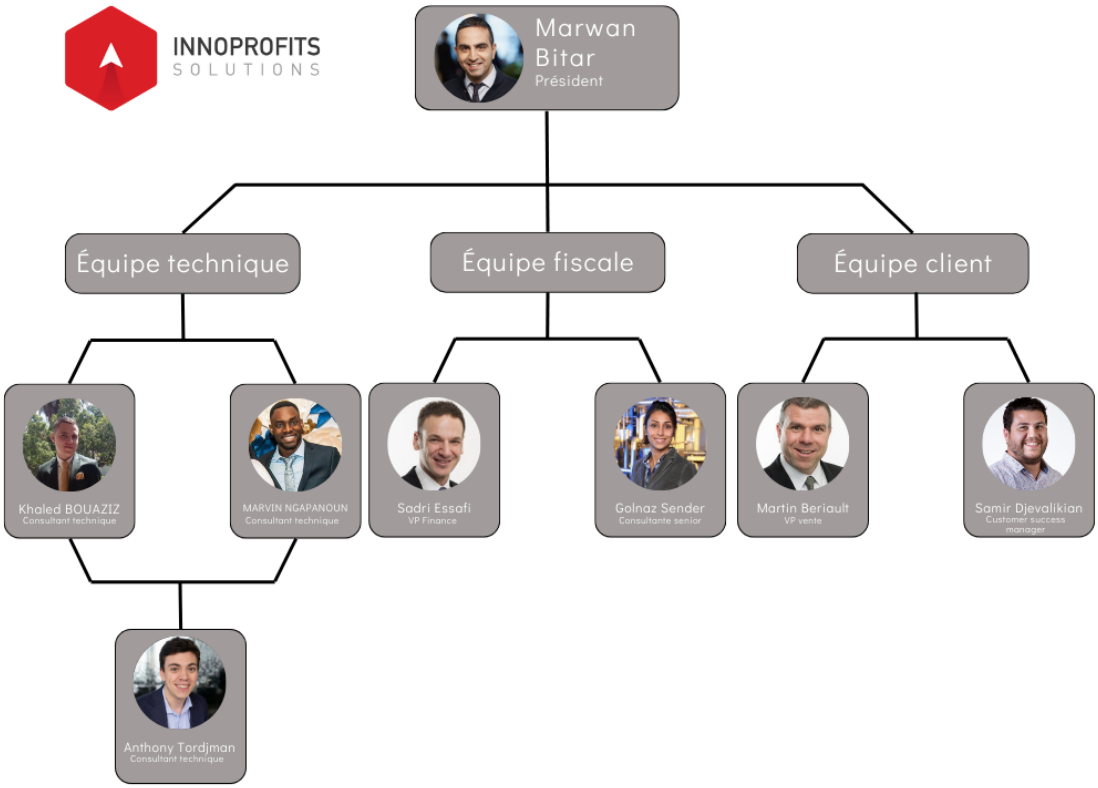


FIGURE 4.4 – L'organigramme de MB inc.

4.10 Conclusion

Au cours de ce chapitre, nous avons pris le temps de présenter l'entreprise MB inc Solutions, son objectif et les services qu'elle propose. Nous avons également expliqué certains concepts clés et exposé le vocabulaire utilisé au sein de l'entreprise. Ensuite, nous avons mis en avant les valeurs suivies par l'entreprise tout au long de son parcours. De plus, nous avons présenté la hiérarchie et la structure organisationnelle de MB inc à travers son organigramme. Pour finir, nous avons mentionné quelques clients pour lesquels MB inc a mené des projets afin de fournir ses multiples offres et services.

Chapitre 5

Étude de l'existant (Le système actuel au sein de MB inc)

5.1 Introduction

L'étude de l'existant est une étape cruciale dans notre projet d'amélioration du processus de rédaction des rapports techniques au sein de MB inc. Dans ce chapitre, nous examinerons en détail le système actuel en place dans notre entreprise. Nous aborderons la structure et les caractéristiques des rapports techniques, ainsi que les sources utilisées pour les rédiger. Nous explorerons également le processus existant au sein de MB inc, en mettant l'accent sur le rôle des différents utilisateurs et en décrivant le contexte dans lequel cette étude est menée. Enfin, nous soulèverons les critiques du fonctionnement actuel, mettant en évidence les problématiques et les opportunités d'amélioration que nous cherchons à résoudre.

5.2 Les rapports techniques

Les rapports techniques sont des documents écrits qui présentent les résultats de travaux scientifiques ou techniques spécifiques. Ils fournissent des informations détaillées sur les méthodes utilisées, les résultats obtenus, les analyses effectuées, les conclusions tirées, les recommandations formulées et les références utilisées. Dans notre contexte d'étude, ces rapports sont utilisés pour communiquer les résultats de recherche et les informations techniques des clients aux autorités gouvernementales afin de solliciter un financement. Dans la suite, nous allons explorer la structure, les caractéristiques et les sources utilisées pour la rédaction de ces rapports chez MB inc.

5.2.1 La structure et les caractéristiques des rapports techniques

Chez MB inc, les rapports techniques sont élaborés en se basant sur la RS&DE, qui correspond à une **investigation** ou **recherche systématique** d'ordre **scientifique** ou **technologique**, effectuée par voie d'**expérimentation** ou d'**analyse**.

Ces rapports obéissent à une structure clairement définie, qui se compose de trois parties essentielles. Chaque partie doit répondre à une question représentant l'idée générale de la section à rédiger. Le tableau ci-dessous résume en détail chaque partie :

Partie N°	Nom	Nombre de mots maximal	Titre (La question à poser)
1	Incertitudes et/ou Obstacles technologiques	350	A. Quels obstacles technologiques avez-vous dû surmonter pour réaliser les avancements VISÉS ? (35 lignes maximum) - projet de développement expérimental.
2	Contenu scientifique et technique	700	B. Quels travaux avez-vous effectués au cours de l'année d'imposition pour surmonter les obstacles technologiques ? (70 lignes maximum) – projet de Développement expérimental
3	Avancements technologiques	350	C. Quels Avancements technologiques avez-vous essayé de réaliser ? (35 lignes maximum) – projet de développement expérimental

TABLE 5.1 – Tableau résumant la structure globale d'un rapport technique.

Abordons chaque partie à part :

Partie 01 : Incertitudes et/ou Obstacles technologiques

Dans un rapport technique, la première partie consiste à aborder les incertitudes et/ou les obstacles technologiques rencontrés lors du projet de recherche et développement. Ces incertitudes peuvent être liées à la faisabilité technique du projet, à la complexité des technologies utilisées, ou encore aux risques associés à la mise en place de la solution proposée. Il est donc primordial de décrire ces incertitudes de manière claire et précise, en utilisant un vocabulaire d'ingénierie adéquat et en expliquant les raisons pour lesquelles ces incertitudes ont été rencontrées. En identifiant et en abordant ces incertitudes dès le début du rapport, les lecteurs du rapport pourront mieux comprendre les défis rencontrés dans la réalisation du projet et les solutions proposées pour les surmonter. Cela permet également de démontrer la rigueur scientifique et technique de l'équipe en charge du projet.

Incertitude	- Impossible de prédire l'atteinte d'un résultat. - Impossible de prédire la façon d'atteindre un résultat.
Obstacle	- Limite de la base de connaissances.

TABLE 5.2 – La différence entre **Incertitude** et **Obstacle**.

Partie 02 : Contenu scientifique et technique

Le contenu scientifique et technique dans un rapport technique consiste à décrire les travaux effectués pour surmonter les obstacles et les incertitudes technologiques mentionnés dans la première partie du rapport. Cette partie du rapport doit être rédigée avec créativité pour raconter une histoire cohérente et claire qui met en évidence les travaux effectués pour atteindre les objectifs de la recherche. Il est important d'aborder trois volets essentiels dans cette partie, à savoir **la formulation des hypothèses**, **la vérification par des expérimentations et des analyses**, et **les conclusions** qui en découlent.
Pour formuler des hypothèses, il faut utiliser des connaissances techniques et scientifiques pour établir des hypothèses de travail qui serviront de base aux expérimentations et analyses. Ensuite, des expérimentations doivent être effectuées pour vérifier les hypothèses formulées. Ces expérimentations peuvent être de différentes natures selon le domaine d'application, comme les travaux de génie, la conception, la recherche opérationnelle, l'analyse mathématique, la recherche psychologique, les essais, la collecte de données ou encore la programmation informatique.
Une fois les expérimentations et les analyses terminées, il est important de présenter les conclusions tirées de ces travaux. Ces conclusions doivent être claires, précises et justifiées par les résultats obtenus.

Partie 03 : Avancements technologiques

La troisième partie d'un rapport technique est consacrée à la présentation des avancements technologiques réalisés dans le cadre du projet en question. Cette partie doit mettre en avant les progrès technologiques obtenus, ainsi que la compréhension des relations scientifiques qui ont permis de les atteindre. L'objectif principal est de montrer comment le projet a repoussé l'état des connaissances dans le domaine, en créant ou en améliorant un produit ou un procédé existant. Cette partie suit une structure bien définie, qui peut être adaptée en fonction des deux premières parties du rapport.
Les avancements technologiques doivent être décrits de manière précise et détaillée, en expliquant les innovations qui ont été apportées et les résultats obtenus. Il est également important de souligner les implications pratiques et potentielles des avancements technologiques, comme leur impact sur l'industrie, l'environnement ou la société en général. Enfin, il est recommandé d'utiliser des exemples concrets pour illustrer les avancements technologiques décrits dans cette partie du rapport.

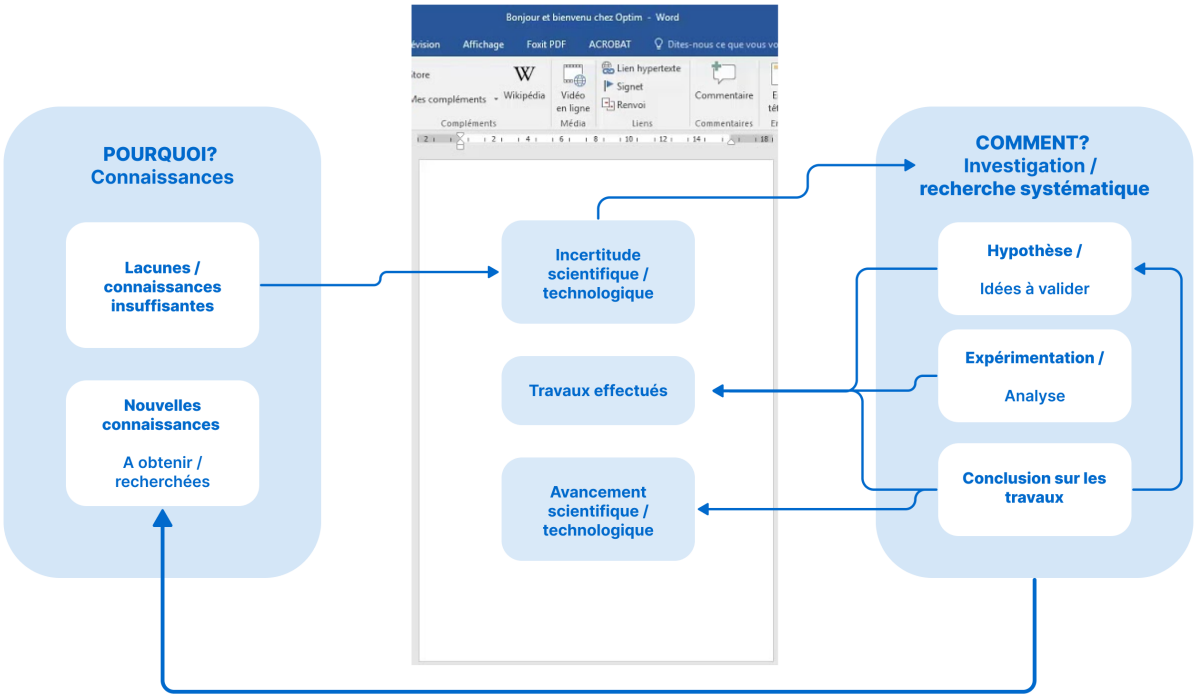


FIGURE 5.1 – La structure d'un rapport technique.

5.2.2 Les sources des rapports techniques

Les sources pour la rédaction des rapports techniques peuvent varier en fonction de la nature du projet et des travaux effectués. Cependant, certaines sources sont généralement utilisées pour garantir la qualité et la crédibilité des résultats présentés dans le rapport. Parmi ces sources, on peut citer les publications scientifiques, les brevets, les standards et normes techniques, les rapports précédemment publiés, les entrevues avec des experts et les observations sur le terrain. Les rédacteurs doivent également prendre en compte les sources d'information nécessaires pour répondre à toutes les questions et incertitudes identifiées dans les parties précédentes du rapport technique. En général, il est important que toutes les sources utilisées soient dûment citées et référencées pour permettre une traçabilité et une vérification ultérieure des résultats présentés.

La prise de notes lors de la rencontre avec le client est également une étape essentielle dans la rédaction d'un rapport technique chez MB inc. Elle permet de recueillir toutes les informations nécessaires pour bien comprendre les objectifs du client, les spécifications techniques requises, les contraintes, les ressources disponibles, etc. Les notes prises lors de la rencontre avec le client sont ensuite utilisées pour guider la rédaction du rapport technique et s'assurer que toutes les exigences sont bien prises en compte. Il est important que les notes soient précises, claires et bien organisées pour faciliter la rédaction du rapport technique et éviter les erreurs ou les omissions.

5.3 Le processus de rédaction des rapports techniques

Au sein de ce processus, **la rédaction des rapports techniques** revêt une importance capitale. Le rôle clé du conseiller technique prend tout son sens à ce stade. Il est chargé de planifier

et d'organiser une rencontre technique dédiée à l'examen approfondi des détails de chaque projet. Pendant cette rencontre, l'objectif principal est de déterminer avec précision le nombre exact de rapports techniques à rédiger. Le conseiller technique joue un rôle essentiel en **prenant des notes précieuses** qui serviront de base solide lors de la rédaction des rapports. Ces notes qui doivent être méticuleusement prises seront ensuite utilisées pour structurer et formuler de manière précise le contenu des rapports techniques. Une fois rédigés, ces rapports seront soumis à un processus de correction et de validation rigoureux supervisé par le superviseur technique. Cette étape assure la qualité et la rigueur nécessaires pour garantir des rapports techniques fiables et conformes aux normes établies.

FIGURE 5.2 – Le processus de rédaction.

5.3.1 La critique du fonctionnement actuel

Suite à une analyse approfondie des différentes tâches et activités réalisées au sein de MB inc, nous avons observé que le processus de prise de notes s'effectuait de manière désorganisée et chaotique. Il était dépourvu de toute structure cohérente, amenant parfois les consultants à consigner des éléments qu'ils pourraient eux-mêmes ne pas comprendre par la suite. Ce mode de fonctionnement peut induire une multitude de problèmes, comme la difficulté à retrouver des informations spécifiques, la perte de données précieuses et le gaspillage de temps précieux en déchiffrage et réorganisation.

Concernant la rédaction des rapports techniques, étape cruciale de l'activité globale de MB inc, elle s'effectue actuellement manuellement, sans aucune forme d'automatisation. Ce procédé entraîne une dépense de temps significative et exige un effort conséquent de la part des équipes spécialisées. De surcroît, cette tâche nécessite l'intervention de multiples acteurs, incluant les consultants et les superviseurs techniques, avant d'obtenir une validation finale. Cette multiplicité des intervenants amplifie les risques de divergences, compliquant et prolongeant la tâche en question. Par conséquent, cela peut occasionner des retards dans le travail des autres équipes et freiner la progression générale du processus.

L'impact de cette rédaction manuelle des rapports techniques ne se limite pas à la simple perte de temps. Elle peut également entraver la demande de financement, retarder la réalisation des projets clients et impacter négativement l'ensemble du processus. Il est à souligner qu'un consultant technique a généralement besoin d'une journée entière, voire plus, pour rédiger un seul rapport manuellement.

Il convient également de souligner que d'autres facteurs peuvent influencer la durée globale de la rédaction des rapports. Parmi ceux-ci figurent la récupération des documents, la structuration des idées et des notes précédemment prises, ainsi que la saisie des informations. Ces processus constituent autant d'éléments supplémentaires qui augmentent la complexité et la longueur de la tâche.

5.4 L'identification des opportunités d'amélioration

L'analyse de cette situation révèle plusieurs opportunités d'amélioration qui pourraient optimiser le processus de rédaction des rapports techniques et améliorer l'efficacité globale de MB inc.

- **Automatisation et Intégration de l'IA :** L'adoption de technologies d'intelligence artificielle pourrait offrir des possibilités considérables pour améliorer l'efficacité du processus. Par exemple, des outils basés sur l'IA pourraient être utilisés pour structurer et synthétiser les notes prises lors des rencontres techniques, ou pour générer automatiquement des descriptions techniques des projets à partir de ces notes.
- **Normalisation de la prise de notes :** En établissant des directives claires et uniformes pour la prise de notes, nous pourrions assurer une meilleure cohérence dans les informations recueillies et faciliter leur compréhension et leur utilisation ultérieure.
- **Formation et support continu :** En fournissant aux consultants une formation adéquate et un soutien continu, nous pourrions améliorer leur capacité à réaliser efficacement leurs tâches. Cela pourrait comprendre une formation sur les meilleures pratiques en matière de prise de notes et de rédaction de rapports, ainsi que l'utilisation des outils d'automatisation.
- **Optimisation de la collaboration :** L'amélioration de la collaboration entre les différents acteurs impliqués dans la rédaction des rapports pourrait aider à réduire les erreurs et incohérences. Cela pourrait impliquer la mise en place d'un système plus efficace de révision et validation des rapports.

Ces opportunités, si elles sont mises en œuvre, pourraient aider à résoudre les défis actuellement rencontrés par MB inc et améliorer l'efficacité de leur processus de rédaction de rapports techniques.

5.4.1 Énoncé de la problématique

La problématique principale réside dans l'inefficacité actuelle du processus de rédaction des rapports techniques chez MB inc, qui peut être décomposée en deux problématiques distinctes mais interconnectées :

- Comment peut-on standardiser et structurer les méthodes de prise de notes pour améliorer la qualité, la clarté, et l'efficacité du processus de rédaction des rapports techniques, tout en réduisant la confusion et la perte d'informations précieuses ?
- Comment l'automatisation et l'intelligence artificielle peuvent-elles être utilisées pour surmonter les défis actuels liés à l'efficacité de la rédaction de rapports techniques chez MB inc ?

5.5 Conclusion

L'étude de l'existant au sein de MB inc nous a permis de mieux comprendre le système actuel de rédaction des rapports techniques et son impact sur nos opérations. Nous avons examiné en détail la structure et les caractéristiques des rapports techniques, ainsi que les sources utilisées pour les rédiger. De plus, nous avons analysé le processus existant au sein de notre entreprise, en identifiant les différents utilisateurs impliqués et en décrivant le contexte dans lequel cette étude est menée.

Cependant, cette analyse a également mis en évidence des problématiques et des limitations du fonctionnement actuel. Les retards, la charge de travail excessive pour nos experts, le risque d'incohérences entre les différentes versions des rapports sont autant de défis que nous devons relever. C'est dans ce contexte que nous avons identifié des opportunités d'amélioration et que nous avons entrepris ce projet visant à améliorer la prise de notes et à intégrer l'IA pour l'aide à la rédaction des rapports techniques en générant des descriptions techniques pertinentes pour les projets.

Dans les prochains chapitres, nous explorerons les différentes solutions envisagées pour relever ces défis et améliorer notre processus. Nous examinerons les avantages et les contraintes de la génération automatique des descriptions techniques des projets, ainsi que les étapes clés de sa mise en place. Notre objectif ultime est de fournir des descriptions techniques de haute qualité de manière plus rapide et plus efficace pour les projets, tout en libérant nos experts pour des tâches à plus forte valeur ajoutée.

Troisième partie
Conception et Réalisation de la solution

Chapitre 6

Conception de la solution

6.1 Introduction

Après une analyse approfondie des besoins de notre entreprise et des procédures de travail existantes, nous avons opté pour une manière plus structurée et efficiente pour la prise de notes lors des rencontres techniques avec les clients, et nous avons identifié une solution qui repose sur l'IA pour automatiser le processus de rédaction. Dans les prochaines sections, nous aborderons la conception de cette solution en rappelant les besoins initiaux et les objectifs que nous visons. Nous détaillerons également les différentes étapes nécessaires à sa mise en œuvre, tout en présentant notre vision de son application concrète.

6.2 Rappel sur le besoin

Le besoin émanant de MB inc est clairement défini : optimiser le processus de génération de rapports techniques afin d'améliorer l'efficacité, la précision et la cohérence de ces documents. Cela comprend la structuration et l'organisation de la prise de notes lors des rencontres techniques, ainsi que l'automatisation de la rédaction des rapports. La standardisation de ces procédures vise à réduire le temps passé sur ces tâches, minimiser les erreurs, et éliminer les retards qui pourraient affecter le processus de financement et la réalisation des projets des clients. En répondant à ce besoin, MB inc cherche à améliorer non seulement sa productivité interne, mais aussi la qualité du service fourni à ses clients. En fin de compte, l'objectif est de rendre le processus plus fluide et efficace, tout en conservant un haut niveau de précision et de qualité dans la rédaction des rapports techniques.

6.3 La solution proposée

Pour répondre à la première problématique liée à la prise de notes non structurée, nous proposons **un système de prise de notes** en utilisant des **tags**. Cette solution consiste à utiliser un ensemble de balises préétablies qui seront utilisées par le consultant lors de la prise de notes. Ces tags permettront de classer et d'organiser l'information en temps réel, ce qui facilitera grandement la structuration des idées pour la rédaction du rapport technique. L'utilisation de tags rendra la tâche moins chaotique, améliorera la cohérence des notes prises et aidera à mieux saisir l'essence de la discussion lors de la rencontre avec le client. En outre, cela diminuera le risque de perdre des informations cruciales et facilitera le processus de relecture et de compréhension de notes.

Pour la deuxième problématique, qui concerne l'automatisation de la rédaction des rapports techniques, nous suggérons d'utiliser un **modèle pré-entrainé** de génération de textes et l'adapter pour générer des descriptions techniques de projets à partir des notes structurées obtenues grâce au système de tags, en appliquant une technique d'apprentissage automatique appelée **Transfer Learning**. Cette approche permettrait non seulement de gagner du temps et de réduire les efforts manuels, mais aussi d'améliorer la cohérence et la qualité des rapports produits.

Pour mettre en œuvre la solution proposée, plusieurs étapes clés doivent être suivies :

- **Collecte des données :** La première étape consiste à collecter les données pertinentes pour notre tâche spécifique. Il s'agit de textes, de documents, de corpus linguistiques et de tout autre type de données nécessaires à notre projet.
- **Sélection du modèle pré-entraîné :** Une fois les données collectées, il est nécessaire de choisir un modèle pré-entraîné basé sur les Transformers qui est adapté à notre tâche spécifique.
- **Adaptation du modèle :** Dans cette étape, nous procédons au fine-tuning du modèle pré-entraîné sélectionné. Cela implique d'ajuster les poids et les paramètres des couches supérieures du modèle pour le faire correspondre à notre tâche spécifique.
- **Prétraitement des données :** Avant d'entraîner le modèle, il est souvent nécessaire de prétraiter les données en les nettoyant, en les normalisant et en les structurant de manière appropriée. Cela inclut des étapes telles que la tokenization, la suppression des stopwords, la lemmatisation, etc.
- **Entraînement du modèle :** Une fois que les données ont été préparées et que le modèle a été adapté, il est temps de procéder à l'entraînement du modèle. Cette étape consiste à présenter les données d'apprentissage au modèle et à ajuster ses poids en utilisant des techniques d'optimisation.
- **Évaluation et ajustement :** Après l'entraînement, il est essentiel d'évaluer les performances du modèle sur un ensemble de données de test. Cela permet de mesurer la qualité des prédictions du modèle et d'identifier d'éventuels problèmes ou lacunes. Si nécessaire, des ajustements supplémentaires peuvent être apportés au modèle pour améliorer ses performances.
- **Déploiement et utilisation :** Une fois que le modèle a été entraîné et évalué avec succès, il peut être déployé dans un environnement de production pour être utilisé dans des scénarios réels.

Ainsi, le système de prise de notes agit comme un pont entre les utilisateurs et le modèle de génération de texte, facilitant la collecte de données, l'organisation de l'information et l'accès aux résultats du modèle. En intégrant étroitement ce système à notre solution, nous assurons une expérience utilisateur fluide et efficace tout au long du processus, de **la collecte des données initiales** à **l'utilisation finale** des résultats générés par le modèle.

6.4 L'idée d'application envisagée

Nous n'envisageons pas de générer un rapport intégral en une seule étape, mais plutôt de générer des descriptions techniques dans chaque partie du rapport de manière individuelle pour le projet entrepris.

Pour cela, il est nécessaire de disposer d'un ensemble de données spécifique pour chacune des trois parties du rapport. La formation du modèle pour générer les descriptions techniques pour chacune des parties requiert des entrées distinctes. Par conséquent, pour former le modèle à produire trois sections différentes, il nous faut trois ensembles de données distincts, chacun associé à une partie spécifique. L'ensemble de données devrait comporter des colonnes servant d'entrées, ainsi qu'une colonne faisant office de sortie. Nous explorerons tous ces aspects en profondeur dans le chapitre consacré à la mise en œuvre de la solution.

6.5 Conclusion

En conclusion, ce chapitre a présenté notre approche pour répondre aux besoins de MB inc en matière de génération des descriptions techniques de projets. Nous avons proposé une solution qui combine une structuration plus efficace de la prise de notes et l'application de l'intelligence artificielle pour automatiser le processus de rédaction. Cette solution vise à améliorer l'efficacité, la précision et la cohérence des rapports techniques, tout en réduisant le temps et les efforts nécessaires pour leur production. Dans le prochain chapitre, nous aborderons la mise en œuvre de cette solution, en détaillant les étapes nécessaires pour la réaliser.

Chapitre 7

Réalisation de la solution

7.1 Introduction

Ce chapitre se concentre sur la mise en œuvre concrète de la solution développée dans ce mémoire. Il détaille les différentes étapes clés de la réalisation, depuis la construction du dataset jusqu'au déploiement de l'API et la création de l'interface utilisateur. Les sous-sections suivantes présenteront en détail chaque étape, en mettant l'accent sur les choix méthodologiques, les techniques utilisées et les résultats obtenus.

7.2 Construction du dataset utilisé

Pour réaliser notre solution, il était crucial d'avoir un dataset complet regroupant tous les rapports techniques rédigés chez MB inc jusqu'à ce jour. Cependant, ce dataset n'était pas disponible au sein de l'entreprise. Les rapports techniques étaient plutôt stockés et répartis dans divers dossiers sous format word (.docx).

Ainsi, nous avons entrepris la création de ce dataset à partir de zéro. Détaillons les différentes étapes et tâches que nous avons effectuées pour mener à bien cette tâche.

7.2.1 Identification et collecte des rapports techniques

Nous avons effectué une exploration minutieuse afin de collecter tous les rapports historiques. Cette étape était essentielle pour garantir l'inclusion de tous les documents pertinents.

7.2.2 Nettoyage et normalisation des rapports

Nous avons effectué un processus rigoureux de nettoyage et de normalisation des rapports afin de garantir la qualité des données. Les étapes impliquées étaient les suivantes :

- Suppression des caractères non pertinents.
- Correction des erreurs dans les textes.
- Traduction des rapports écrits en anglais vers le français. Nous avons utilisé un outil en ligne de traduction des fichiers pour obtenir des rapports normalisés et dans la même langue.
- Normalisation des titres de chaque partie d'un rapport pour assurer leur cohérence et leur uniformité. Cela permettra par la suite d'extraire chaque partie de tous les rapports en les identifiant par des titres normalisés.

7.2.3 Centralisation des rapports collectés

Nous avons procédé au transfert des rapports vers Google Drive pour faciliter leur stockage et leur accessibilité. Cette étape nous a permis de charger les rapports dans un emplacement centralisé, offrant ainsi une meilleure organisation et une gestion plus efficace des documents. De plus, en utilisant Google Drive, nous avons bénéficié de fonctionnalités avancées telles que la possibilité de partager et de collaborer facilement sur les rapports avec d'autres membres de l'équipe. Cela a contribué à améliorer la productivité et la collaboration dans le cadre de notre projet.

7.2.4 Extraction des contenus des rapports

Une fois que nous avons traité, nettoyé et chargé les rapports dans un dossier dédié sur Google Drive, nous avons entrepris l'extraction de leur contenu en utilisant l'API de Google Drive. Cette méthode nous a offert la possibilité d'accéder aux fichiers stockés sur Google Drive et d'extraire leur contenu de manière automatisée. Les étapes suivies pour réaliser cette extraction sont :

- **Configuration de l'API de Google Drive :**
  Nous avons créé un projet dans la console de développement de Google Cloud et activé l'API de Google Drive. Cela nous a permis d'obtenir les clés d'authentification nécessaires pour accéder aux fichiers dans Google Drive.
- **Autorisation d'accès aux fichiers Google Drive :**
  Nous avons configuré l'autorisation d'accès à Google Drive en utilisant le protocole OAuth. Cela nous a permis d'obtenir un jeton d'accès valide pour accéder aux fichiers Google Drive depuis notre application.
- **Récupération de la liste des fichiers Word dans Google Drive :**
  À l'aide de l'API de Google Drive, nous avons récupéré la liste des fichiers Word présents dans un dossier spécifique (identifié par son URL) de Google Drive. Cette étape nous a permis d'identifier les rapports techniques à extraire.
- **Conversion des fichiers Word en texte :**
  En utilisant des bibliothèques de traitement de texte telles que python-docx, nous avons converti les fichiers Word téléchargés en texte brut. Cette étape nous a permis d'obtenir le contenu des rapports techniques sous une forme exploitable. Nous avons extrait le contenu complet de chaque rapport, puis identifié et extrait chaque partie individuelle en utilisant les titres normalisés des parties des rapports.
- **Structuration des données :**
  Après avoir extrait le contenu des fichiers Word, nous avons structuré les données en vue de les enregistrer dans un dataset. Cela implique la création de champs pour chaque type de

```python
# Boucler sur tous les fichiers .docx
for item in files:
    file_id = item['id']
    file_name = item['name']
    #request = service.files().export_media(fileId=file_id, mimeType='text/plain')
    request = service.files().get_media(fileId=file_id)
    file_content = io.BytesIO()
    downloader = MediaIoBaseDownload(file_content, request)
    done = False
    while done is False:
        status, done = downloader.next_chunk()

    #converter le io.BytesIO() to document
    document = Document(file_content)
    data = ""
    for p in document.paragraphs:
        data += p.text
        #data +="\n"
    #print(data0)

    #converter le io.BytesIO() to document   (partie01)
    incertitudes =""
    bol_incertitudes=False
    for p in document.paragraphs:
        if p.text=="A.¬Quels obstacles technologiques avez-vous dû surmonter pour réaliser les avancements VISÉS ? (35 ligne
            bol_incertitudes=True
        if p.text=="B.¬Quels travaux avez-vous effectués au cours de l'année d'imposition pour surmonter les obstacles techn
            bol_incertitudes=False
        if bol_incertitudes==True :
            incertitudes+=p.text

    #converter le io.BytesIO() to document   (partie02)
    travaux =""
    bol_travaux=False
    for p in document.paragraphs:
        #condition de demarage
        if p.text=="B.¬Quels travaux avez-vous effectués au cours de l'année d'imposition pour surmonter les obstacles techn
            bol_travaux=True
        #condition de fin
        if p.text=="C.¬Quels Avancements technologiques avez-vous essayé de réaliser ? (35 lignes maximum) - projet de déve
            bol_travaux=False
        if bol_travaux==True :
            travaux+=p.text

    #converter le io.BytesIO() to document   (partie03)
    avancements =""
    bol_avancements=False
    for p in document.paragraphs:
        if p.text=="C.¬Quels Avancements technologiques avez-vous essayé de réaliser ? (35 lignes maximum) - projet de déve
            bol_avancements=True
        if bol_avancements==True :
            avancements+=p.text
```

FIGURE 7.1 – Conversion des fichiers Word en texte et extraction du contenu.

données extraites (Nom du fichier, Contenu complet, Partie 01 (obstacles et incertitudes), Partie 02 (Travaux), Partie 03 (Avancements Technologiques)).

- **Stockage du contenu extrait :**
  Le contenu extrait des rapports techniques a été stocké dans un dataset au format CSV. Cela nous a permis de gérer et de manipuler facilement les données extraites pour les étapes ultérieures. De plus, nous avons structuré tous les rapports existants et les avons regroupés dans un emplacement centralisé.

Grâce à l'utilisation de l'API de Google Drive, nous avons pu automatiser l'extraction du contenu des rapports techniques stockés sur cette plateforme. Cette approche nous a permis de récupérer rapidement et efficacement le contenu des fichiers Word, et de collecter les données brutes nécessaires à la création de notre dataset.

7.2.5 Extraction d'informations pertinentes

Dans le but d'enrichir notre dataset, nous avons envisagé d'effectuer une extraction d'informations supplémentaires pertinentes :

- **Enrichissement avec les informations de "Client" et "Année" :** Les titres des fichiers extraits ont été normalisés selon le format "client-année.docx". Pour améliorer notre dataset,

```
        writer.writerow([file_name, data, incertitudes, travaux, avancements])
```

```
In [17]: dataset=pd.read_csv('Extraction.csv')
         dataset
```

Out[17]:

	Nom du fichier	Contenu complet	Partie 01 (obstacles et incertitudes)	Partie 02 (Travaux)	Partie 03 (Avancements Technologiques)
0	01.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
1	V4.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
2	PB.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
3	VF.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
4	PB.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
...	...	...	...	...	...
95	V2.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
96	V2.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
97	SM es.docx	A.\tQuels obstacles technologiques avez-vous d...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
98	(1).docx	\nA.\tQuels obstacles technologiques avez-vous...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...
99	3C.docx	\nA.\tQuels obstacles technologiques avez-vous...	A.\tQuels obstacles technologiques avez-vous d...	B.\tQuels travaux avez-vous effectués au cours...	C.\tQuels Avancements technologiques avez-vous...

100 rows × 5 columns

FIGURE 7.2 – Stockage du contenu extrait et création du dataset.

nous avons ajouté deux colonnes supplémentaires : "Client" et "Année". Cela nous permet de mieux organiser les rapports et facilitera les recherches ultérieures des utilisateurs qui souhaitent trouver des informations spécifiques.

- **Extraction des titres des rapports :** Nous avons également envisagé d'extraire les titres des rapports et de les inclure dans une colonne distincte appelée "Titre". Cette étape permet de regrouper les rapports en fonction de leur sujet principal, offrant ainsi une vue d'ensemble plus claire du contenu de chaque rapport.
- **Conservation de la langue d'origine :** Bien que tous les rapports de notre dataset soient rédigés en français après avoir effectué un processus de nettoyage, y compris la traduction de l'anglais vers le français, nous avons jugé utile d'ajouter une autre colonne intitulée "Langue de rédaction". Cette colonne indiquera la langue initiale dans laquelle le rapport a été rédigé, afin de conserver une trace de la rédaction d'origine.

En ajoutant ces informations supplémentaires dans notre dataset, nous améliorons sa qualité et sa convivialité, ce qui facilitera les futures recherches d'informations pertinentes pour les utilisateurs.

7.2.6 Classification des rapports

Dans le but d'améliorer la gestion et l'analyse de notre ensemble de rapports, nous avons pensé à ajouter une colonne nommée "Classe (Domaine)". Cette colonne serait dédiée à la classification des rapports en fonction du type de projet entrepris. L'objectif de cette stratégie était d'optimiser et d'automatiser le processus de classification, permettant ainsi d'améliorer l'efficacité de notre analyse et de faciliter la recherche d'informations spécifiques dans notre dataset.

Cette tâche s'est révélée être la plus délicate. Bien que notre ambition était de l'exécuter en utilisant le machine learning (algorithme de classification), cela s'est avéré impossible en l'ab-

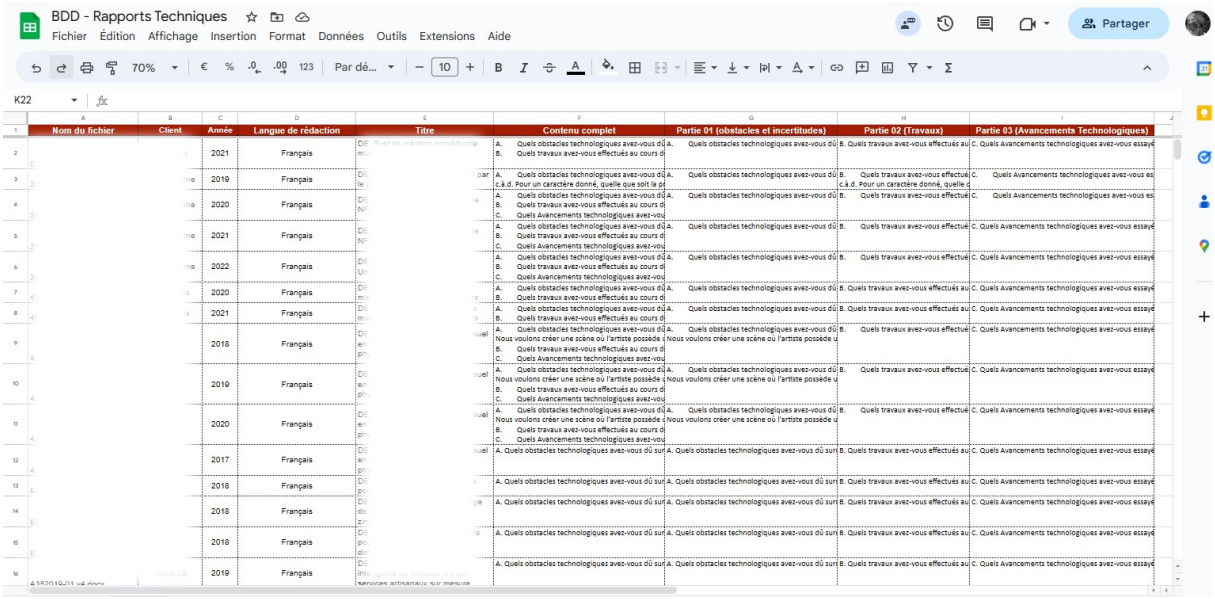


FIGURE 7.3 – Le dataset actualisé après l'extraction des informations pertinentes.

sence d'une liste exhaustive de classes préétablies permettant d'attribuer une catégorie spécifique à chaque rapport.

Par conséquent, nous avons dû faire usage d'un outil d'extraction ciblant une section clé de chaque "Partie 03" de chaque rapport. Cette section contient généralement des indications et des informations sur le domaine ou la classe du projet entrepris. Ces informations ont ensuite été utilisées pour déterminer et attribuer le domaine pertinent, tout en mettant à jour le dataset. Les étapes suivies sont :

- **Etape 01 : Identification de la partie 03 et affichage du paragraphe**
  Dans cette première phase, notre objectif est de localiser précisément la "Partie 03" dans chaque rapport. Une fois identifiée, nous extrayons le paragraphe associé pour un examen plus approfondi. L'opération d'impression du paragraphe nous permet d'obtenir une représentation visuelle de l'information que nous traitons, facilitant ainsi la compréhension et l'analyse ultérieure.
- **Etape 02 : Construction d'une liste de mots de la partie 03**
  Suite à l'extraction du paragraphe, la seconde étape consiste à construire une liste exhaustive de mots contenus dans cette partie du rapport. Cette étape nous permet de cataloguer et d'organiser les termes présents, offrant ainsi une meilleure vue d'ensemble du contenu et facilitant l'identification de mots-clés potentiels.

```
In [28]: # extraction du domaine from the last paragraph witch begin par  C.*Quels AVANCEMENTS technologiques a ....

#Etape 01 : identification de la partie 03
request = service.files().get_media(fileId=df['id'][1])
file_content = io.BytesIO()
downloader = MediaIoBaseDownload(file_content, request)
done = False
while done is False:
    status, done = downloader.next_chunk()

#converter le io.BytesIO() to document
document = Document(file_content)
data_domaine = ""
bol_partie03=False
for p in document.paragraphs:
    if p.text=="C.	Quels Avancements technologiques avez-vous essayé de réaliser ? (35 lignes maximum) - projet de développement
        print("j'ai trouverer la partie 03 ")
        bol_partie03=True
    if bol_partie03 :
        data_domaine+=p.text
print(data_domaine)

j'ai trouverer la partie 03
C.	Quels Avancements technologiques avez-vous essayé de réaliser ? (35 lignes maximum) - projet de développement expérimen
talle projet concerne le domaine de développement de logiciel informatique appliqué à la conception et développement de jeux vi
```

FIGURE 7.4 – Identification de la partie 03 et affichage du paragraphe.

```
In [29]: #Etape 02 : Construction d'une liste des mots de la partie 03
mot=""
list_mots=[]
for c in data_domaine:
    if c!= " " and c!='.':
        mot+=c
    # on sauvegarde les .
    if c==".":
        list_mots.append(".")
    if c==" " or c==".":
        list_mots.append(mot)
        mot=""

In [30]: list_mots
Out[30]: ['.',
          'C',

In [31]: for mot in list_mots:
    if mot==".":
        print ("point trouvé")

point trouvé
point trouvé
point trouvé
point trouvé
point trouvé
point trouvé
```

FIGURE 7.5 – Construction d'une liste de mots de la partie 03.

- **Etape 03 : Identification de la phrase renfermant le domaine du projet**
  Pour la troisième étape, nous employons un processus analytique pour tenter de déceler la

phrase qui fait référence au domaine du projet. En nous appuyant sur notre répertoire de termes et une compréhension approfondie du contexte, nous visons à localiser la phrase qui nous fournira l'information cruciale sur la classification du projet. Cela contribue à renforcer l'exactitude et l'efficacité de notre système de classification.

```
In [32]: #Etape 03 : On essaye de trouver la phrase qui contient le domaine
domaine=""
domaine_bol=False
for mot in list_mots:
    #on detecte le mot domaines or domaines
    if mot=="domaine" or mot=="concerne":
        domaine_bol=True
    #si on detecte le mot domaine ou bien domaines dans le text on comence la concatenation
    if domaine_bol:
        domaine+=mot + " "
        #si on trouve le . on arrete la concatenation
        if mot==".":
            print("point trouvé")
            break
## cette phrase doit etre ajouter dans la colone domaine dans le dataset ou bien utilisée pour avoir le domaine
print(domaine)

point trouvé
concerne le domaine de développement de logiciel informatique appliqué à la conception et développement de jeux .
```

FIGURE 7.6 – Identification de la phrase renfermant le domaine du projet.

Cette stratégie de classification, nous a permis de gagner en efficacité, de comprendre les différents types de projets représentés dans nos rapports, et finalement de soutenir notre capacité à prendre des décisions éclairées basées sur des données structurées et bien organisées.

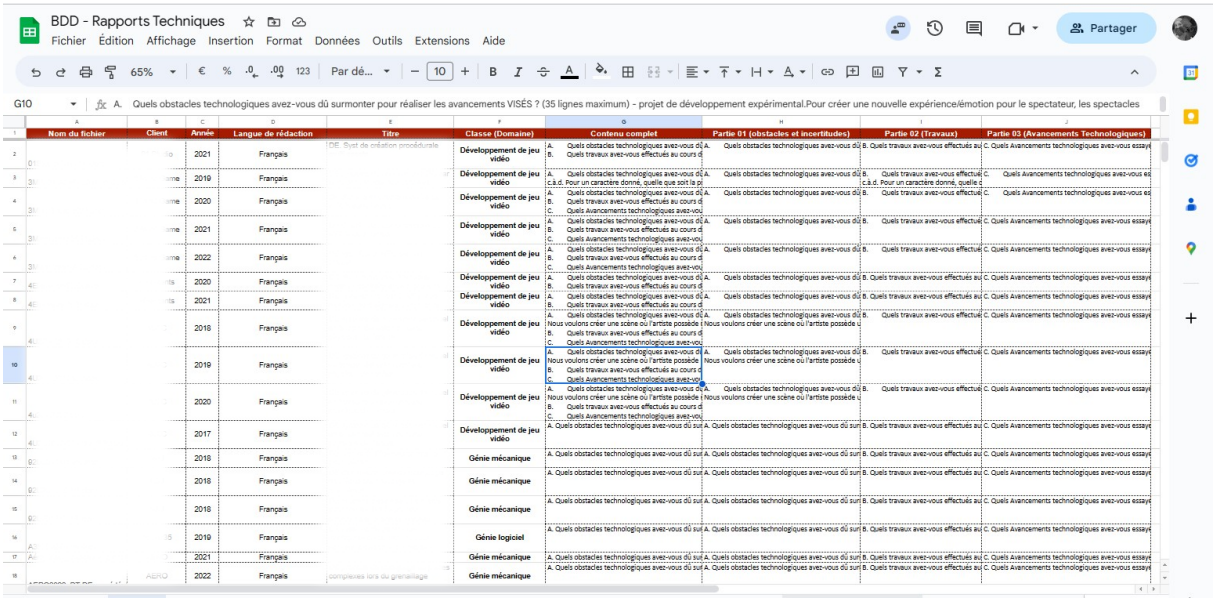
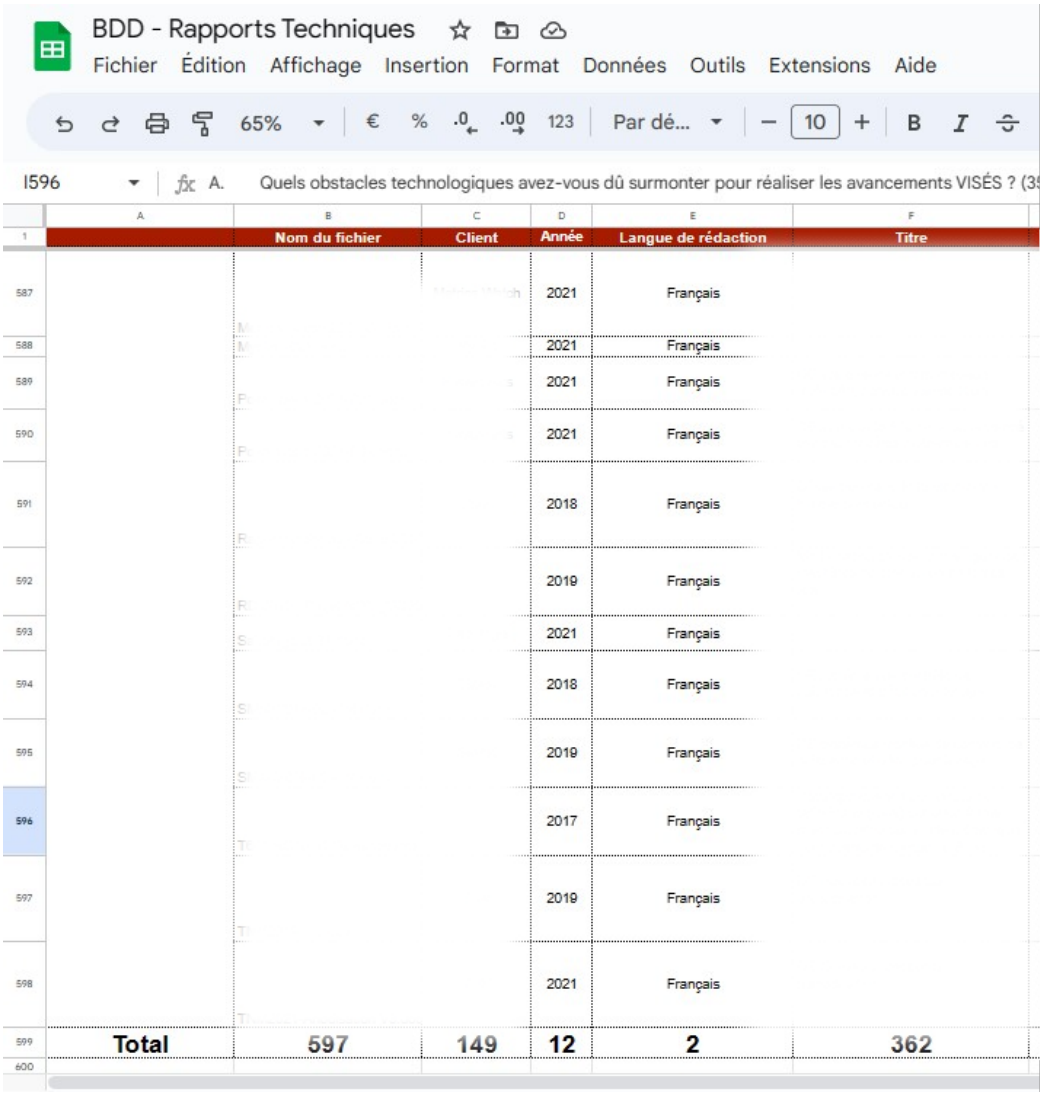


FIGURE 7.7 – Le dataset actualisé avec catégorisation (classification des rapports).

7.2.7 Analyse statistique des données

Nous avons créé un dataset contenant 597 rapports au total. Ces rapports proviennent de 362 projets différents, répartis dans 27 domaines distincts, et rédigés au cours de ces dernières années. De plus, ces projets ont été réalisés pour 149 clients différents. Voici un aperçu statistique sur quelques colonnes de ce dataset :



	Nom du fichier	Client	Année	Langue de rédaction	Titre
587			2021	Français	
588			2021	Français	
589			2021	Français	
590			2021	Français	
591			2018	Français	
592			2019	Français	
593			2021	Français	
594			2018	Français	
595			2019	Français	
596			2017	Français	
597			2019	Français	
598			2021	Français	
599	**Total**	**597**	**149**	**12**	**2**	**362**

FIGURE 7.8 – Statistiques sur les données.

Analysons le nombre de domaines couverts dans notre ensemble de rapports techniques :

D'après les données graphiques, il est clair que les projets de "Développement de jeux vidéo" sont les plus fréquemment traités au sein de MB inc. On constate que l'entreprise compte 59 clients actifs dans ce domaine spécifique, ce qui représente un pourcentage de 39,59% de l'ensemble de ses clients.

7.2.8 Compréhension des données

L'approche de l'automatisation de la tâche rédaction s'appuie sur la génération d'idées clés (descriptions techniques), telles que les incertitudes, les hypothèses et les avancées technologiques, spécifiques à chaque projet entrepris. Chaque secteur comporte une multitude de projets, chacun ayant ses propres aspects distinctifs en termes d'incertitudes, d'hypothèses et de progrès technologiques. Cette complexité rend difficile la reconnaissance exhaustive de toutes les idées pertinentes dans chaque domaine, sans parler de chaque projet individuel.

Pour surmonter cette difficulté, nous avons opté pour une stratégie de focalisation sur un type

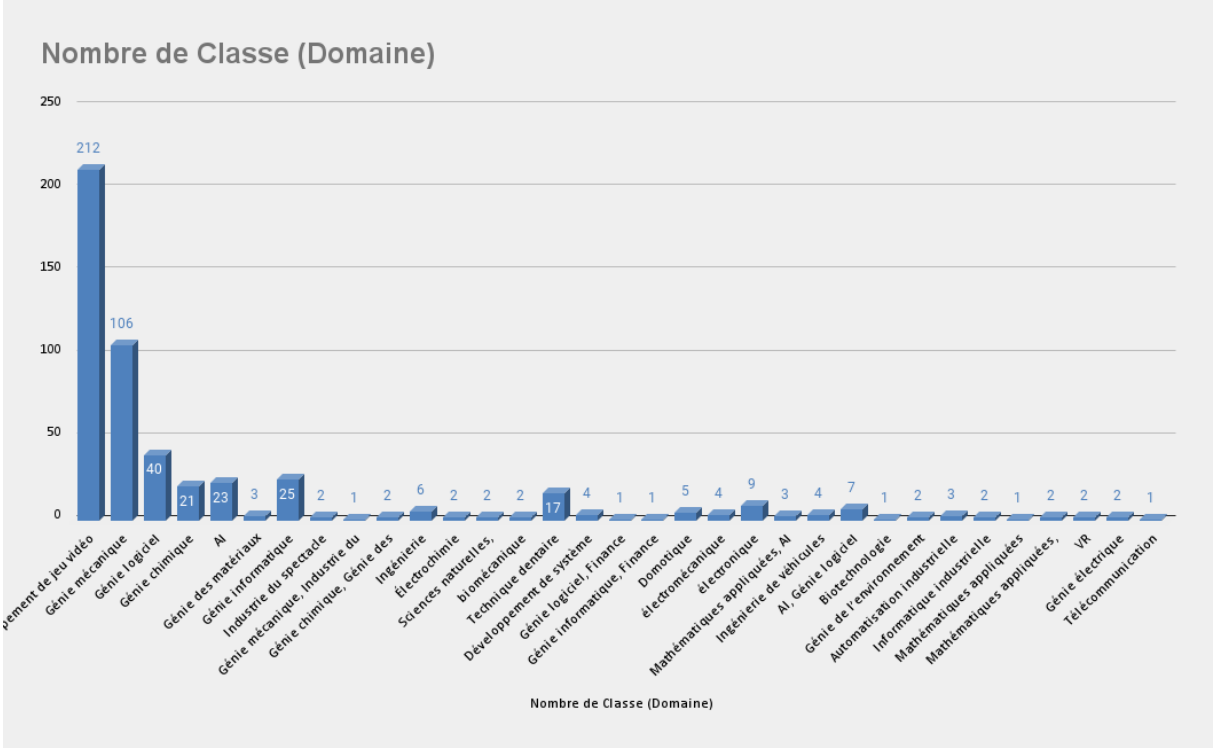


FIGURE 7.9 – Graphique illustrant le nombre de domaines présents dans l'ensemble des rapports techniques de notre dataset.

particulier de projet, à savoir le domaine du **"Développement de jeu vidéo"**. Cette catégorie de projet s'est démarquée par sa prédominance, avec une moyenne de 212 rapports compilés.

L'objectif de cette démarche est de concentrer notre attention sur un secteur unique, en identifiant le maximum d'idées abordées dans ses rapports. Cette démarche devrait nous permettre de constituer une **typologie** efficace concernant les incertitudes, les hypothèses et les avancées technologiques spécifiques à ce domaine.

En affinant ainsi notre objectif, nous cherchons à améliorer l'efficacité de la rédaction automatique de rapports dans le domaine du "Développement de jeux vidéo". En structurant et en standardisant les concepts et les idées clés issus de ce secteur, cette typologie servira d'outil précieux pour accélérer le processus de génération des descriptions techniques et augmenter leur précision. En outre, elle offrira une meilleure compréhension des tendances et des problématiques récurrentes dans le domaine, ce qui pourrait être bénéfique pour la prise de décision stratégique et la planification de projets futurs.

7.2.9 Préparation des données d'entraînement

Comme précisé précédemment, notre approche de génération se fait par étapes, en créant individuellement des descriptions techniques pour chacune des trois parties du rapport plutôt que de générer le rapport complet en une seule fois. Pour ce faire, nous envisageons de constituer trois ensembles de données distincts, chaque ensemble étant spécifique à une partie du rapport et utilisé pour former le modèle à générer la description technique de la partie concernée.

Dans l'immédiat, notre attention se portera sur la partie 01. Nous allons nous consacrer à la

préparation de l'ensemble de données essentiel pour générer uniquement des descriptions relatives à cette partie, étant donné son importance prédominante. Dans le cadre d'un projet, l'étape cruciale est le début, où l'on identifie les incertitudes et les défis technologiques. Les sections suivantes, à savoir les hypothèses (partie 02) et les avancements technologiques (partie 03), dépendent de cette première étape et seront générées en fonction des incertitudes et des obstacles identifiés (autrement dit, l'approche est généralisée pour les deux autres parties).

Comme déjà indiqué, notre objectif est de construire une typologie efficace pour les incertitudes du domaine du "Développement de jeux vidéo". L'objectif est multiple :

- **Identification et Gestion des Risques :** Une typologie d'incertitudes permet d'identifier les différents types de risques qui peuvent survenir au cours d'un projet. En comprenant ces incertitudes, les équipes de projet peuvent planifier à l'avance et mettre en place des stratégies pour les gérer efficacement.
- **Aide à la Prise de Décision :** En comprenant les différentes incertitudes qui peuvent survenir, les décideurs peuvent prendre des décisions plus éclairées. Une typologie d'incertitudes peut fournir un cadre pour évaluer les options et choisir les actions qui minimisent le risque.
- **Amélioration de la Communication et de la Collaboration :** Une typologie d'incertitudes peut aider à améliorer la communication au sein des équipes de projet en fournissant un langage commun pour discuter des risques. Cela peut également faciliter la collaboration, car les membres de l'équipe ont une meilleure compréhension des défis potentiels à surmonter.
- **Formation et Développement :** Une typologie d'incertitudes peut être utilisée comme outil de formation pour aider les nouveaux membres de l'équipe à comprendre les défis auxquels ils peuvent être confrontés. Cela peut contribuer à leur préparation et à leur capacité à répondre efficacement aux situations imprévues.
- **Évaluation et Amélioration des Processus :** En identifiant les incertitudes communes, une organisation peut évaluer et améliorer ses processus pour mieux gérer ces situations à l'avenir.

Pour atteindre notre objectif, nous envisageons de mettre en place un **système de tags (système d'étiquetage)** structuré comme suit :

- Chaque rapport est défini par cinq (5) tags : **Type du projet**, **Zone d'innovation**, **Type du jeu**, **Engin** et **Technologies utilisées**.
- À chaque rapport, on attribuera un ou plusieurs éléments correspondants de ces tags.
- L'engin spécifié pour un projet est unique. De même, un projet et un jeu ne peuvent avoir qu'un seul type défini. Cependant, un projet peut être associé à plusieurs zones d'innovation. Par conséquent, les colonnes "Type de projet", "Type du jeu" et "Engin" sont destinées à une **sélection unique**, tandis que les colonnes "Zone d'innovation" et "Technologies utilisées" permettent une **sélection multiple**.

En adoptant un système de tags, nous avons non seulement facilité et organisé la prise de notes pour le consultant lors des rencontres techniques avec les clients, mais avons aussi créé un système de classification qui permet une recherche d'informations aisée. Les tags agissent comme des indices précieux, indiquant le type de contenu dans chaque rapport, permettant une navigation efficace à travers notre dataset de projets.

En outre, ce système de tags s'est avéré être un excellent outil pour analyser des tendances. En surveillant la fréquence et la combinaison de tags spécifiques, nous avons été en mesure de déceler

des motifs ou des tendances dans le type de projets que nous avons réalisés, ou dans les zones d'innovation qui sont actuellement populaires. Plus encore, en explorant les diverses combinaisons de tags, nous avons stimulé notre créativité, générant de nouvelles idées ou perspectives.

Cependant, en dépit de ces avantages, l'utilisation de ce système de tags comme entrées pour notre modèle n'a pas été aussi efficace que prévu. Les tags, bien qu'utiles pour la navigation et l'analyse des tendances, peuvent ne pas fournir suffisamment de données pour un apprentissage efficace du modèle.

Afin de remédier à ce problème, nous avons trouvé une manière de transformer ces tags en questions pertinentes qui servent d'entrées pour le modèle. Ces questions sont centrées sur les défis, les obstacles et les incertitudes qui peuvent survenir lors d'un projet, transformant ainsi notre modèle en un outil capable d'anticiper et de résoudre des problèmes potentiels.

Pour structurer notre dataset, nous avons décidé de faire en sorte que les incertitudes et les obstacles mentionnés dans chaque rapport constituent nos sorties d'entraînement. À cette fin, nous avons ajouté deux nouvelles colonnes à notre base de données : "Prompt (Input)" et "Target "Incertitudes" (Output)". Ces colonnes contiennent respectivement les questions formulées à partir des tags et les incertitudes ou obstacles identifiés.

Afin de rendre le format de nos données plus intelligible pour le modèle, nous avons procédé à une division de chaque incertitude en une ligne distincte. Cela a impliqué la duplication de chaque rapport selon le nombre d'incertitudes, ce qui a nettoyé nos données et permis une augmentation de ces dernières. C'est une étape cruciale pour améliorer la robustesse de notre modèle en lui fournissant un ensemble de données d'apprentissage plus diversifié.

Ensuite, nous avons formulé des questions pour chaque incertitude en nous basant sur le système de tags que nous avons créé. Ce processus a abouti à la création de notre ensemble de données d'entraînement final, qui comprend 212 rapports et s'étend sur **607 lignes** au total.

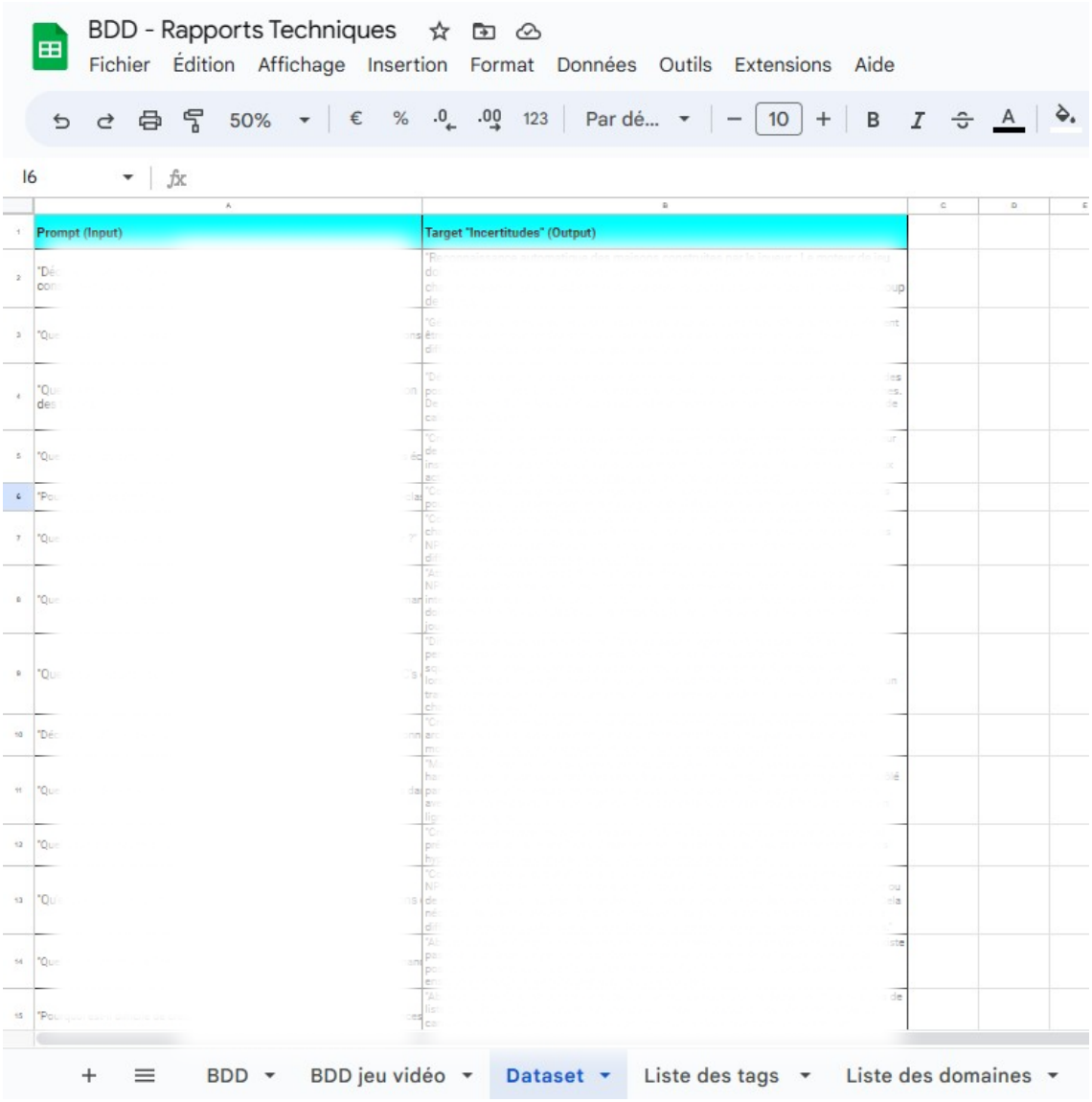


FIGURE 7.10 – Notre dataset d'entraînement.

Cet ensemble de données, soigneusement préparé et enrichi grâce à nos efforts de **nettoyage** et d'**augmentation des données**, est une ressource précieuse pour entraîner notre modèle à répondre de manière efficace aux incertitudes et aux obstacles évoqués dans nos projets.

7.3 Modélisation

7.3.1 Choix du modèle

De nombreux modèles sont disponibles et peuvent être ajustés pour notre tâche de création de texte liée à la description technique des projets. Le choix du modèle est influencé par divers facteurs. Dans notre scénario, nous avons décidé d'entraîner deux modèles spécifiques, **GPT-Neo** et **BART**, et de les évaluer l'un par rapport à l'autre. Cette décision a été prise pour les raisons suivantes :

- **Capacités de génération de texte :** GPT-Neo et BART sont deux des modèles de langage les plus performants disponibles à l'heure actuelle. Ils sont conçus pour comprendre le contexte d'un texte et générer du contenu en conséquence. C'est précisément ce dont on a besoin pour notre tâche.
- **Formulation de question-réponse :** GPT-Neo et BART sont tous deux capables de formuler des réponses cohérentes et contextuellement pertinentes à des questions. Dans notre base de données, nous avons une colonne qui agit comme une question (input) et une autre qui est un petit paragraphe (output). Les deux modèles peuvent gérer ce type de données.
- **Compréhension du langage naturel :** Ces deux modèles ont été entraînés sur de grandes quantités de texte, leur permettant de comprendre et de générer du texte dans divers styles et contextes. Cela peut aider à générer des descriptions techniques précises et appropriées pour nos projets.
- **Extensibilité :** GPT-Neo et BART sont tous deux très extensibles, ce qui signifie qu'on peut les fine-tuner pour de nombreuses tâches différentes. Si nous aurons besoin de d'ajouter de nouvelles fonctionnalités à l'avenir, ces modèles pourraient probablement gérer cela.
- **BART pour la restructuration de l'information :** BART est un modèle de langage bidirectionnel. Cela signifie qu'il lit l'ensemble du contexte d'entrée avant de générer une sortie, ce qui peut être très utile pour réorganiser l'information, ce qui est souvent nécessaire dans les descriptions techniques.
- **GPT-Neo pour la cohérence du texte :** GPT-Neo, un descendant de la famille GPT, est un modèle de langage autonome qui est spécialement bon pour générer des textes longs et cohérents. Cela pourrait être bénéfique pour produire des descriptions détaillées et fluides.

Il est important de noter que, bien que ces modèles soient utiles pour la prédiction du mot suivant et la génération de texte, le choix du meilleur modèle dépend de nombreux facteurs, notamment la longueur du contexte, la quantité de données disponibles pour l'entraînement et les exigences spécifiques de notre tâche. Une bonne approche serait de tester ces modèles sur l'ensemble de nos données et de voir lequel donne les meilleurs résultats.

7.3.2 Implémentation

Le processus d'implémentation pour les deux modèles GPT-Neo et BART suit les mêmes étapes :

- **Installation des dépendances :**
  Les packages Python nécessaires pour ce projet sont installés via pip. Ces packages sont : torch (la bibliothèque de calcul tensoriel qui alimente les modèles de PyTorch), transformers (la bibliothèque de Hugging Face qui fournit les modèles de langage pré-entraînés et les outils pour les travailler), et pandas (la bibliothèque de manipulation de données qui est utilisée pour lire et préparer les données).
- **Importation des packages nécessaires :**
  Ce code importe les classes et fonctions nécessaires de ces packages. Cela comprend les classes de modèle et de tokenizer spécifiques au modèle entraîné, la classe Trainer et TrainingArguments qui aident à l'entraînement et à la configuration du modèle, la classe Dataset qui permet de définir une interface personnalisée pour travailler avec les données, et la fonction train_test_split qui permet de diviser facilement les données en ensembles d'entraînement et de validation.
- **Définition du Dataset :**
  Cette étape consiste à créer une classe personnalisée qui hérite de la classe Dataset de PyTorch. Cette classe personnalisée est utilisée pour traiter les données de texte, les convertir en tokens avec le tokenizer, puis en tenseurs qui peuvent être utilisés pour l'entraînement du modèle.
- **Préparation du modèle :**
  Le modèle pré-entraîné est téléchargé de Hugging Face. Certains paramètres du modèle sont gelés pour empêcher leur mise à jour lors de l'entraînement, cela est généralement fait pour le transfert d'apprentissage où seuls les paramètres des dernières couches du modèle sont mis à jour. L'appareil pour l'entraînement est également configuré ici, en utilisant un GPU si disponible, sinon un CPU.
- **Chargement des données :**
  Les données pour l'entraînement du modèle sont chargées à partir d'un fichier CSV à l'aide de pandas. Les colonnes de texte pertinentes sont extraites et converties en listes.
- **Nettoyage des données :**
  Toutes les duplications de données sont supprimées, et les caractères indésirables (dans ce cas, les guillemets) sont également supprimés.
- **Préparation des jeux de données d'entraînement et de validation :**
  Les données sont divisées en un ensemble d'entraînement et un ensemble de validation à l'aide de la fonction train_test_split. Cela permet de disposer d'un ensemble de données distinct pour évaluer les performances du modèle qui n'a pas été utilisé pendant l'entraînement.
- **Préparation du tokenizer :**
  Un token de padding est ajouté au tokenizer, qui est utilisé pour remplir toutes les séquences de texte à la même longueur pour le traitement en lots. Ensuite, les embeddings du modèle sont redimensionnés pour inclure cet ajout.
- **Création des jeux de données :**
  Les ensembles de données d'entraînement et de validation sont créés en utilisant la classe de jeu de données personnalisée définie précédemment. Les données de texte sont passées à travers le tokenizer et préparées pour l'entraînement du modèle.

- **Définition des arguments d'entraînement :**
  Les paramètres pour l'entraînement du modèle sont définis à l'aide de la classe TrainingArguments. Cela comprend le nombre d'époques d'entraînement, la taille du lot, le nombre de steps de warmup, la dégradation du poids, etc.
- **Initialisation du Trainer :**
  Le Trainer est préparé avec le modèle, les arguments d'entraînement et les jeux de données. Le Trainer s'occupe de toute la boucle d'entraînement, y compris la mise à jour des paramètres du modèle et le suivi des métriques.
- **Entraînement du modèle :**
  Le modèle est formé en utilisant la méthode train du Trainer. Cette méthode parcourt toutes les époques, met à jour les paramètres du modèle à chaque étape et suit les métriques.

Il convient de souligner que, malgré l'uniformité des étapes globales, des variations spécifiques peuvent survenir en fonction des caractéristiques propres à chaque modèle. Par exemple, le choix du modèle pré-entraîné à utiliser, la procédure de tokenisation, ainsi que certains paramètres de configuration du modèle, peuvent varier entre GPT-Neo et BART. Toutefois, la structure générale du processus d'implémentation demeure constante. Vous trouverez le code source détaillé pour l'entraînement de chaque modèle dans la section "Annexes" du document.

Il convient de noter aussi que l'entraînement de modèles de langage avancés comme GPT-Neo et BART nécessite généralement des ressources informatiques substantielles, en particulier, l'utilisation d'une unité de traitement graphique (GPU) est indispensable.

Un GPU est capable de traiter un grand nombre d'opérations simultanément, ce qui est idéal pour les calculs matriciels qui sont couramment utilisés dans le domaine de l'apprentissage profond. Par rapport aux unités de traitement central (CPU), les GPU sont généralement capables de traiter les calculs liés à l'entraînement des modèles d'apprentissage profond beaucoup plus rapidement. C'est pourquoi ils sont souvent préférés pour ce genre de tâches.

De plus, compte tenu de la taille conséquente de ces modèles, il est nécessaire d'avoir un GPU doté d'une capacité mémoire significative pour pouvoir stocker les modèles et les données pendant l'entraînement.

Résultats de l'entraînement :

Epoch	Training Loss	Validation Loss
1	No log	1.580473
2	1.919100	1.331327
3	1.919100	1.259176
4	1.196700	1.239716
5	0.845400	1.277813
6	0.845400	1.368354
7	0.483300	1.423842
8	0.208700	1.487547
9	0.208700	1.513566
10	0.132400	1.542761

TABLE 7.1 – Résultats de l'entraînement : GPT-Neo

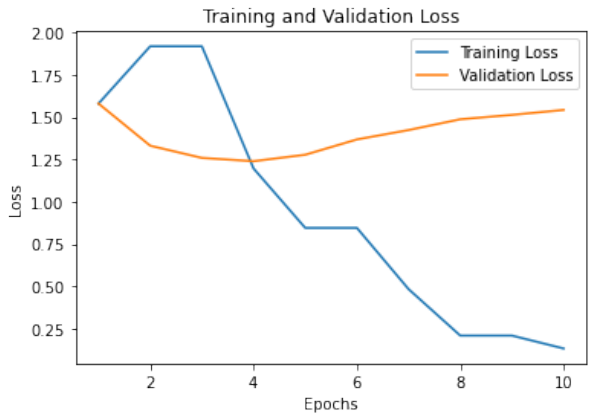


FIGURE 7.11 – Graphe des pertes : GPT-Neo

Epoch	Training Loss	Validation Loss
1	11.482100	10.091373
2	8.296100	7.356726
3	6.574100	5.017141
4	4.877200	3.932638
5	4.141600	3.329330
6	3.375200	2.597049
7	2.492400	1.720202
8	1.603800	1.090688
9	1.028500	0.925022
10	0.963300	0.885293

TABLE 7.2 – Résultats de l'entraînement : BART

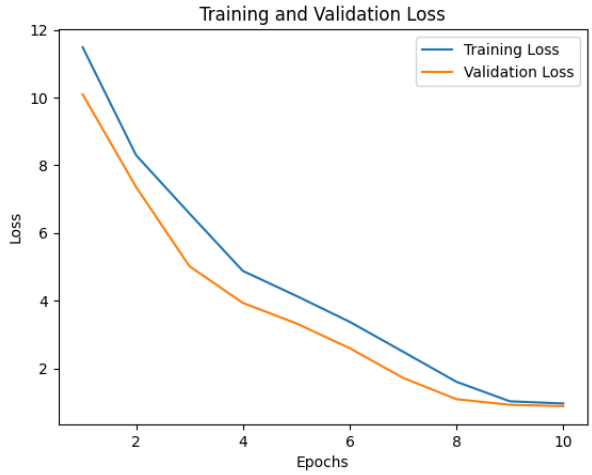


FIGURE 7.12 – Graphe des pertes : BART

Interprétation des résultats :

Ces résultats indiquent que les deux modèles ont réussi à apprendre à partir de nos données d'entraînement, mais il y a certaines différences notables à souligner.

- **Modèle 01 : GPT-Neo**
  Pour GPT-Neo, la perte de validation diminue jusqu'à la 4ème époque puis commence à augmenter. C'est un signe classique de surajustement : le modèle s'améliore sur les données d'entraînement, mais sa performance sur les données de validation (données qu'il n'a jamais vues) commence à se détériorer. Cela indique que le modèle commence à mémoriser les données d'entraînement au lieu de généraliser à partir de celles-ci. Une des stratégies à mettre en place pour lutter contre le surapprentissage est : la régularisation (déjà faite), ou l'arrêt de l'entraînement plus tôt **(early stopping)**, c'est-à-dire l'arrêt précoce de l'entraînement lorsque la perte de validation commence à augmenter. Les résultats sont les suivants :

Epoch	Training Loss	Validation Loss
1	No log	1.610854
2	1.819200	1.355733
3	1.819200	1.301840
4	1.199600	1.292707

TABLE 7.3 – Résultats de l'entraînement après application d'**early stopping** : GPT-Neo

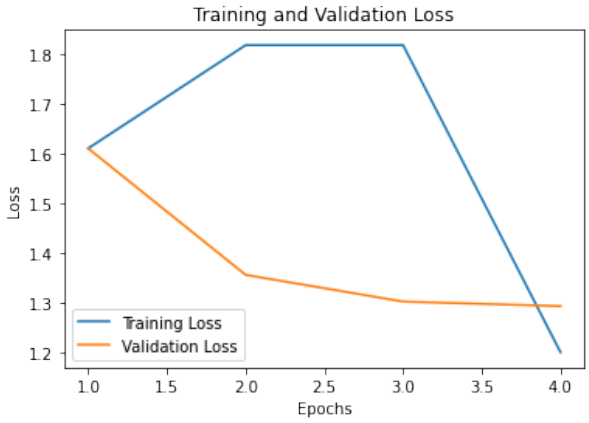


FIGURE 7.13 – Graphe des pertes après application d'**early stopping** : GPT-Neo

Le modèle a été entraîné pendant 4 époques avant l'arrêt anticipé. L'erreur d'entraînement (Training Loss) diminue constamment, passant de "No log" à 1.199600, ce qui indique que le modèle continue d'apprendre et de s'améliorer sur les données d'entraînement. L'erreur de validation (Validation Loss) diminue également de manière constante, passant de 1.610854 à 1.292707. C'est un bon signe, cela signifie que le modèle s'améliore et généralise bien sur les données non vues pendant l'entraînement. Comparé à notre précédente tentative d'entraînement de GPT-Neo sans arrêt anticipé, où l'erreur de validation a commencé à augmenter après la 4ème époque (indiquant un surapprentissage), l'arrêt anticipé ait aidé à prévenir le surapprentissage dans ce cas.

- **Modèle 02 : BART**
  Comme pour GPT-Neo, l'erreur d'entraînement de BART diminue constamment, passant de 11.482100 à 0.963300. C'est un bon signe d'apprentissage. À la différence de GPT-Neo, l'erreur de validation de BART diminue constamment à chaque époque, passant de 10.091373 à 0.885293. C'est un très bon signe, car cela indique que le modèle continue d'améliorer ses performances sur les données de validation, et qu'il généralise bien à de nouvelles données.

Cependant, il est important de noter que la perte de validation n'est qu'une mesure de performance. Il serait utile de tester les deux modèles sur un ensemble de test distinct pour voir comment ils se comportent dans la pratique.

7.3.3 Tests

Nous allons scrupuleusement examiner la performance de nos deux modèles en les confrontant à un ensemble de données de test distinct.

a. Modèle 01 : GPT-Neo

In [27]:
```
# Prenez un prompt de test au hasard
prompt = "                                                                                ?"

# Encodez le prompt pour qu'il puisse être utilisé par le modèle
input_ids = tokenizer.encode(prompt, return_tensors='pt').to(device)

# Générez la sortie
output = model.generate(input_ids, max_length=250, num_return_sequences=1, temperature=0.7)

# Décodez la sortie pour obtenir le texte généré
decoded_output = [tokenizer.decode(output[0], skip_special_tokens=True)]
print(decoded_output)
```

```
The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your i
nput's `attention_mask` to obtain reliable results.
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
```

FIGURE 7.14 – Résultat du Test 01 du modèle GPT-Neo.

In [31]:
```
# Prenez un prompt de test au hasard
prompt = "Qu                                                                          ?"

# Encodez le prompt pour qu'il puisse être utilisé par le modèle
input_ids = tokenizer.encode(prompt, return_tensors='pt').to(device)

# Générez la sortie
output = model.generate(input_ids, max_length=240, num_return_sequences=1, temperature=0.7)

# Décodez la sortie pour obtenir le texte généré
decoded_output = [tokenizer.decode(output[0], skip_special_tokens=True)]
print(decoded_output)
```

```
The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your i
nput's `attention_mask` to obtain reliable results.
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
```

FIGURE 7.15 – Résultat du Test 02 du modèle GPT-Neo.

In [30]:
```
# Prenez un prompt de test au hasard
prompt = "Qu                                                                          ?"

# Encodez le prompt pour qu'il puisse être utilisé par le modèle
input_ids = tokenizer.encode(prompt, return_tensors='pt').to(device)

# Générez la sortie
output = model.generate(input_ids, max_length=240, num_return_sequences=1, temperature=0.7)

# Décodez la sortie pour obtenir le texte généré
decoded_output = [tokenizer.decode(output[0], skip_special_tokens=True)]
print(decoded_output)
```

```
The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your i
nput's `attention_mask` to obtain reliable results.
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
```

FIGURE 7.16 – Résultat du Test 03 du modèle GPT-Neo.

b. Modèle 02 : BART

```
In [36]: # Supposons que 'question' est la question à laquelle vous voulez répondre
         question1 = "[illegible] ?"

         # Encodez la question et passez-la au modèle
         inputs1 = tokenizer(question1, return_tensors='pt').to(device)
         outputs1 = model.generate(inputs1['input_ids'], max_length=300, num_beams=5)

         # Décodez la sortie du modèle pour obtenir la réponse
         answer1 = tokenizer.decode(outputs1[0], skip_special_tokens=True)

         print(answer1)
```

FIGURE 7.17 – Résultat du Test 01 du modèle BART.

```
In [37]: # Supposons que 'question' est la question à laquelle vous voulez répondre
         question2 = "[illegible] ?"

         # Encodez la question et passez-la au modèle
         inputs2 = tokenizer(question2, return_tensors='pt').to(device)
         outputs2 = model.generate(inputs2['input_ids'], max_length=300, num_beams=5)

         # Décodez la sortie du modèle pour obtenir la réponse
         answer2 = tokenizer.decode(outputs2[0], skip_special_tokens=True)

         print(answer2)
```

FIGURE 7.18 – Résultat du Test 02 du modèle BART.

```
In [38]: # Supposons que 'question' est la question à laquelle vous voulez répondre
         question3 = "[illegible] ?"

         # Encodez la question et passez-la au modèle
         inputs3 = tokenizer(question3, return_tensors='pt').to(device)
         outputs3 = model.generate(inputs3['input_ids'], max_length=300, num_beams=5)

         # Décodez la sortie du modèle pour obtenir la réponse
         answer3 = tokenizer.decode(outputs3[0], skip_special_tokens=True)

         print(answer3)
```

FIGURE 7.19 – Résultat du Test 03 du modèle BART.

Interprétation :

Dans l'ensemble, les deux modèles génèrent des textes pertinents et cohérents en réponse aux prompts. Cependant, il y a des différences dans la qualité et la pertinence des résultats.

- Les réponses générées par GPT-Neo sont assez détaillées et spécifiques. Elles couvrent bien le sujet des prompts et donnent une explication approfondie des problèmes et des objectifs liés à la réalité virtuelle et à la création de personnages. Cependant, certaines parties des textes semblent répétitives, en particulier dans le test 1 et le test 3.

- Les réponses générées par BART sont plus courtes et moins spécifiques. Elles soulignent les défis et les obstacles technologiques à surmonter, mais ne donnent pas beaucoup de détails sur comment aborder ces problèmes. Il y a également des parties du texte qui sont un peu ambiguës et pas clairement compréhensibles, par exemple "réaliser les NPC" dans le test 2 ou "répondre à la précision de manière précisante" dans le test 3.

En conclusion, d'après les résultats de ces tests, GPT-Neo donne des réponses plus détaillées et plus spécifiques, mais peut être répétitif. BART, en revanche, donne des réponses plus courtes et plus générales, mais peut parfois être ambigu et manquer de clarté.

7.3.4 Évaluation

Évaluation générale :

Les résultats de l'évaluation fournissent des informations supplémentaires sur la performance des modèles et leur efficacité.

Modèle	GPT-Neo	BART
eval_loss	1.292707085609436	0.8852930665016174
eval_runtime	71.9209	79.9859
eval_samples_per_second	0.848	1.525
eval_steps_per_second	0.056	0.025
epoch	4.0	10.0

TABLE 7.4 – Résultats de l'évaluation en utilisant la fonction "*trainer.evaluate()*".

L'interprétation de ces résultats est la suivante :

- **eval_loss :** Il s'agit de l'erreur de validation calculée après l'entraînement du modèle. Plus le chiffre est bas, mieux c'est, car cela signifie que le modèle fait moins d'erreurs sur l'ensemble des données de validation. BART a un eval_loss de 0.885, ce qui est inférieur à celui de GPT-Neo, qui est de 1.292. Cela suggère que BART performe mieux sur l'ensemble de validation que GPT-Neo.
- **eval_runtime :** Il s'agit du temps nécessaire pour évaluer le modèle. Ici, GPT-Neo a un temps d'exécution de 71.9209 secondes, ce qui est plus rapide que BART qui a un temps d'exécution de 79.9859 secondes. Cela signifie que GPT-Neo est plus rapide à évaluer que BART.
- **eval_samples_per_second :** Il s'agit du nombre d'échantillons traités par seconde pendant l'évaluation. Plus ce chiffre est élevé, plus le modèle est rapide.
  BART a un eval_samples_per_second de 1.525, ce qui est plus élevé que GPT-Neo, qui est de 0.848. Cela signifie que BART est capable de traiter les échantillons plus rapidement que GPT-Neo pendant l'évaluation.
- **eval_steps_per_second :** Il s'agit du nombre de pas d'évaluation exécutés par seconde. Plus ce chiffre est élevé, plus le modèle est rapide. GPT-Neo a un eval_steps_per_second de 0.056, ce qui est plus élevé que BART, qui est de 0.025. Cela signifie que GPT-Neo est capable d'exécuter les pas d'évaluation plus rapidement que BART.
- **epoch :** Il s'agit du nombre d'époques d'entraînement terminées pour chaque modèle. GPT-Neo a été entraîné pendant 4 époques, tandis que BART a été entraîné pendant 10 époques.

Calcul de perplexité :

La perplexité est une mesure couramment utilisée en langage naturel pour évaluer la performance des modèles de langue. Une perplexité plus faible indique que le modèle de langue est plus confiant dans ses prédictions. En d'autres termes, si un modèle assigne une probabilité plus élevée aux mots qu'il prédit correctement, sa perplexité sera plus faible.

Modèle	**Perplexité**
GPT-Neo	3.64
BART	2.42

TABLE 7.5 – Perplexités calculées pour les 2 modèles entraînés.

Dans notre cas, le modèle BART a une perplexité de 2.42, ce qui est inférieur à celle du modèle GPT-Neo qui a une perplexité de 3.64. Cela signifie que, sur la base de la mesure de la perplexité, BART est le modèle qui a performé de manière supérieure sur nos données. Il est plus sûr de ses prédictions par rapport au modèle GPT-Neo.

Cependant, il est important de noter que, bien que la perplexité soit une mesure utile, elle ne devrait pas être la seule mesure utilisée pour évaluer la performance d'un modèle de langue. D'autres facteurs tels que la compréhension de la langue, la capacité à générer du texte cohérent et pertinent, et la capacité à accomplir des tâches spécifiques devraient également être pris en compte.

Analyse de la similarité des textes :

Consiste à faire une évaluation de la qualité de la génération de textes par rapport à un standard idéal ou une "réponse correcte". Cela peut aider à comprendre comment les sorties du modèle se rapprochent du texte de référence souhaité. Le tableau suivant résume les résultat de cette analyse entre deux textes générés par nos 2 modèles et les textes de référence correspondants.

	Matrices de similarité
GPT-Neo	$\begin{bmatrix} 1.0000002 & 0.960011 \\ 0.960011 & 1.0 \end{bmatrix}$
BART	$\begin{bmatrix} 1.0000002 & 0.9557236 \\ 0.9557236 & 1.0 \end{bmatrix}$

TABLE 7.6 – Résultats de l'analyse de similarité

L'analyse basée sur la similarité de cosinus entre les textes produits par les modèles GPT-Neo et BART pour un test donné et le texte de référence révèle des scores de 0.96 et 0.955 respectivement. Ces scores sont assez élevés, indiquant une grande similarité entre les textes générés par les modèles et le texte de référence. Cela suggère une performance remarquable des modèles dans la génération de textes qui se rapprochent grandement de ce que l'on pourrait attendre dans une situation réelle.

Évaluation de la diversité des prédictions :

Les scores de diversité indiquent à quel point les prédictions d'un modèle sont diversifiées. Cela peut être un indicateur de la créativité du modèle, ou de sa capacité à générer des résultats variés et non redondants.

Modèle	Diversity score
GPT-Neo	0.707459009429955
BART	0.6775597269624574

TABLE 7.7 – Scores de diversité calculés pour les 2 modèles entraînés.

Dans notre tableau, GPT-Neo a un score de diversité de 0.707 et BART a un score de 0.678. Ces valeurs indiquent que les deux modèles ont une capacité relativement similaire à générer des prédictions diversifiées, avec un léger avantage pour le modèle GPT-Neo.

En termes plus simples, dans le contexte de notre étude, GPT-Neo a tendance à générer des réponses plus variées par rapport à BART.

7.4 Déploiement de la solution

Le déploiement constitue une étape fondamentale dans le contexte d'un projet qui fait appel au Transfer Learning. Cette phase implique l'intégration du modèle préalablement entraîné dans un environnement de production, permettant ainsi de s'attaquer à des problématiques réelles et de générer des solutions pratiques et efficaces.

Dans le contexte particulier de notre projet, nous avons orchestré le déploiement de nos modèles entrainés. Notre ambition est de donner un aperçu exhaustif et minutieux de notre processus de déploiement, en mettant en exergue les aspects techniques, tout en accentuant l'importance cruciale de l'expérience utilisateur.

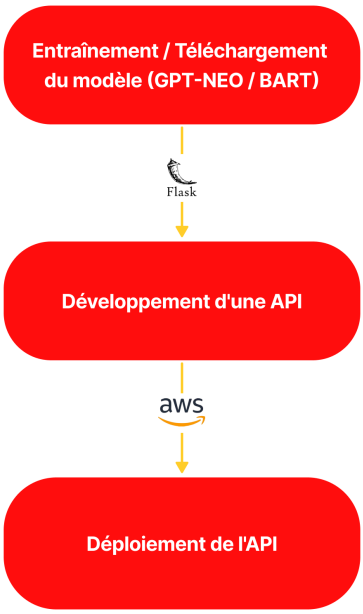


FIGURE 7.20 – Le processus du déploiement.

7.4.1 Architecture du déploiement

Dans cette partie, nous avons mis en place une architecture de déploiement sophistiquée pour faciliter l'interaction entre l'utilisateur et nos modèle de génération de textes, GPT-Neo et BART. Cette architecture est le fruit d'une réflexion approfondie et d'un travail méticuleux visant à créer un système robuste, performant et facile à utiliser. Elle est conçue pour gérer efficacement les requêtes des utilisateurs, les transmettre à nos modèles et renvoyer les réponses générées de manière fluide et efficace.

Cette architecture de déploiement s'articule autour de trois composants principaux : l'**API**, le **modèle à implémenter**, et l'**interface utilisateur**. Chacun de ces composants joue un rôle crucial dans le fonctionnement global du système et a été conçu avec une attention particulière portée à la performance, à la fiabilité et à l'expérience utilisateur.

- **L'API, ou interface de programmation d'application :**
  Le pont qui relie l'utilisateur à notre modèle. Elle est responsable de la réception des requêtes de l'utilisateur, de leur traitement et de la transmission des résultats générés par le modèle. Pour assurer une interaction fluide et efficace, nous avons construit notre API en utilisant un framework web robuste et performant.
- **Le modèle à implémenter :**
  qui est au cœur de notre système, est responsable de la génération de textes. Il prend en entrée les promptes transmises par l'API, les traite et génère des réponses sous forme de texte.

- **L'interface utilisateur :**
  Le point de contact entre l'utilisateur et notre système. Elle a été conçue pour être intuitive et facile à utiliser, permettant aux utilisateurs d'envoyer des promptes et de recevoir des réponses de manière simple et directe. L'interface est également responsable de la présentation des résultats générés par le modèle de manière claire et lisible.

Dans le processus de fonctionnement, l'utilisateur envoie une prompte à travers l'interface utilisateur. Cette prompte est ensuite transmise à l'API, qui la redirige vers le modèle. Ce dernier génère du texte basé sur la prompte, et cette réponse est renvoyée à l'utilisateur via l'API et l'interface utilisateur.

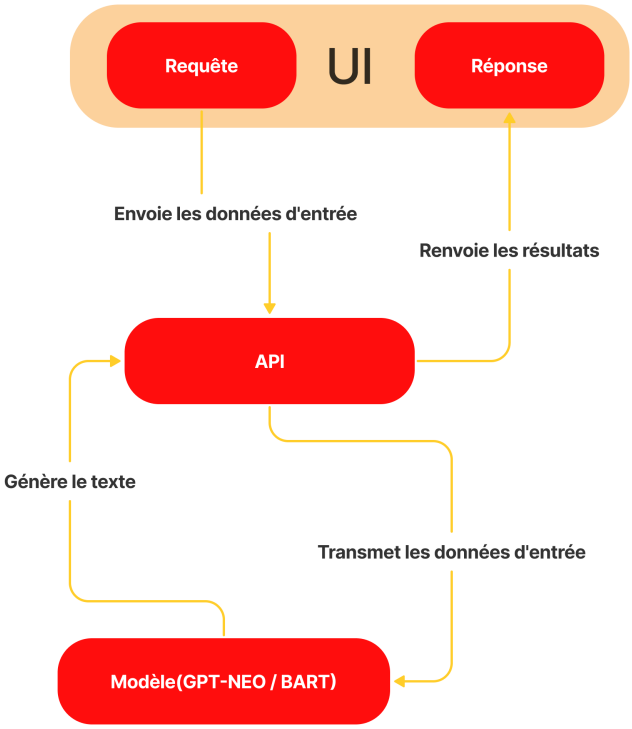


FIGURE 7.21 – L'architecture du déploiement.

7.4.2 Développement de l'API

Le développement de l'API est une étape cruciale dans le déploiement de nos modèles. L'API, ou Interface de Programmation d'Applications, est un ensemble de règles et de protocoles qui permet à différentes applications logicielles de communiquer entre elles. Dans notre cas, l'API sert de pont entre l'utilisateur et nos modèles (GPT-Neo / BART).

Nous avons choisi d'utiliser **Flask**, un micro-framework web en Python, pour développer notre API. Flask a été choisi pour sa simplicité, sa flexibilité et sa facilité d'intégration avec d'autres

bibliothèques Python. De plus, Flask est léger et ne nécessite pas de dépendances externes, ce qui le rend idéal pour notre projet.

Notre API est conçue avec une seule route, */predict*, qui accepte uniquement les requêtes POST. Les requêtes POST sont utilisées pour envoyer des données à être traitées à un serveur. Dans notre cas, les données envoyées sont les promptes de texte saisis par l'utilisateur et que le modèle choisi va les générer. Lorsqu'une requête POST est reçue sur la route */predict*, plusieurs étapes se produisent :

- **Extraction de la prompte :** La prompte de texte est extraite du corps de la requête POST.
- **Encodage de la prompte :** La prompte est ensuite encodée en utilisant le tokenizer approprié. Le tokenizer convertit le texte en une séquence de tokens, qui est une forme que le modèle peut comprendre.
- **Génération de texte :** La prompte encodée est transmise au modèle, qui génère une réponse sous forme de texte.
- **Décodage de la réponse :** La réponse générée par le modèle est ensuite décodée en texte lisible par l'utilisateur.
- **Envoi de la réponse :** Enfin, l'API renvoie la réponse générée à l'utilisateur. La réponse est structurée sous forme de JSON pour faciliter son traitement par l'application cliente.

Le figure suivante présente l'architecture utilisée pour le développement de l'API :

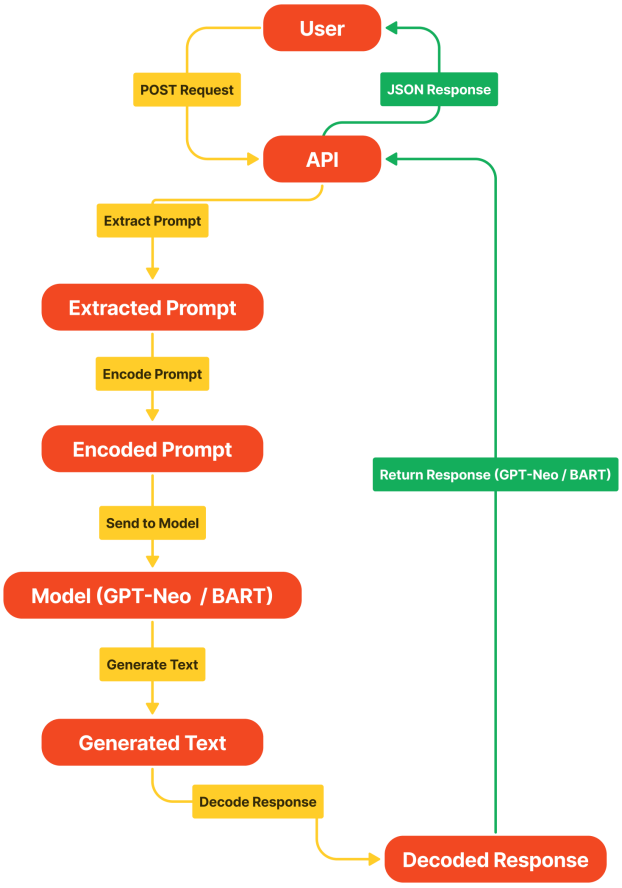


FIGURE 7.22 – L'architecture du développement de l'API.

7.4.3 Déploiement de l'API

Dans cette partie, nous avons opté pour l'utilisation d'**Amazon Web Services (AWS)**, une plateforme de services de cloud computing offrant une multitude de solutions pour le déploiement, la gestion et l'évolutivité de notre application. AWS, avec sa gamme étendue de services et sa présence mondiale, s'est imposé comme un choix naturel pour nous, compte tenu de sa fiabilité, de sa flexibilité et de sa capacité à s'adapter à des charges de travail de toutes tailles.

Parmi les nombreux services proposés par AWS, nous avons spécifiquement choisi d'utiliser **AWS Elastic Compute Cloud (EC2)** et **NGINX** pour le déploiement de notre API.

- **AWS EC2** est un service web qui fournit une capacité de calcul sécurisée et redimensionnable dans le cloud. Il est conçu pour faciliter le développement d'applications à grande échelle et à grande vitesse. L'adoption d'EC2 nous a permis de bénéficier d'une flexibilité accrue, nous permettant de choisir les ressources de calcul et le système d'exploitation qui correspondent le mieux à nos besoins, tout en nous offrant la possibilité de redimensionner ces ressources en fonction de l'évolution de nos besoins.
- **NGINX**, quant à lui, est un logiciel de serveur web open source qui peut également être utilisé comme un proxy inverse, un équilibreur de charge, un proxy de messagerie et un serveur de cache HTTP. NGINX est reconnu pour sa performance, sa stabilité et sa faible consommation de ressources. En l'intégrant dans notre architecture, nous avons pu optimiser la gestion des requêtes HTTP, améliorant ainsi la performance globale de notre API.

En exploitant les capacités d'Amazon Web Services (AWS) EC2 et de NGINX, nous avons réussi à établir une infrastructure solide et efficace pour le déploiement de notre API. Cette infrastructure, non seulement robuste, a démontré une performance exceptionnelle en termes de gestion des requêtes des utilisateurs et de fourniture de réponses précises en temps réel.

Le figure suivante présente les étapes du déploiement de l'API :

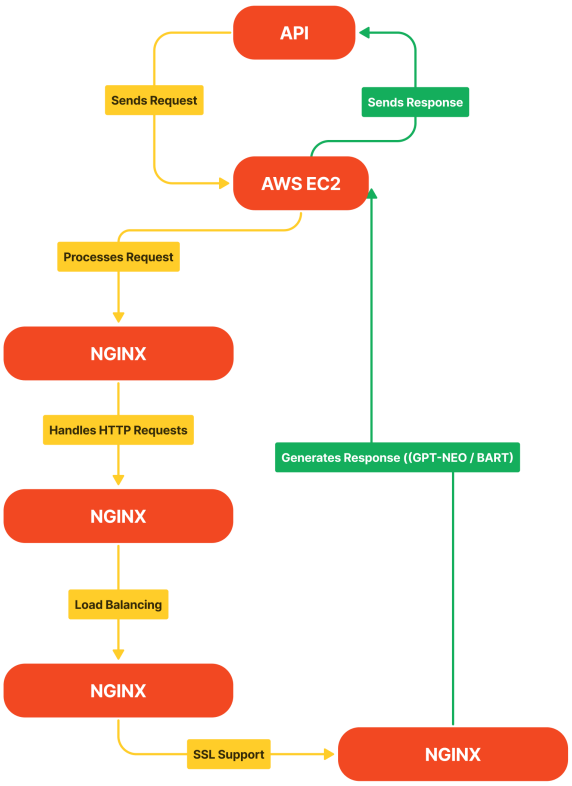


FIGURE 7.23 – Les étapes du déploiement de l'API.

7.4.4 Interface utilisateur

Notre plateforme se distingue par sa richesse en fonctionnalités et sa facilité d'utilisation. Elle héberge une bibliothèque exhaustive de rapports précédemment rédigés, offrant ainsi aux utilisateurs un accès instantané à une mine d'informations et de connaissances. De plus, elle intègre un système de prise de notes sophistiqué, permettant aux consultants de documenter efficacement leurs observations et leurs idées.

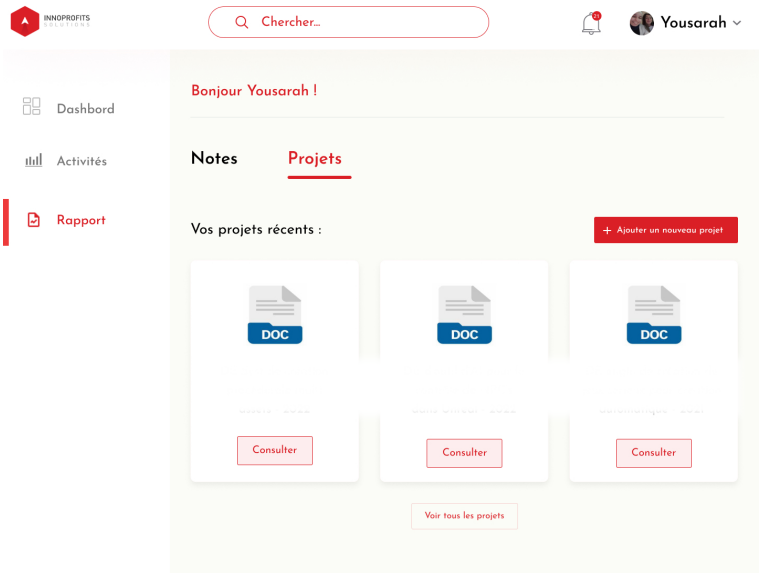


FIGURE 7.24 – Mur des rapports rédigés.

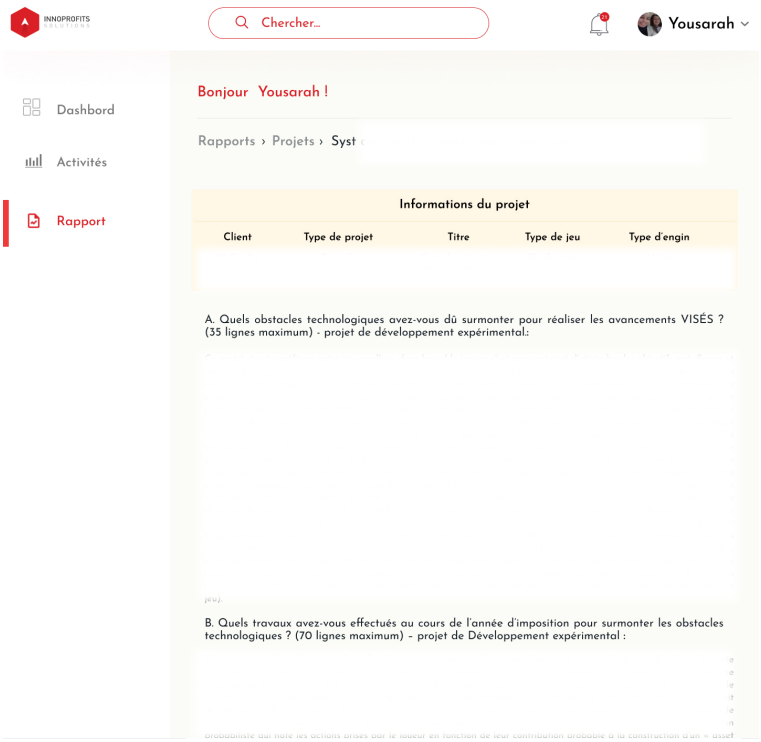


FIGURE 7.25 – Aperçu d'un exemple de rapport.

Dans le cadre d'un projet, le consultant peut définir l'objectif du projet, décrire les pratiques courantes et sélectionner des balises ou "tags" clés. Ces tags peuvent inclure le type du projet, les zones d'innovation, le type de jeu, l'engin utilisé et les technologies utilisées. Cette fonctionnalité de balisage permet une organisation et une récupération efficaces de l'information.

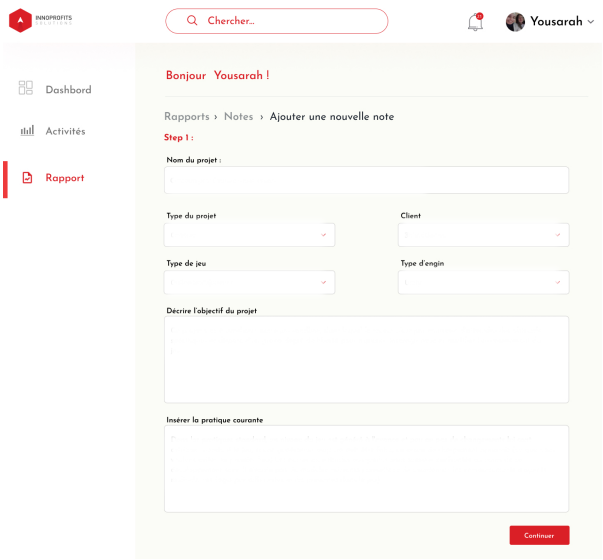


FIGURE 7.26 – Système de prise de notes - Step 01

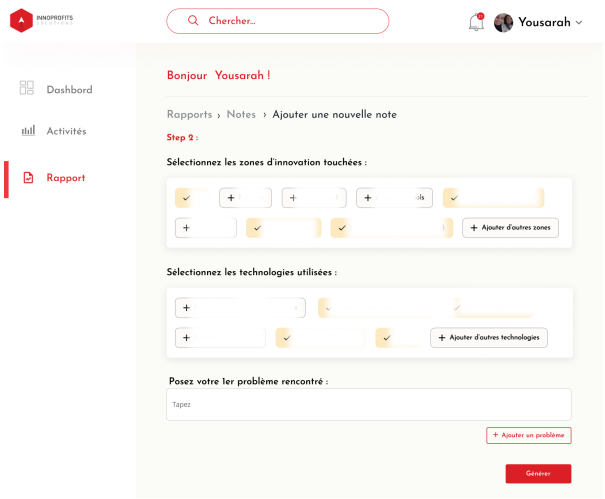


FIGURE 7.27 – Système de prise de notes - Step 02

Une fois ces informations saisies, le consultant a la possibilité de produire un ensemble d'incertitudes correspondant à ces données. À ce stade, l'administrateur peut sélectionner l'un des modèles disponibles, que ce soit GPT-Neo ou BART, pour le mettre en œuvre à l'aide de l'API. Ensuite, l'utilisateur introduit une sollicitation via l'interface utilisateur de la plateforme. Cette sollicitation est traitée par le modèle sélectionné, qui en retour génère une réponse textuelle.

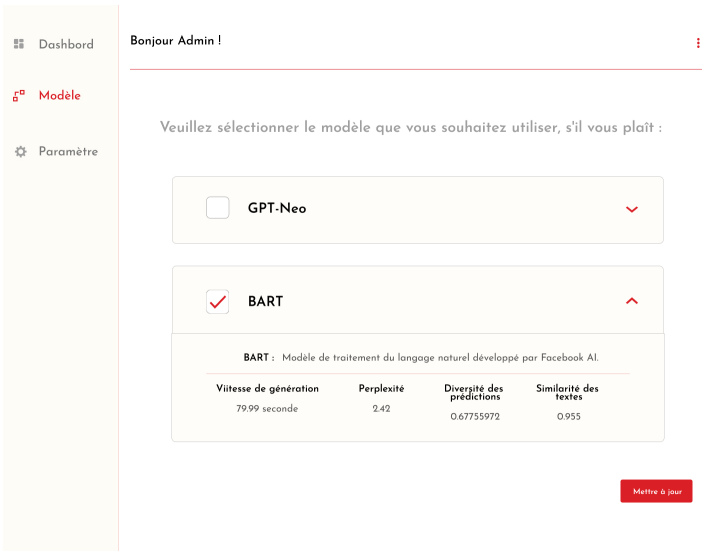


FIGURE 7.28 – Interface Admin.

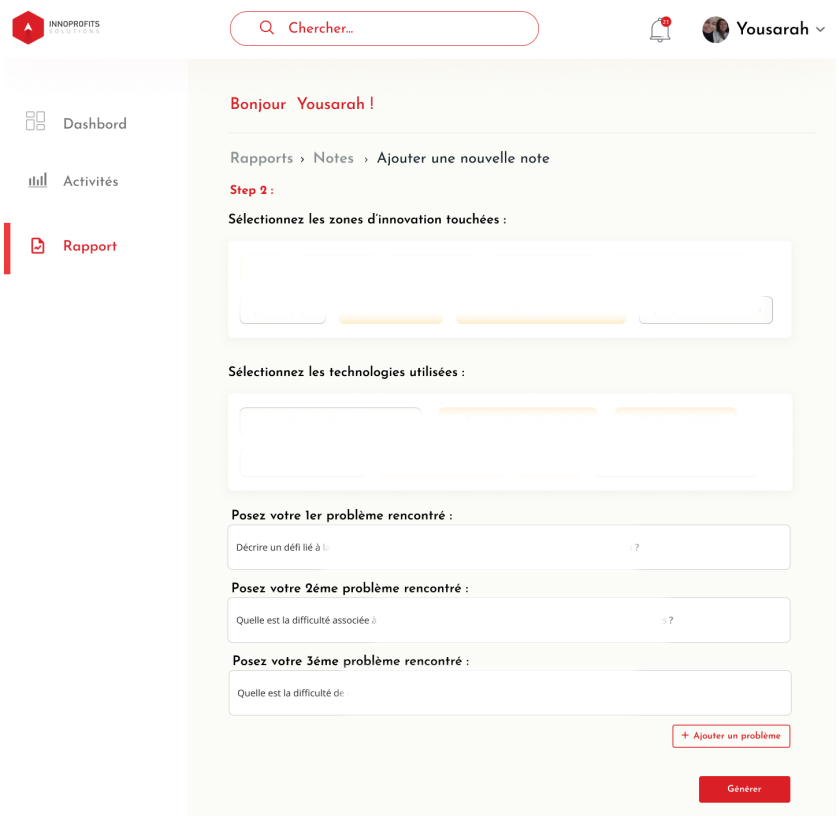


FIGURE 7.29 – Prompts.

7.4.5 Test de l'API

Nous avons effectué des tests sur notre API en utilisant diverses promptes pour évaluer sa performance. Par exemple, nous avons utilisé la prompte "Décrire un défi lié à la reconnaissance automatique des maisons construites par les joueurs ?", et l'API a produit une réponse détaillée et pertinente. Les résultats ont été positifs, montrant que l'API peut générer un texte pertinent rapidement. Nous avons identifié quelques domaines d'amélioration et avons apporté des modifications en conséquence. Des tests supplémentaires sont prévus pour l'avenir afin d'optimiser davantage la performance de l'API.

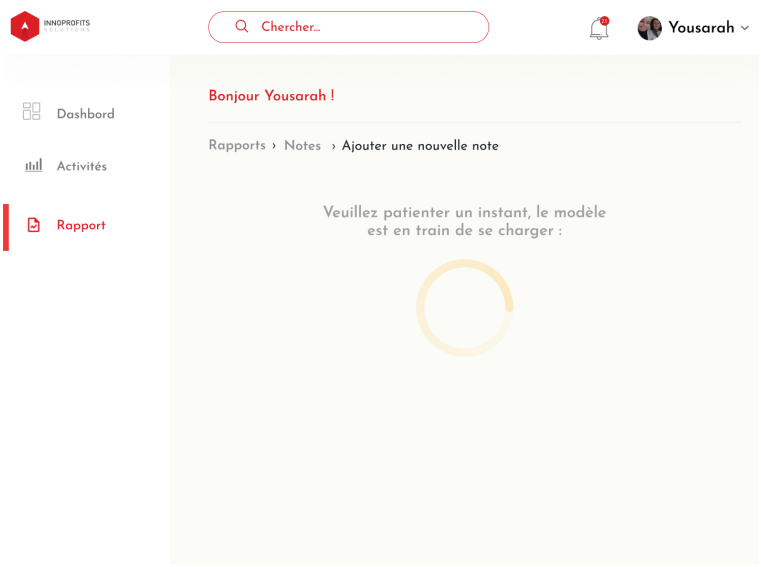


FIGURE 7.30 – Envoi de la requête au modèle.

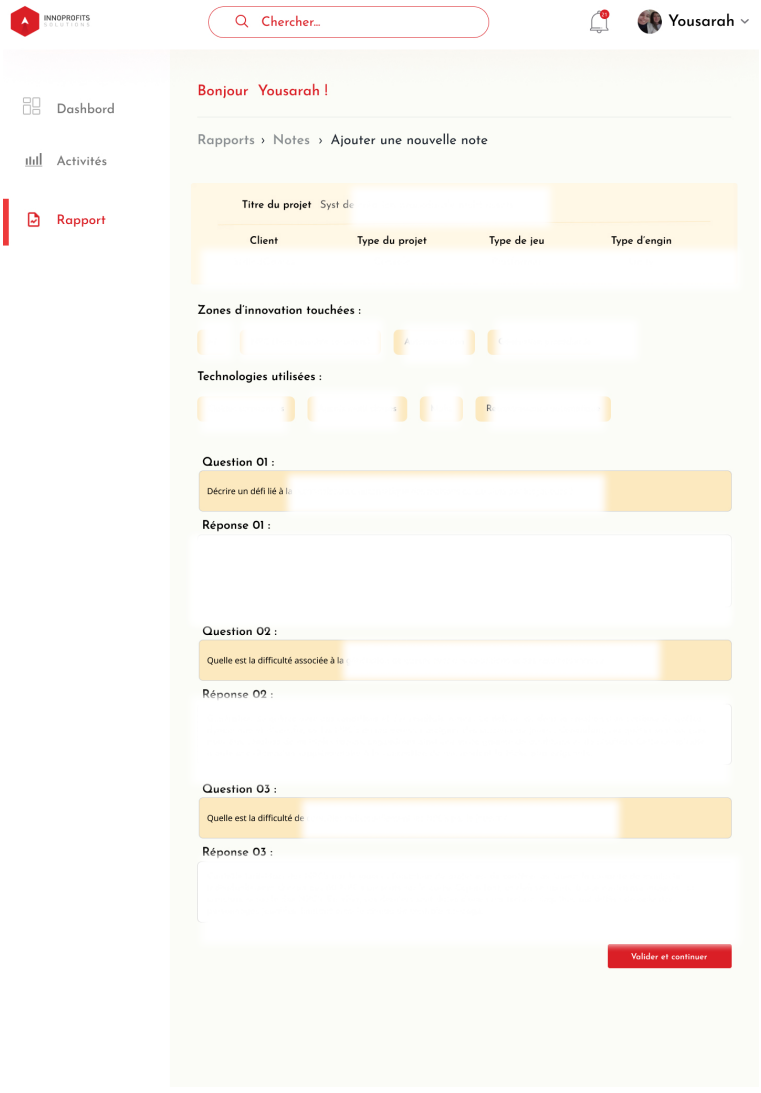


FIGURE 7.31 – Résultat des réponses générées.

7.5 Conclusion

Ce chapitre a permis de concrétiser la solution proposée en mettant en place les différentes étapes nécessaires à sa réalisation. La construction du dataset a été réalisée en identifiant et en collectant les rapports techniques pertinents, suivie d'une phase de nettoyage et de normalisation pour assurer la qualité des données. La centralisation des rapports dans Google Drive a facilité leur gestion et leur accessibilité. L'extraction d'informations pertinentes et la classification des rapports ont permis de structurer les données de manière efficace.

La modélisation a été abordée en sélectionnant les modèles adéquats et en procédant à leur implémentation. Des tests ont été effectués pour évaluer les performances des deux modèles, et les résultats ont été analysés de manière critique. Enfin, le déploiement de la solution a été réalisé en concevant une architecture appropriée, en développant une API fonctionnelle et en créant une interface utilisateur conviviale.

La réalisation de cette solution marque une étape importante dans la concrétisation des objectifs fixés. Les résultats obtenus témoignent de l'efficacité de la méthodologie adoptée et ouvrent de nouvelles perspectives pour l'exploitation de cette solution dans des contextes réels.

Conclusion générale

Dans ce mémoire, nous avons exploré le vaste et dynamique domaine de l'apprentissage profond, du traitement du langage naturel et de l'apprentissage par transfert, et comment ils peuvent être appliqués à l'amélioration du processus de rédaction des rapports techniques au sein de l'organisme MB inc. Nous avons d'abord établi l'état de l'art des techniques d'apprentissage profond et du traitement du langage naturel, en soulignant leurs avantages et défis dans l'automatisation du traitement de la langue.

Nous avons ensuite mené une étude détaillée sur l'organisme d'accueil, MB inc, en mettant l'accent sur ses processus internes et en identifiant des axes d'amélioration pour la rédaction des rapports techniques. Les techniques actuelles de rédaction sont devenues insuffisantes pour répondre aux exigences croissantes en matière de rapidité, de précision et de cohérence.

La solution proposée était une application du traitement du langage naturel et de l'apprentissage par transfert pour automatiser le processus de rédaction en générant automatiquement des descriptions techniques des projets. Nous avons conçu et mis en œuvre une solution innovante qui a montré des résultats prometteurs dans l'amélioration de la productivité et de l'efficacité de l'organisme.

Au cours de notre recherche, nous avons développé un modèle d'IA performant et adapté à cette tâche. Grâce à une structuration et à une organisation efficaces des données, nous avons pu améliorer l'efficacité et la qualité de la communication technique dans diverses industries. Par conséquent, nous avons réduit les coûts et les délais d'interprétation des documents textuels grâce à la génération automatique de textes précis.

Ce travail est une première étape vers l'intégration de l'IA dans le processus de génération de textes, plus particulièrement les descriptions techniques des projets dans les rapports techniques. Il reste encore beaucoup à faire pour optimiser et améliorer le système, y compris la généralisation de la solution sur les parties 02 et 03 des rapports techniques, l'expansion des capacités du système pour traiter une gamme plus large de tâches liées au rapport, ainsi que la mise en œuvre de mécanismes de contrôle de la qualité plus robustes.

Cela dit, malgré les progrès significatifs réalisés, il reste encore de nombreux défis à relever. Les modèles doivent être constamment améliorés pour s'adapter à l'évolution des langues et aux exigences spécifiques de chaque projet. En outre, il faut faire face à des problèmes persistants, tels que l'overfitting, qui peuvent limiter la généralisation de notre modèle à de nouvelles données. Néanmoins, les résultats obtenus à ce jour sont prometteurs et ouvrent la voie à des applications futures.

Perspectives

Au-delà de la génération de textes pour les descriptions techniques des projets dans les rapports, notre système basé sur GPT-Neo pourrait avoir une portée plus large. Il pourrait être utilisé dans de nombreux autres domaines, tels que la rédaction automatique de résumés de réunions, de rapports financiers ou de synthèses de recherche. Plus largement, on peut envisager l'intégration de notre modèle basé sur GPT-Neo à un outil de gestion de projets plus vaste, capable de produire automatiquement des descriptions de projets, des résumés de réunions, des bilans de progression et d'autres documents techniques pertinents. De plus, ces travaux posent des questions importantes sur la manière dont l'IA, notamment les modèles tels que GPT-3, peut transformer notre production et consommation de l'information technique. Les résultats obtenus jusqu'à présent avec GPT-Neo sont prometteurs, et l'utilisation de modèles plus avancés tels que GPT-3 pourrait ouvrir de nouvelles possibilités passionnantes pour cette recherche.

Bibliographie

Bibliographie

[1] "Crédits D'impôts | Innoprofits Solutions | Montréal." [Online]. Available : https://www.innoprofits.com

[2] "principe du transfer learning." [Online]. Available : https://www.bing.com/images/search?q=principe+du+transfer+learning&FORM=HDRSC3

[3] "bag of word representation." [Online]. Available : https://www.bing.com/images/search?q=bag+of+word+representation&FORM=HDRSC3

[4] "CBOW et Skip-Gram." [Online]. Available : https://www.bing.com/images/search?q=CBOW+et+Skip-Gram&FORM=HDRSC3

[5] "Réseau de neurones récurrents," Jun. 2023, page Version ID : 205030979. [Online]. Available : https://fr.wikipedia.org/w/index.php?title=R%C3%A9seau_de_neurones_r%C3%A9currents&oldid=205030979

[6] "types des RNN." [Online]. Available : https://www.bing.com/images/search?q=types+des+RNN&FORM=HDRSC3

[7] "l'architecture des transformers." [Online]. Available : https://www.bing.com/images/search?q=l%27architecture+des+transformers&FORM=HDRSC3

[8] "Accueil," Dec. 2013, last Modified : 2023-07-05. [Online]. Available : https://www.canada.ca/fr.html

[9] "Deep learning," Jun. 2023, page Version ID : 1160955543. [Online]. Available : https://en.wikipedia.org/w/index.php?title=Deep_learning&oldid=1160955543

[10] Olivier, "Introduction à la catégorisation de textes," Oct. 2020, section : Data Science. [Online]. Available : https://ledatascientist.com/introduction-a-la-categorisation-de-textes/

[11] abrecy, "Text Mining : Classification Automatique de textes," Apr. 2021. [Online]. Available : https://www.headmind.com/fr/text-mining-classification-automatique-de-textes/

[12] T. Iqbal and S. Qureshi, "The survey : Text generation models in deep learning," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 6, pp. 2515–2528, Jun. 2022. [Online]. Available : https://linkinghub.elsevier.com/retrieve/pii/S1319157820303360

[13] ReAct : Synergizing Reasoning and Acting in Language Models, Oct. 2022, oCLC : 1381572001. [Online]. Available : http://arxiv.org/abs/2210.03629

[14] J. Duan, H. Zhao, Q. Zhou, M. Qiu, and M. Liu, "A Study of Pre-trained Language Models in Natural Language Processing," in 2020 IEEE International Conference on Smart Cloud (SmartCloud). Washington DC, WA, USA : IEEE, Nov. 2020, pp. 116–121. [Online]. Available : https://ieeexplore.ieee.org/document/9265932/

[15] A. Koroleva, S. Kamath, and P. Paroubek, "Measuring semantic similarity of clinical trial outcomes using deep pre-trained language representations," Journal of Biomedical Informatics, vol. 100, p. 100058, 2019. [Online]. Available : https://linkinghub.elsevier.com/retrieve/pii/S2590177X19300575

[16] Modern Methods for Text Generation, Sep. 2020, oCLC : 1228431547. [Online]. Available : http://arxiv.org/abs/2009.04968

[17] R. Kansal, A. Li, J. Duarte, N. Chernyavskaya, M. Pierini, B. Orzari, and T. Tomei, "Evaluating generative models in high energy physics," Physical Review D, vol. 107, no. 7, p. 076017, Apr. 2023. [Online]. Available : https://link.aps.org/doi/10.1103/PhysRevD.107.076017

[18] ChatGPT Outperforms Crowd-Workers for Text-Annotation Tasks, Mar. 2023, oCLC : 1381613368. [Online]. Available : http://arxiv.org/abs/2303.15056

[19] https ://www.facebook.com/linformatique, "Découvrez BARD : la réponse de Google à ChatGPT," section : Actualités. [Online]. Available : https://www.linformatique.org/bard-google-chatgpt.html

[20] E. Raffin, "LegiGPT : le chatbot français qui répond à vos questions juridiques," May 2023, section : Tech. [Online]. Available : https://www.blogdumoderateur.com/legigpt-chatbot-francais-repond-questions-juridiques/

[21] T. rédac, "Recurrent Neural Network (RNN) : de quoi s'agit-il ?" Jul. 2021. [Online]. Available : https://datascientest.com/recurrent-neural-network

[22] "Aller plus loin en deep learning avec les réseaux de neurones récurrents (RNNs)." [Online]. Available : https://france.devoteam.com/paroles-dexperts/aller-plus-loin-en-deep-learning-avec-les-reseaux-de-neurones-recurrents-rnns/

[23] T. Keldenich, "RNN - Comprendre rapidement les Réseaux de Neurones Récurrents," Feb. 2021. [Online]. Available : https://inside-machinelearning.com/les-reseaux-de-neurones-recurrents-rnn/

[24] Y. Poiron, "Qu'est-ce que Google Bard, et comment l'utiliser ?" Feb. 2023. [Online]. Available : https://www.blog-nouvelles-technologies.fr/253587/quest-ce-que-google-bard-et-comment-utiliser/

[25] "Les Transformers, incontournables du Deep Learning." [Online]. Available : https://blent.ai/blog/a/transformers-deep-learning

[26] "EleutherAI/gpt-neo-2.7B · Hugging Face." [Online]. Available : https://huggingface.co/EleutherAI/gpt-neo-2.7B

[27] "What is transfer learning in NLP ?" [Online]. Available : https://www.isahit.com/blog/what-is-transfer-learning-in-nlp

[28] "What Is Transfer Learning ? A Guide for Deep Learning | Built In." [Online]. Available : https://builtin.com/data-science/transfer-learning

[29] "Transfer learning : définition, exemples, fine tuning..." Oct. 2022. [Online]. Available : https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501859-transfer-learning/

[30] T. rédac, "Transfer Learning : Qu'est-ce que c'est ?" Jul. 2020. [Online]. Available : https://datascientest.com/transfer-learning

Annexes

Annexe A

Les bibliothèques utilisées

A.1 Présentation de Python

Python est un langage de programmation à haut niveau, interprété et orienté objet. Il a été créé par Guido van Rossum et a été publié pour la première fois en 1991. Depuis lors, il est devenu l'un des langages de programmation les plus populaires et les plus largement utilisés.

Ce langage est apprécié pour sa lisibilité et son code clair. Il est souvent décrit comme un langage qui est aussi proche que possible de l'anglais simple, ce qui rend son code particulièrement facile à lire et à comprendre.

Un autre avantage majeur de Python est sa polyvalence. Il est utilisé dans un large éventail de domaines, y compris le développement web, le développement de logiciels, l'analyse de données, l'apprentissage automatique, l'intelligence artificielle, la science des données, la bio-informatique, le calcul scientifique et bien d'autres.

L'usage de Python est très répandu dans le domaine du Traitement Automatique du Langage Naturel, notamment grâce à sa lisibilité, sa simplicité d'utilisation et la variété de ses bibliothèques dédiées. Les applications de Python en NLP vont de la génération de texte à la traduction automatique, en passant par l'analyse de sentiments et le résumé automatique de textes.

A.2 Présentation des bibliothèques

Les bibliothèques en programmation sont des ensembles d'outils, de modules, de méthodes et de classes qui étendent les fonctionnalités du langage utilisé. La plupart des bibliothèques disponibles en Python sont gratuites et open source. Dans cette section, nous allons présenter les bibliothèques utilisées dans ce projet.

A.2.1 NLTK (Natural Language Toolkit)

NLTK est l'une des bibliothèques NLP les plus connues pour Python. Elle offre des fonctionnalités pour le fractionnement des textes, le marquage grammatical, l'analyse syntaxique et la reconnaissance d'entités nommées.

A.2.2 SpaCy

SpaCy est une bibliothèque NLP moderne, rapide et robuste qui fournit de nombreux modèles pré-entraînés pour de multiples langues. SpaCy est particulièrement utile pour des tâches comme l'extraction d'information, la reconnaissance d'entités nommées et la dépendance syntaxique. Il existe un certain nombre d'étapes importantes dans les relations de SpaCy avec les relations linguistiques :

- Téléchargez et appelez la bibliothèque
- Construire un pipeline
- Utilisation de jetons (tokens)
- Choix des mots spécifiques (Tagging)
- Comprendre les jetons d'attributs (Attributes Tokens)

Ce dessin montre le chemin emprunté par spacy, et voici le pipeline :

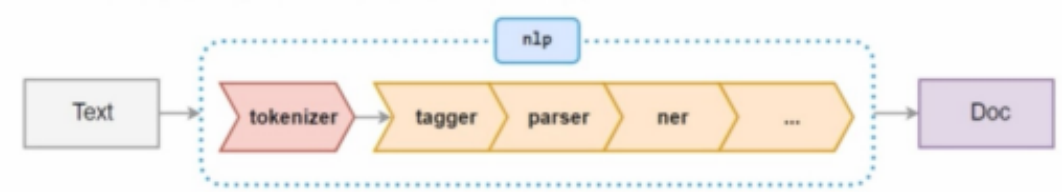


FIGURE 1.1 – Pipeline SpaCy

Au début, le texte est saisi, puis toutes les opérations NLP sont effectuées, telles que le tokenizer, puis le tagger, puis la matrice d'analyseur, suivi du processus de reconnaissance d'objet appelé Recognition Entity Named (NER), ainsi que Stemming, POS, lemmatisation et autres.

A.2.3 Gensim

Gensim est une bibliothèque spécialisée dans le modelage des sujets et la similarité des documents. Elle est utile pour construire des modèles de word embedding comme Word2Vec, FastText et Doc2Vec.

A.2.4 Transformers (de Hugging Face)

Cette bibliothèque offre une collection de modèles pré-entraînés pour des tâches de NLP basées sur des architectures de type "Transformer", comme BERT, GPT-2, T5, et bien d'autres. Ces modèles sont extrêmement utiles pour des tâches de génération de texte, de traduction, de résumé automatique

A.2.5 TensorFlow

TensorFlow est une bibliothèque open source d'apprentissage automatique développée par Google Brain. Elle permet aux développeurs de construire et de déployer des applications d'apprentissage automatique, d'apprentissage en profondeur et de traitement du langage naturel. Elle est

principalement utilisée pour la recherche et la production en matière d'apprentissage automatique et a un soutien pour les réseaux de neurones profonds et autres algorithmes de machine learning.

A.2.6 PyTorch

PyTorch est une bibliothèque d'apprentissage en profondeur pour le langage de programmation Python, développée principalement par l'équipe de recherche en intelligence artificielle de Facebook. Il est connu pour sa flexibilité et sa facilité d'utilisation, offrant un contrôle dynamique de la structure des réseaux de neurones et une intégration forte avec le reste de l'écosystème Python.

A.2.7 Regex Tester

La bibliothèque Regex Tester est un outil utilisé pour tester et expérimenter les expressions régulières (regex). Les expressions régulières sont des motifs de recherche utilisés pour trouver des correspondances dans du texte en fonction de motifs spécifiques.

La bibliothèque Regex Tester offre une interface conviviale où on peut saisir notre expression régulière et voir instantanément les correspondances dans le texte de test. Elle permet d'itérer rapidement sur les expressions régulières et de voir comment elles fonctionnent.

En utilisant la bibliothèque Regex Tester, on peut :

- Saisir notre expression régulière dans une zone de texte dédiée.
- Entrer le texte de test pour voir les correspondances avec notre expression régulière.
- Visualiser les résultats de la correspondance, généralement avec des surbrillances ou des mises en évidence.
- Tester et déboguer nos expressions régulières en temps réel, en modifiant et en adaptant les motifs pour obtenir les résultats souhaités.
- Expérimenter avec les différentes options et métacaractères des expressions régulières pour mieux comprendre leur fonctionnement.

La bibliothèque Regex Tester est un outil précieux pour les développeurs, les testeurs et toute personne travaillant avec des expressions régulières. Elle permet de gagner du temps et d'assurer la précision des motifs de recherche.

A.2.8 Numpy

NumPy est une bibliothèque Python qui fournit un support pour les grands tableaux et matrices multidimensionnelles, ainsi que des fonctions mathématiques de haut niveau pour opérer sur ces structures de données. Elle est très importante dans la recherche scientifique et les domaines connexes pour une manipulation efficace des données et est largement utilisée dans diverses applications mathématiques.

A.2.9 Pandas

Pandas est une bibliothèque de manipulation et d'analyse de données open-source, rapide, puissante, flexible et facile à utiliser, construite sur NumPy. Pandas introduit des structures de données comme les DataFrame et les Series qui sont conçues pour manipuler des données tabulaires et autres formes de données structurées de manière plus efficace.

A.2.10 Scikit-learn (sklearn)

Scikit-learn est une bibliothèque d'apprentissage automatique pour Python. Elle offre des algorithmes d'apprentissage supervisés et non supervisés, y compris la classification, la régression, le clustering et la réduction de dimensionnalité. De plus, elle fournit des outils pour l'ajustement de modèle, le prétraitement de données, la sélection et évaluation de modèle, et bien d'autres.

A.2.11 Matplotlib

Matplotlib est une bibliothèque de visualisation de données en Python. Elle fournit des interfaces pour créer des graphiques statiques, animés et interactifs en Python. Matplotlib est très flexible et peut produire une grande variété de visualisations, y compris des histogrammes, des graphiques à barres, des graphiques de dispersion, et bien plus encore.

Annexe B

Outils et technologies informatiques utilisés

B.1 Dropbox

Dropbox est un service de stockage de fichiers basé sur le cloud qui a été lancé en 2007. Il permet aux utilisateurs de stocker leurs fichiers en ligne et de les synchroniser avec différents appareils. En d'autres termes, une fois qu'un fichier est enregistré dans Dropbox, il sera accessible à partir de n'importe quel ordinateur, smartphone ou tablette connecté à Internet.

Dropbox fonctionne comme un dossier sur votre ordinateur, mais avec quelques différences majeures. Tout ce que qui est placé dans ce dossier est automatiquement sauvegardé dans le cloud et synchronisé avec tous les appareils sur lesquels vous avez installé Dropbox. Cela signifie que les fichiers sont accessibles n'importe où, n'importe quand, à condition d'avoir une connexion Internet.

En plus de la synchronisation des fichiers, Dropbox propose plusieurs autres fonctionnalités utiles. Par exemple, il permet le partage de fichiers et de dossiers, ce qui est idéal pour la collaboration en équipe. Il propose également une fonction de sauvegarde et de restauration, qui peut aider à récupérer des fichiers perdus ou accidentellement supprimés. De plus, il offre une certaine quantité de stockage gratuit, avec la possibilité d'acheter plus d'espace si nécessaire.

Dropbox peut être utilisé pour une variété d'applications, y compris le stockage de documents. Il est également couramment utilisé pour le partage de fichiers, la collaboration sur des projets, la sauvegarde de données importantes et la synchronisation de fichiers entre différents appareils.

B.2 API

Une API, ou Interface de Programmation d'Applications (Application Programming Interface en anglais), est un ensemble de règles et de spécifications que les logiciels peuvent suivre pour communiquer entre eux. Elle sert d'interface entre différents logiciels afin qu'ils puissent interagir et échanger des données.

Les APIs sont utilisées partout dans la programmation informatique et elles permettent aux différentes composantes d'un logiciel de communiquer entre elles. Par exemple, lorsque vous utilisez une application sur votre téléphone mobile pour consulter la météo, l'application utilise probablement une API pour récupérer les données météorologiques d'un service sur internet.

Elles peuvent être utilisées pour accéder aux fonctionnalités d'un autre logiciel ou service. Par exemple, une entreprise peut utiliser l'API de Facebook pour permettre aux utilisateurs de se connecter à son site web en utilisant leurs identifiants Facebook.

B.3 Google Suite

Google Workspace, anciennement connu sous le nom de "G Suite", est une collection d'outils et de services de productivité et de collaboration basés sur le cloud, développés par Google, destinés à faciliter la collaboration et la productivité au sein des entreprises et des organisations. Ces services incluent :

B.3.1 Google Drive

Google Drive est un service de stockage et de synchronisation de fichiers basé sur le cloud lancé par Google en avril 2012. Il permet aux utilisateurs de stocker des fichiers sur leurs serveurs, de synchroniser des fichiers sur différents appareils et de partager des fichiers.

- **Stockage en ligne :** Google Drive offre un espace de stockage gratuit de 15 Go par défaut pour chaque utilisateur. Cet espace est utilisé pour sauvegarder différents types de fichiers tels que des documents, des images, des vidéos, des présentations, des fichiers audio, etc.
- **Synchronisation multiplateforme :** L'utilisateur peut installer l'application Google Drive sur son ordinateur (Windows ou macOS) ou son appareil mobile (Android ou iOS) pour synchroniser automatiquement ses fichiers entre son appareil et le cloud. Cela signifie qu'il peut accéder à ses fichiers depuis n'importe quel appareil connecté à Internet.
- **Partage de fichiers :** L'utilisateur peut partager ses fichiers et ses dossiers avec d'autres utilisateurs en leur accordant des autorisations spécifiques, telles que la lecture seule ou la possibilité de modifier. Cela facilite la collaboration et le travail en équipe, car plusieurs utilisateurs peuvent accéder et travailler sur les mêmes fichiers simultanément.
- **Collaboration en temps réel :** Avec Google Drive, on collaborer en temps réel sur des documents, des feuilles de calcul et des présentations grâce aux applications Google Docs, Sheets et Slides intégrées. Cela permet à plusieurs utilisateurs de travailler ensemble sur un même fichier, d'apporter des modifications et de voir les mises à jour en temps réel.
- **Organisation des fichiers :** Google Drive permet de créer des dossiers et de les organiser de manière logique pour une meilleure gestion des fichiers. On peut également ajouter des étiquettes et des couleurs aux fichiers pour les classer et les retrouver plus facilement.
- **Recherche avancée :** Google Drive dispose d'une fonctionnalité de recherche puissante qui permet de trouver rapidement des fichiers en fonction de leur nom, de leur contenu ou des propriétés associées. On peut également appliquer des filtres et des options de recherche avancées pour affiner nos résultats.
- **Intégrations avec d'autres applications :** Google Drive s'intègre avec de nombreuses autres applications Google, telles que Google Docs, Sheets, Slides, Gmail, Google Photos, etc. Cela facilite le partage de fichiers et la collaboration entre différentes applications.
- **Versioning et historique des fichiers :** Google Drive enregistre automatiquement les différentes versions d'un fichier, permettant de restaurer des versions précédentes si nécessaire. Il conserve également un historique des modifications apportées à un fichier, ce qui permet de suivre les changements et les contributions des utilisateurs.

- **Sécurité des fichiers :** Google Drive assure la sécurité des fichiers en utilisant le cryptage des données en transit et au repos. On peut également définir des autorisations d'accès et des paramètres de confidentialité pour protéger les fichiers.
- **Applications tierces et API :** Google Drive offre une API qui permet aux développeurs d'intégrer des fonctionnalités de stockage et de gestion des fichiers dans leurs propres applications. Il existe également de nombreuses applications tierces compatibles avec Google Drive pour étendre ses fonctionnalités.

Google Drive API

L'API Google Drive est une interface de programmation d'application (API) fournie par Google, qui permet aux développeurs d'interagir avec Google Drive et d'intégrer ses fonctionnalités de stockage et de gestion des fichiers dans leurs propres applications. Pour utiliser l'API Google Drive, voici les étapes générales à suivre :

- **Création d'un projet dans la Console API Google :** Accéder à la Console API Google (https ://console.developers.google.com/) et créer un nouveau projet. Activer l'API Google Drive pour ce projet.
- **Configuration des identifiants d'authentification :** Dans la Console API Google, créer des identifiants d'authentification pour le projet. Sélectionner le type d'identifiant approprié pour l'application, tel qu'un identifiant de compte de service, un ID client OAuth ou un ID Web client. Cela dépendra du type d'application développé.
- **Autorisation l'accès à l'API Google Drive :** Obtenir les clés d'API nécessaires pour autoriser l'accès à l'API Google Drive dans l'application. Cela implique généralement d'obtenir un jeton d'accès valide en utilisant le flux d'authentification OAuth 2.0.
- **Intégration de l'API dans l'application :** Utiliser les bibliothèques client et les exemples de code fournis par Google pour intégrer l'API Google Drive dans l'application. Utiliser des langages de programmation tels que Python, Java, JavaScript, etc., en fonction des préférences et de l'environnement de développement.
- **Interaction avec l'API Google Drive :** Utiliser les méthodes et les fonctionnalités fournies par l'API pour créer, lire, mettre à jour et supprimer des fichiers, gérer les autorisations d'accès, effectuer des recherches, suivre les modifications, etc. Se référencier à la documentation de l'API Google Drive pour obtenir des informations détaillées sur les opérations disponibles.
- **Gestion des erreurs et les exceptions :** S'assurer de gérer les erreurs et les exceptions qui peuvent se produire lors de l'utilisation de l'API Google Drive. Cela inclut la gestion des problèmes d'authentification, des limites d'utilisation de l'API, des conflits de fichiers, etc.
- **Test et déploiement de l'application :** Tester l'application pour s'assurer que les fonctionnalités de l'API Google Drive sont correctement intégrées et fonctionnent comme prévu. Une fois satisfait, l'utilisateur peut déployez son application pour permettre aux utilisateurs d'interagir avec l'API Google Drive.

B.3.2 Google Docs

Google Docs est une application en ligne proposée par Google qui permet aux utilisateurs de créer, modifier et collaborer sur des documents. Il fait partie de la suite d'outils de productivité

Google Drive, qui comprend également Google Sheets (pour les feuilles de calcul) et Google Slides (pour les présentations). Voici quelques fonctionnalités clés de Google Docs :

- **Édition et mise en forme du texte :** Avec Google Docs, on peut créer différents types de documents tels que des lettres, des rapports, etc. Il offre de nombreuses options de formatage, notamment le formatage du texte, les titres, les listes à puces et numérotées. On peut également insérer des images, des tableaux et des liens dans nos documents.
- **Collaboration en temps réel :** Google Docs permet à plusieurs utilisateurs de travailler simultanément sur un document. Les modifications apportées par un utilisateur sont visibles en temps réel pour les autres participants, facilitant ainsi la collaboration et la co-édition.
- **Synchronisation et sauvegarde automatiques :** Les documents créés dans Google Docs sont enregistrés automatiquement pendant que vous travaillez. Ils sont stockés dans le cloud, ce qui permet d'y accéder à partir de n'importe quel appareil avec une connexion Internet.
- **Contrôle des versions :** Google Docs enregistre automatiquement l'historique des modifications apportées à un document, ce qui vous permet de revenir à des versions précédentes si nécessaire. Vous pouvez également consulter l'historique des modifications et voir qui a apporté quelles modifications.
- **Partage et autorisations :** Vous pouvez partager vos documents avec d'autres utilisateurs en leur accordant des autorisations spécifiques, telles que la lecture seule, la modification ou le commentaire. Vous pouvez également contrôler l'accès en définissant des mots de passe ou en limitant l'accès aux utilisateurs spécifiques.
- **Intégration avec d'autres outils Google :** Google Docs s'intègre parfaitement avec d'autres outils Google tels que Google Drive, Google Sheets et Google Slides. Vous pouvez importer des données à partir de ces outils, collaborer sur des projets et organiser vos fichiers de manière cohérente.
- **Accessibilité hors ligne :** Vous pouvez accéder à vos documents Google Docs et les éditer même lorsque vous n'êtes pas connecté à Internet. Les modifications effectuées hors ligne seront synchronisées une fois que vous serez à nouveau en ligne.
- **Exportation et formats de fichiers :** Vous pouvez exporter vos documents Google Docs dans différents formats, tels que Microsoft Word, PDF, texte brut, etc. Cela vous permet de partager vos documents avec des utilisateurs qui n'utilisent pas nécessairement Google Docs.

En résumé, Google Docs est un outil populaire et puissant pour la création et la collaboration sur des documents en ligne, offrant praticité, flexibilité et facilité d'utilisation.

B.3.3 Google Sheets

Google Sheets est une application en ligne développée par Google, qui permet de créer, modifier et collaborer sur des feuilles de calcul. Il fait partie de la suite d'outils de productivité Google Drive, aux côtés de Google Docs et Google Slides. Google Sheets offre de nombreuses fonctionnalités pour faciliter la manipulation et l'analyse des données dans un navigateur Web :

- **Création de tableaux :** On peut créer des tableaux dans Google Sheets pour organiser vos données en colonnes et lignes. Chaque colonne peut représenter un champ différent dans votre base de données.

- **Fonctions et formules :** Google Sheets propose une large gamme de fonctions et de formules intégrées qui permettent d'effectuer des calculs, des manipulations de données et des opérations logiques. On peut utiliser ces fonctions pour effectuer des requêtes et des filtrages sur nos données.
- **Filtrage et tri :** Google Sheets permet de filtrer et de trier facilement les données en fonction de certains critères. Cela peut aider à extraire des enregistrements spécifiques ou à trier les données selon un ordre particulier.
- **Importation et exportation de données :** On peut importer des données à partir d'autres sources dans Google Sheets, par exemple à partir d'un fichier CSV ou d'une base de données externe. De plus, on peut exporter les données dans différents formats pour les utiliser dans d'autres applications.
- **Collaboration en temps réel :** Comme pour les autres produits Google, on peut collaborer en temps réel avec d'autres utilisateurs sur une feuille de calcul. Cela permet à plusieurs personnes de travailler simultanément sur la base de données, de mettre à jour les enregistrements et de voir les modifications en temps réel.

En résumé, Google Sheets est un outil puissant et polyvalent pour créer, gérer et analyser des données dans des feuilles de calcul. Il offre la commodité du stockage dans le cloud et la collaboration en temps réel, ce qui en fait un choix populaire pour les individus, les équipes et les entreprises.

B.3.4 Google Keep

Google Keep est une application de prise de notes développée par Google. Elle est conçue pour aider les utilisateurs à capturer, organiser et partager leurs idées, rappels et listes de tâches sur plusieurs appareils. Voici une liste de fonctionnalités clés de Google Keep :

- **Prise de notes :** On peut créer des notes textuelles et ajouter des images à nos notes.
- **Listes de contrôle :** Google Keep permet de créer des listes de tâches avec des cases à cocher pour aider à suivre les progrès.
- **Rappels :** On peut définir des rappels pour nos notes, ce qui nous permet de recevoir des notifications à une date et une heure spécifiques.
- **Synchronisation multiplateforme :** Nos notes sont automatiquement synchronisées entre nos appareils, ce qui nous permet d'y accéder depuis n'importe où.
- **Couleurs et étiquettes :** On peut personnaliser l'apparence de nos notes en leur attribuant différentes couleurs. De plus, on peut ajouter des étiquettes pour organiser nos notes en catégories.
- **Recherche :** Google Keep dispose d'une fonction de recherche intégrée qui nous permet de trouver rapidement des notes en fonction de mots-clés.
- **Collaborer :** On peut partager des notes avec d'autres utilisateurs, ce qui nous permet de travailler en collaboration sur des idées ou des tâches.
- **Intégration avec d'autres services Google :** Google Keep s'intègre avec d'autres services Google tels que Google Docs et Google Agenda, nous permettant d'importer des notes dans des documents ou de synchroniser des rappels avec notre calendrier.

- **Accessibilité hors ligne :** On peut accéder à nos notes même lorsqu'on n'a pas de connexion Internet active, et les modifications seront synchronisées une fois qu'on sera à nouveau en ligne.
- **Capture vocale :** On peut enregistrer des notes vocales à l'aide de notre microphone, ce qui nous permet de capturer rapidement des idées à l'oral.

  Ces fonctionnalités font de Google Keep une application pratique et polyvalente pour la prise de notes, la gestion des tâches et la collaboration.

B.3.5 Google Meet

Google Meet est une plateforme de visioconférence en ligne développée par Google. Elle permet aux utilisateurs de tenir des réunions virtuelles, des appels vidéo et des sessions de collaboration en temps réel.

Google Meet est devenu un outil essentiel pour les réunions à distance et le travail d'équipe virtuel. Sa simplicité d'utilisation, ses fonctionnalités de collaboration et sa capacité à accueillir de grandes réunions en font une solution populaire pour les communications en ligne.

B.3.6 Google Agenda

Google Agenda est un service de gestion des calendriers en ligne développé par Google. Il permet aux utilisateurs de planifier, d'organiser et de partager des événements, des réunions et des rappels.

Google Agenda est un outil pratique pour gérer votre emploi du temps, planifier des réunions, des événements personnels et professionnels, et rester organisé. Sa simplicité d'utilisation, ses fonctionnalités de partage et son intégration avec d'autres services Google en font un calendrier populaire pour de nombreux utilisateurs.

B.4 Jupyter

Jupyter est un environnement de développement interactif open-source qui permet de créer, exécuter et partager du code dans des notebooks. Les notebooks Jupyter sont des documents qui peuvent contenir du code exécutable, des visualisations, des textes explicatifs et des éléments multimédias, le tout combiné dans un seul environnement. Voici quelques caractéristiques clés de Jupyter :

- **Support de plusieurs langages de programmation :** Jupyter prend en charge plusieurs langages de programmation, notamment Python, R, Julia et bien d'autres. Cela permet aux utilisateurs d'écrire et d'exécuter du code dans le langage de leur choix, directement dans les cellules des notebooks.
- **Exécution interactive du code :** Les notebooks Jupyter permettent d'exécuter du code de manière interactive, cellule par cellule. Cela facilite l'exploration des données, les expérimentations et les analyses itératives. Les résultats des calculs et les visualisations sont affichés en temps réel à mesure que vous exécutez les cellules.
- **Intégration de documentation :** Les notebooks Jupyter permettent d'inclure des textes explicatifs, des titres, des sous-titres et des commentaires au sein des cellules de texte. Cela

permet de fournir des explications détaillées, de documenter le code et de partager des informations contextuelles avec d'autres utilisateurs.

- **Visualisations interactives :** Jupyter offre des bibliothèques et des outils de visualisation, tels que Matplotlib, Seaborn, Plotly, qui permettent de créer des graphiques, des diagrammes, des tableaux de bord interactifs et d'autres représentations visuelles directement dans les notebooks.

En résumé, Jupyter est un environnement de développement interactif populaire qui facilite l'écriture, l'exécution et le partage de code, ainsi que la création de documents combinant du code, des visualisations et des explications. C'est un outil polyvalent utilisé par de nombreux développeurs, scientifiques et analystes de données pour explorer, analyser et communiquer des informations.

B.5 Figma

Figma est une plateforme de conception d'interfaces utilisateur (UI) basée sur le cloud. Elle offre un éditeur de conception en temps réel qui permet aux utilisateurs de créer, modifier et collaborer sur des maquettes, des prototypes et des designs interactifs directement dans le navigateur. Figma propose des outils puissants pour la conception d'interfaces utilisateur, tels que des outils de dessin, de typographie et de création de formes. Il permet également de créer des prototypes interactifs en reliant les différentes pages ou écrans du design. La collaboration est facilitée grâce à la possibilité de travailler simultanément sur un même fichier, de laisser des commentaires et de réviser les designs en temps réel. Figma propose également des bibliothèques de composants réutilisables, un historique des versions et des intégrations avec d'autres outils et services. Il est largement utilisé par les designers et les équipes de conception pour créer des interfaces utilisateur modernes et collaboratives.

B.6 GPU

Un GPU (Graphical Processing Unit) est un processeur spécialisé conçu pour accélérer les calculs graphiques, les affichages et les tâches parallèles. Grâce à leur architecture parallèle, les GPU peuvent effectuer simultanément de nombreux calculs indépendants, offrant ainsi des performances élevées dans des domaines tels que les jeux vidéo, la modélisation 3D, l'apprentissage automatique et les calculs scientifiques. Les fabricants comme NVIDIA et AMD proposent des frameworks et des langages de programmation spécifiques (comme CUDA et OpenCL) pour exploiter les capacités de calcul parallèle des GPU. Les GPU sont devenus essentiels pour l'accélération des calculs graphiques, l'apprentissage automatique et les calculs parallèles, et sont devenus un élément clé dans de nombreux domaines nécessitant des calculs intensifs et une accélération des performances.

CUDA

CUDA (Compute Unified Device Architecture) est une interface de programmation d'applications (API) créée par Nvidia. Elle permet aux développeurs de tirer parti des capacités de calcul des cartes graphiques (GPU) pour des tâches autres que le rendu graphique.

En exploitant les GPU, CUDA permet d'accélérer considérablement les calculs en parallèle, rendant les GPU utiles pour une variété de tâches de calcul intensif, comme l'apprentissage pro-

fond (deep learning), le traitement d'image, l'ingénierie assistée par ordinateur, les simulations numériques et bien d'autres encore.

CUDA fournit un ensemble de bibliothèques et de compilateurs qui permettent aux développeurs d'écrire des programmes en C, C++, Python et Fortran qui exécutent des portions de code (appelées kernels) sur le GPU.

CUDA est largement utilisé dans la recherche et l'industrie, notamment dans le domaine de l'apprentissage automatique, où la capacité à effectuer de nombreux calculs en parallèle peut considérablement accélérer l'entraînement des modèles.

B.7 AWS

Amazon Web Services (AWS) est une plateforme de services de cloud computing qui offre une multitude de services, dont EC2 (Elastic Compute Cloud), qui fournit une capacité de calcul évolutive dans le cloud. EC2 permet aux utilisateurs de lancer des instances de serveur virtuel pour diverses applications. L'un de ces usages pourrait être l'exécution de NGINX, un serveur web open-source, sur une instance EC2. NGINX est connu pour sa performance, sa stabilité, et sa faible consommation de ressources, et peut être utilisé comme un serveur web, un reverse proxy, ou un load balancer. Ensemble, AWS, EC2 et NGINX offrent une solution robuste et évolutive pour héberger et gérer des applications web.

Annexe C

Notions et concepts avancés

C.1 L'augmentation de données pour les données textuelles (Data Augmentation)

C.1.1 Définition

La Data Augmentation pour les textes est une technique utilisée en apprentissage automatique, spécialement en apprentissage profond (deep learning),elle consiste à augmenter la quantité de données d'entraînement sans collecter de nouvelles données. Cela se fait en créant de nouvelles versions des textes existants grâce à diverses transformations qui modifient la forme du texte mais préservent son sens. Cela peut aider à améliorer la performance des modèles de traitement du langage naturel (NLP) en leur permettant d'apprendre à partir d'un ensemble de données plus varié.

C.1.2 Les techniques de Data Augmentation

Parmi les techniques couramment utilisées pour la Data Augmentation de textes :

- **Synonym Replacement :** Cette technique remplace des mots dans le texte par leurs synonymes.
- **Random Insertion :** Cette technique ajoute des mots aléatoires dans le texte.
- **Random Swap :** Cette technique échange aléatoirement deux mots dans une phrase.
- **Random Deletion :** Cette technique supprime aléatoirement un mot dans une phrase.
- **Back Translation :** Cette technique traduit un texte dans une autre langue puis le re-traduit en langue originale.

C.1.3 Sa Relation avec le Transfer Learning

L'augmentation de données pour les données textuelles peut être utilisée en conjonction avec le Transfer Learning pour améliorer encore les performances. Par exemple, on peut appliquer des techniques d'augmentation de données comme la paraphrase, l'insertion de synonymes, le remplacement de mots ou la traduction de retour pour augmenter la taille et la diversité de notre ensemble de données, ce qui donne au modèle plus de données à partir desquelles apprendre pour la nouvelle tâche, améliorant ainsi sa capacité à généraliser à partir de ces nouvelles données.

C.1.4 Importance de l'augmentation de données dans l'apprentissage machine

L'augmentation de données est particulièrement importante pour les données textuelles en raison de la nature complexe et variée du langage. Elle permet de générer des variations linguistiques supplémentaires qui ne sont pas présentes dans l'ensemble de données d'origine, ce qui peut aider à améliorer la robustesse et la performance du modèle.

- **Prévention du surapprentissage :** Avec un ensemble de données d'entraînement limité, les modèles ont tendance à surapprendre et à mémoriser les données d'entraînement au lieu de généraliser à partir d'elles. La Data Augmentation peut aider à prévenir cela en introduisant de la variété dans l'ensemble de données d'entraînement.
- **Amélioration de la robustesse du modèle :** Les modèles de traitement du langage naturel (NLP) doivent gérer une grande variété de phraséologie, de contexte, de dialectes et d'autres complexités linguistiques. En utilisant l'augmentation de données, on peut créer un ensemble de données d'entraînement qui expose le modèle à un plus grand éventail de ces complexités, ce qui peut aider à améliorer sa robustesse et sa précision.
- **Gestion des déséquilibres des classes :** L'augmentation de données peut être particulièrement utile pour gérer les déséquilibres des classes dans les données textuelles, par exemple lorsqu'un certain type de sentiment ou de sujet est sous-représenté dans les données d'entraînement.

C.2 L'Overfitting

C.2.1 Qu'est-ce que l'Overfitting ?

L'overfitting, ou surapprentissage, se produit lorsqu'un modèle de machine learning apprend trop bien les données d'entraînement au point de ne pas pouvoir généraliser efficacement à de nouvelles données inconnues. En d'autres termes, le modèle est tellement complexe qu'il capture non seulement les tendances générales dans les données, mais aussi le bruit et les anomalies. Cela conduit souvent à une excellente performance sur les données d'entraînement, mais une mauvaise performance sur les données de test ou de validation.

C.2.2 Techniques pour éviter l'Overfitting

Parmi les techniques couramment utilisées pour éviter l'overfitting :

- **Collecter plus de données :** Si possible, collecter plus de données peut aider à améliorer la capacité du modèle à généraliser.
- **Utiliser la validation croisée (Cross-validation) :** Cette technique divise les données d'entraînement en plusieurs sous-ensembles et entraîne le modèle sur chaque sous-ensemble, ce qui aide à évaluer comment le modèle pourrait se comporter avec de nouvelles données.
- **La Régularisation :** La régularisation ajoute une pénalité à la fonction de coût du modèle pour réduire la complexité du modèle et éviter l'overfitting. Des exemples de techniques de régularisation incluent L1 et L2.
- **Le Dropout :** Dans les réseaux de neurones, le dropout consiste à désactiver aléatoirement certains neurones pendant l'entraînement, ce qui peut aider à prévenir l'overfitting.

- **L'Arrêt précoce (Early stopping) :** L'arrêt précoce (Early stopping) consiste à arrêter l'entraînement lorsque la performance sur l'ensemble de validation cesse de s'améliorer, même si la performance sur l'ensemble d'entraînement continue de s'améliorer.

C.2.3 Sa relation avec le Transfer Learning

Le surapprentissage et le Transfer Learning sont deux concepts liés dans le domaine de l'apprentissage machine et ils interagissent souvent dans la pratique.

Le Transfer Learning, comme expliqué précédemment, consiste à utiliser un modèle pré-entraîné sur une grande quantité de données et à l'adapter à une nouvelle tâche spécifique. Il est particulièrement utile lorsque les données pour la nouvelle tâche sont limitées, car le modèle a déjà appris des caractéristiques générales qui peuvent être transférées à la nouvelle tâche.

C'est là que le surapprentissage entre en jeu. Lorsque les modèles sont entraînés sur de petites quantités de données, ils sont plus susceptibles de surapprendre ces données - ils peuvent apprendre le "bruit" spécifique dans l'ensemble de données d'entraînement au lieu de la tendance générale. Lorsque ces modèles sont ensuite testés sur de nouvelles données, ils peuvent ne pas bien se généraliser.

C'est là que le Transfer Learning peut aider. En utilisant un modèle qui a été pré-entraîné sur un grand ensemble de données, on peut tirer parti de ce que le modèle a déjà appris sur ces données, ce qui peut aider à prévenir le surapprentissage sur le petit ensemble de données de notre nouvelle tâche. Le modèle a déjà appris à généraliser à partir du grand ensemble de données sur lequel il a été pré-entraîné, il est donc moins susceptible de surapprendre le petit ensemble de données de la nouvelle tâche.

C.2.4 Pourquoi l'évitement de l'Overfitting est-il important ?

L'évitement de l'overfitting est crucial pour obtenir des modèles de machine learning qui sont capables de généraliser à de nouvelles données. Un modèle qui est overfitting aura une performance médiocre lorsqu'il sera confronté à de nouvelles données, ce qui limite son utilité dans des applications réelles. En utilisant des techniques pour éviter l'overfitting, nous pouvons développer des modèles plus robustes et fiables.

C.3 La méthode CRISP-DM

C.3.1 Qu'est-ce que la méthode CRISP-DM ?

CRISP-DM, qui signifie Cross Industry Standard Process for Data Mining, est une méthodologie standard qui est largement utilisée pour structurer les projets de data mining et d'apprentissage automatique. Elle propose un processus itératif qui se décompose en six grandes étapes :

- **Compréhension du métier (Business Understanding) :** Cette étape implique de comprendre les objectifs du projet et les exigences du point de vue de l'entreprise. Elle nécessite une interaction étroite avec les parties prenantes pour définir clairement le problème à résoudre.
- **Compréhension des Données (Data Understanding) :** Cette étape consiste à recueillir les données nécessaires pour le projet, à les explorer et à comprendre leurs caractéristiques.

Elle peut comprendre des tâches comme la visualisation des données, l'identification des problèmes de qualité des données, etc.

- **Préparation des Données (Data Preparation) :** Il s'agit du processus de nettoyage et de transformation des données pour les rendre prêtes à être utilisées par les modèles d'apprentissage automatique. Cela peut impliquer des tâches comme le traitement des valeurs manquantes, la transformation des variables, la réduction de la dimensionnalité, etc
- **Modélisation (Modeling) :** Cette étape implique la sélection de modèles appropriés, leur entraînement sur les données préparées, et l'ajustement de leurs paramètres pour obtenir la meilleure performance possible.
- **Évaluation (Evaluation) :** Il s'agit d'évaluer la performance des modèles sur un ensemble de données de test ou de validation pour s'assurer qu'ils sont prêts à être déployés. Cela peut impliquer des mesures de performance comme l'accuracy, la précision, etc.
- **Déploiement (Deployment) :** Cette dernière étape implique la mise en œuvre des modèles dans un environnement de production où ils peuvent être utilisés pour faire des prédictions sur de nouvelles données.

C.3.2 Importance et utilité de la méthode

La méthode CRISP-DM est importante pour plusieurs raisons. D'abord, elle offre un cadre structuré et standardisé pour mener des projets de data mining. Cela aide à garantir que toutes les étapes clés du processus sont prises en compte et que rien n'est négligé.

En outre, le cadre CRISP-DM est largement reconnu et adopté dans l'industrie, ce qui signifie qu'il est largement compris et accepté. Cela peut faciliter la communication et la collaboration entre les équipes, en particulier dans les grands projets où de nombreuses parties prenantes peuvent être impliquées.

De plus, CRISP-DM est conçu pour être flexible et adaptable à divers contextes. Il peut être utilisé pour une large gamme de projets de data mining, quel que soit le secteur d'activité, le type de données ou le problème spécifique à résoudre.

Enfin, le fait de suivre une méthode éprouvée comme CRISP-DM peut aider à augmenter les chances de réussite d'un projet. Cela peut aider à éviter des erreurs courantes, à gérer efficacement les ressources et à atteindre les objectifs du projet de manière plus efficace et efficiente.

C.4 Gestion de projets

C.4.1 Définition et importance de la gestion de projets

La gestion de projets est une pratique qui implique la planification, l'organisation, la motivation et le contrôle des ressources pour atteindre des objectifs spécifiques dans un temps limité. Un projet est un effort temporaire conçu pour produire un produit, un service ou un résultat unique.

La gestion de projets est importante car elle assure qu'il y a une planification claire et définie pour atteindre les objectifs du projet. Elle sert à établir des attentes claires, à maintenir un haut degré d'organisation, à gérer efficacement les ressources et à minimiser ou éviter les risques potentiels. Une gestion de projets efficace peut également aider à garantir la qualité du produit final, améliorer l'efficacité du travail d'équipe, et contribuer à la réalisation des objectifs dans les délais et le budget prévus.

C.4.2 Principes fondamentaux de la gestion de projets

Cycle de vie d'un projet :

Le cycle de vie d'un projet décrit les phases distinctes par lesquelles passe un projet de sa conception à sa clôture. Les phases typiques comprennent l'initiation, la planification, l'exécution, le suivi et le contrôle, et la clôture. Chaque phase a des objectifs spécifiques et nécessite des ressources et un planning particuliers.

Les cinq groupes de processus :

- **Initiation :** Cette phase comprend la définition du projet et la détermination de sa faisabilité. Les objectifs du projet, les parties prenantes, les livrables attendus et les bénéfices attendus sont identifiés lors de cette phase.
- **Planification :** La phase de planification détaille les étapes, les tâches et les ressources nécessaires pour atteindre les objectifs du projet. Cela inclut la création d'un échéancier de projet, la détermination des coûts estimés, et la planification des ressources nécessaires.
- **Exécution :** Durant cette phase, les tâches et les activités définies dans le plan du projet sont mises en œuvre pour créer les livrables du projet.
- **Suivi et contrôle :** Cette phase implique le suivi du progrès du projet pour s'assurer qu'il reste sur la bonne voie. Elle comprend également l'identification et la gestion des problèmes et des risques qui peuvent survenir.
- **Clôture :** La phase de clôture marque la fin du projet. Cela implique la livraison des produits ou des résultats, la libération des ressources du projet, et l'évaluation du succès du projet par rapport aux objectifs initiaux.

Ces principes fondamentaux de la gestion de projets assurent une structure claire et détaillée pour le processus de gestion de projets, aidant les équipes à atteindre leurs objectifs de manière efficace et organisée.

C.4.3 Outils de Gestion de Projets

Les outils de gestion de projets sont essentiels pour gérer efficacement toutes les phases et les aspects d'un projet. Ils aident à la planification, à l'exécution, au suivi et au contrôle des projets. Voici quelques exemples d'outils de gestion de projets couramment utilisés :

- **Logiciels de gestion de projets :** Il existe de nombreux logiciels de gestion de projets sur le marché, comme Microsoft Project, Jira, Trello, Asana, Basecamp, et plus encore. Ces outils aident à la planification des tâches, à la gestion des ressources, à la communication au sein de l'équipe et à la collaboration.
- **Outils de planification :** Des outils comme les diagrammes de Gantt et PERT peuvent aider à visualiser le calendrier du projet, les dépendances entre les tâches, et l'état actuel du projet.
- **Outils de suivi :** Ces outils aident à suivre et à contrôler l'avancement du projet. Ils peuvent inclure des tableaux de bord, des indicateurs clés de performance (KPI) et des outils d'analyse.

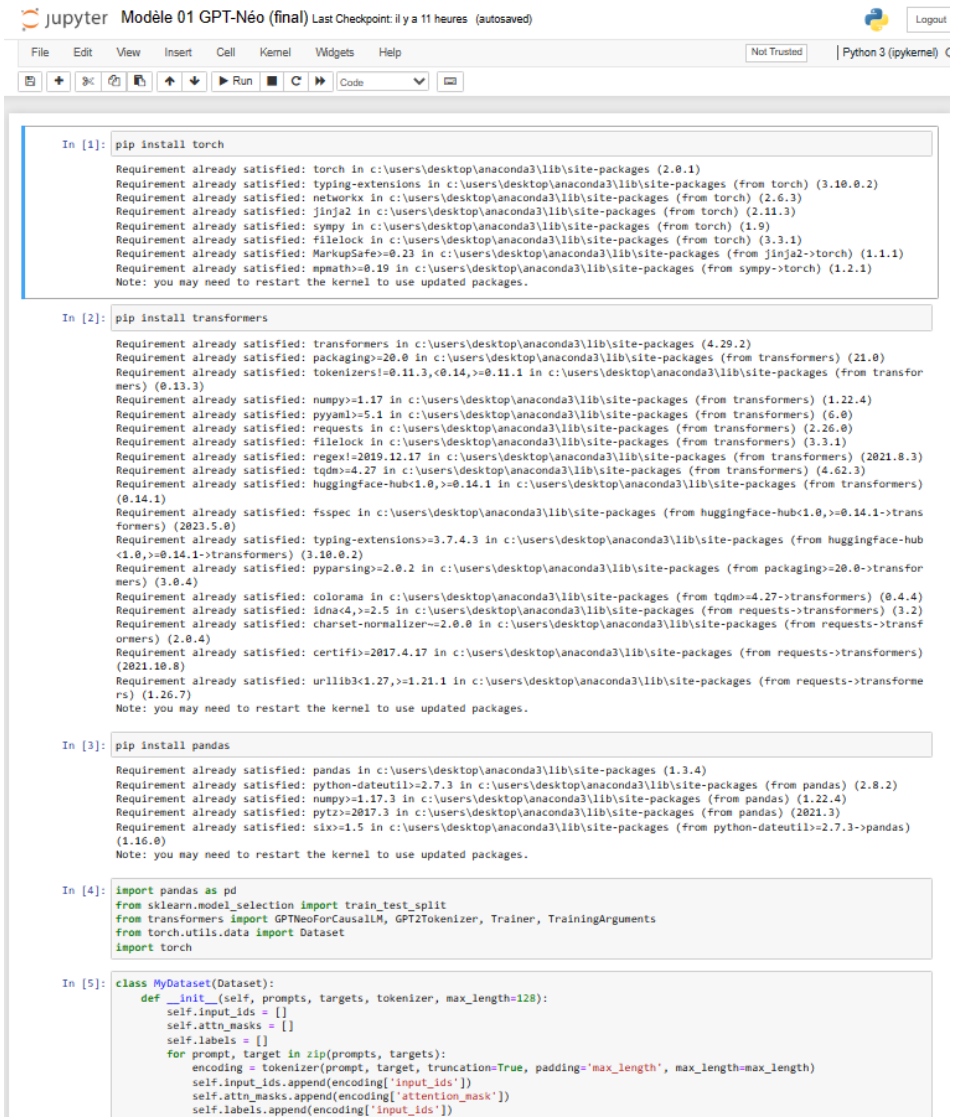
- **Outils de collaboration :** Les outils de collaboration, comme Slack, Microsoft Teams ou Google Workspace, peuvent améliorer la communication et la collaboration au sein de l'équipe de projet.

Annexe D

Codes Source

D.1 Implémentation

D.1.1 Modèle 01 : GPT-Neo



jupyter Modèle 01 GPT-Néo (final) Last Checkpoint: il y a 11 heures (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [5]:
```
class MyDataset(Dataset):
    def __init__(self, prompts, targets, tokenizer, max_length=128):
        self.input_ids = []
        self.attn_masks = []
        self.labels = []
        for prompt, target in zip(prompts, targets):
            encoding = tokenizer(prompt, target, truncation=True, padding='max_length', max_length=max_length)
            self.input_ids.append(encoding['input_ids'])
            self.attn_masks.append(encoding['attention_mask'])
            self.labels.append(encoding['input_ids'])

    def __getitem__(self, idx):
        return {
            'input_ids': torch.tensor(self.input_ids[idx]),
            'attention_mask': torch.tensor(self.attn_masks[idx]),
            'labels': torch.tensor(self.labels[idx])
        }

    def __len__(self):
        return len(self.input_ids)
```

In [6]:
```
# 3. Préparation du modèle
model_size = '1.3B'  # Taille du modèle GPT-Neo
model = GPTNeoForCausalLM.from_pretrained(f"EleutherAI/gpt-neo-{model_size}")
```

In [7]:
```
model.config.num_layers
```

Out[7]: 24

In [8]:
```
tokenizer = GPT2Tokenizer.from_pretrained(f"EleutherAI/gpt-neo-{model_size}")
```

In [9]:
```
# Freeze the first layers
for i, param in enumerate(model.parameters()):
    if i < 10:  # Adjust this number depending on how many layers you want to freeze
        param.requires_grad = False
```

In [10]:
```
# Set device to GPU if available
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
model.to(device)
```

Out[10]:
```
GPTNeoForCausalLM(
  (transformer): GPTNeoModel(
    (wte): Embedding(50257, 2048)
    (wpe): Embedding(2048, 2048)
    (drop): Dropout(p=0.0, inplace=False)
    (h): ModuleList(
      (0-23): 24 x GPTNeoBlock(
        (ln_1): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
        (attn): GPTNeoAttention(
          (attention): GPTNeoSelfAttention(
            (attn_dropout): Dropout(p=0.0, inplace=False)
            (resid_dropout): Dropout(p=0.0, inplace=False)
            (k_proj): Linear(in_features=2048, out_features=2048, bias=False)
            (v_proj): Linear(in_features=2048, out_features=2048, bias=False)
            (q_proj): Linear(in_features=2048, out_features=2048, bias=False)
            (out_proj): Linear(in_features=2048, out_features=2048, bias=True)
          )
        )
        (ln_2): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
        (mlp): GPTNeoMLP(
          (c_fc): Linear(in_features=2048, out_features=8192, bias=True)
          (c_proj): Linear(in_features=8192, out_features=2048, bias=True)
          (act): NewGELUActivation()
          (dropout): Dropout(p=0.0, inplace=False)
        )
      )
    )
```

jupyter Modèle 01 GPT-Néo (final) Last Checkpoint: il y a 11 heures (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [10]:
```
# Set device to GPU if available
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
model.to(device)
```

Out[10]:
```
GPTNeoForCausalLM(
  (transformer): GPTNeoModel(
    (wte): Embedding(50257, 2048)
    (wpe): Embedding(2048, 2048)
    (drop): Dropout(p=0.0, inplace=False)
    (h): ModuleList(
      (0-23): 24 x GPTNeoBlock(
        (ln_1): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
        (attn): GPTNeoAttention(
          (attention): GPTNeoSelfAttention(
            (attn_dropout): Dropout(p=0.0, inplace=False)
            (resid_dropout): Dropout(p=0.0, inplace=False)
            (k_proj): Linear(in_features=2048, out_features=2048, bias=False)
            (v_proj): Linear(in_features=2048, out_features=2048, bias=False)
            (q_proj): Linear(in_features=2048, out_features=2048, bias=False)
            (out_proj): Linear(in_features=2048, out_features=2048, bias=True)
          )
        )
        (ln_2): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
        (mlp): GPTNeoMLP(
          (c_fc): Linear(in_features=2048, out_features=8192, bias=True)
          (c_proj): Linear(in_features=8192, out_features=2048, bias=True)
          (act): NewGELUActivation()
          (dropout): Dropout(p=0.0, inplace=False)
        )
      )
    )
    (ln_f): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
  )
  (lm_head): Linear(in_features=2048, out_features=50257, bias=False)
)
```

In [11]:
```
# Load the data
df = pd.read_csv('BDD.csv')
prompts = df['Prompt'].tolist()
targets = df['Target'].tolist()
```

In [12]:
```
# Supprimer les doublons en se basant sur toutes les colonnes
df_unique = df.drop_duplicates()
```

In [13]:
```
# Écrire le DataFrame unique dans un nouveau fichier CSV
df_unique.to_csv("BDD1.csv", index=False)
```

In [14]:
```
# Supprimer le caractère " dans toutes les colonnes
df = df.replace('"', '', regex=True)

# Écrire la base de données modifiée dans un nouveau fichier CSV
df.to_csv("BDD2.csv", index=False)
```

In [15]:
```
df
```

Out[15]:

	Prompt	Target

In [16]:
```
# Split the data into training and validation sets
prompts_train, prompts_val, targets_train, targets_val = train_test_split(prompts, targets, test_size=0.1)
```

In [17]:
```
# Ajout du padding token
special_tokens_dict = {'pad_token': '<pad>'}
num_added_toks = tokenizer.add_special_tokens(special_tokens_dict)

print('We have added', num_added_toks, 'tokens')
model.resize_token_embeddings(len(tokenizer))  # Important: redimensionner le modèle de sortie
```

```
We have added 1 tokens
```

Out[17]:
```
Embedding(50258, 2048)
```

In [18]:
```
# Create the dataset
train_dataset = MyDataset(prompts_train, targets_train, tokenizer)
val_dataset = MyDataset(prompts_val, targets_val, tokenizer)
```

```
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
truncation strategy. So the returned list will always be empty even if some tokens have been removed.
Be aware, overflowing tokens are not returned for the setting you have chosen, i.e. sequence pairs with the 'longest_first'
```

In [19]:
```
# Define the training arguments
training_args = TrainingArguments(
    output_dir='./results',
    num_train_epochs=10,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=16,
    warmup_steps=500,
    weight_decay=0.01,
    logging_dir='./logs',
```

jupyter Modèle 01 GPT-Néo (final) Last Checkpoint: il y a 11 heures (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted | Python 3 (ipykernel)

Run Code

In [19]:
```
# Define the training arguments
training_args = TrainingArguments(
    output_dir='./results',
    num_train_epochs=10,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=16,
    warmup_steps=500,
    weight_decay=0.01,
    logging_dir='./logs',
    logging_steps=len(prompts_train)//10,  # Change this line
    evaluation_strategy = 'epoch'
)
```

In [20]:
```
# Initialize the Trainer
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=val_dataset,
)
```

In [21]:
```
# Train
trainer.train()
```

```
C:\Users\DESKTOP\anaconda3\lib\site-packages\transformers\optimization.py:407: FutureWarning: This implementation of AdamW is d
eprecated and will be removed in a future version. Use the PyTorch implementation torch.optim.AdamW instead, or set `no_depreca
tion_warning=True` to disable this warning
  warnings.warn(
```

[350/350 6:14:35, Epoch 10/10]

Epoch	Training Loss	Validation Loss
1	No log	1.580473
2	1.919100	1.331327
3	1.919100	1.259176
4	1.196700	1.239716
5	0.845400	1.277813
6	0.845400	1.368354
7	0.483300	1.423842
8	0.208700	1.487547
9	0.208700	1.513566
10	0.132400	1.542761

Out[21]:
```
TrainOutput(global_step=350, training_loss=0.7475839219774518, metrics={'train_runtime': 22540.9656, 'train_samples_per_secon
d': 0.242, 'train_steps_per_second': 0.016, 'total_flos': 5067390886871040.0, 'train_loss': 0.7475839219774518, 'epoch': 10.0})
```

In [50]:
```
eval_results = trainer.evaluate()
print(eval_results)
```

[4/4 02:18]

```
{'eval_loss': 1.5427607297897339, 'eval_runtime': 75.3414, 'eval_samples_per_second': 0.81, 'eval_steps_per_second': 0.053, 'ep
och': 10.0}
```

In [51]:
```
import math

eval_results = trainer.evaluate()
print(f"Perplexity: {math.exp(eval_results['eval_loss']):.2f}")
```

```
Perplexity: 4.68
```

In [22]:
```
import matplotlib.pyplot as plt
```

jupyter Modèle 01 GPT-Néo (final) Last Checkpoint: il y a 11 heures (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted | Python 3 (ipykernel)

Run Code

4	1.196700	1.239716
5	0.845400	1.277813
6	0.845400	1.368354
7	0.483300	1.423842
8	0.208700	1.487547
9	0.208700	1.513566
10	0.132400	1.542761

Out[21]:
```
TrainOutput(global_step=350, training_loss=0.7475839219774518, metrics={'train_runtime': 22540.9656, 'train_samples_per_secon
d': 0.242, 'train_steps_per_second': 0.016, 'total_flos': 5067390886871040.0, 'train_loss': 0.7475839219774518, 'epoch': 10.0})
```

In [50]:
```
eval_results = trainer.evaluate()
print(eval_results)
```

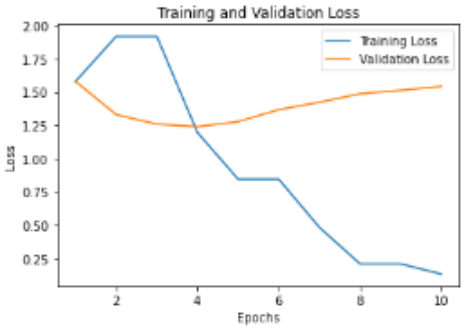
[4/4 02:18]

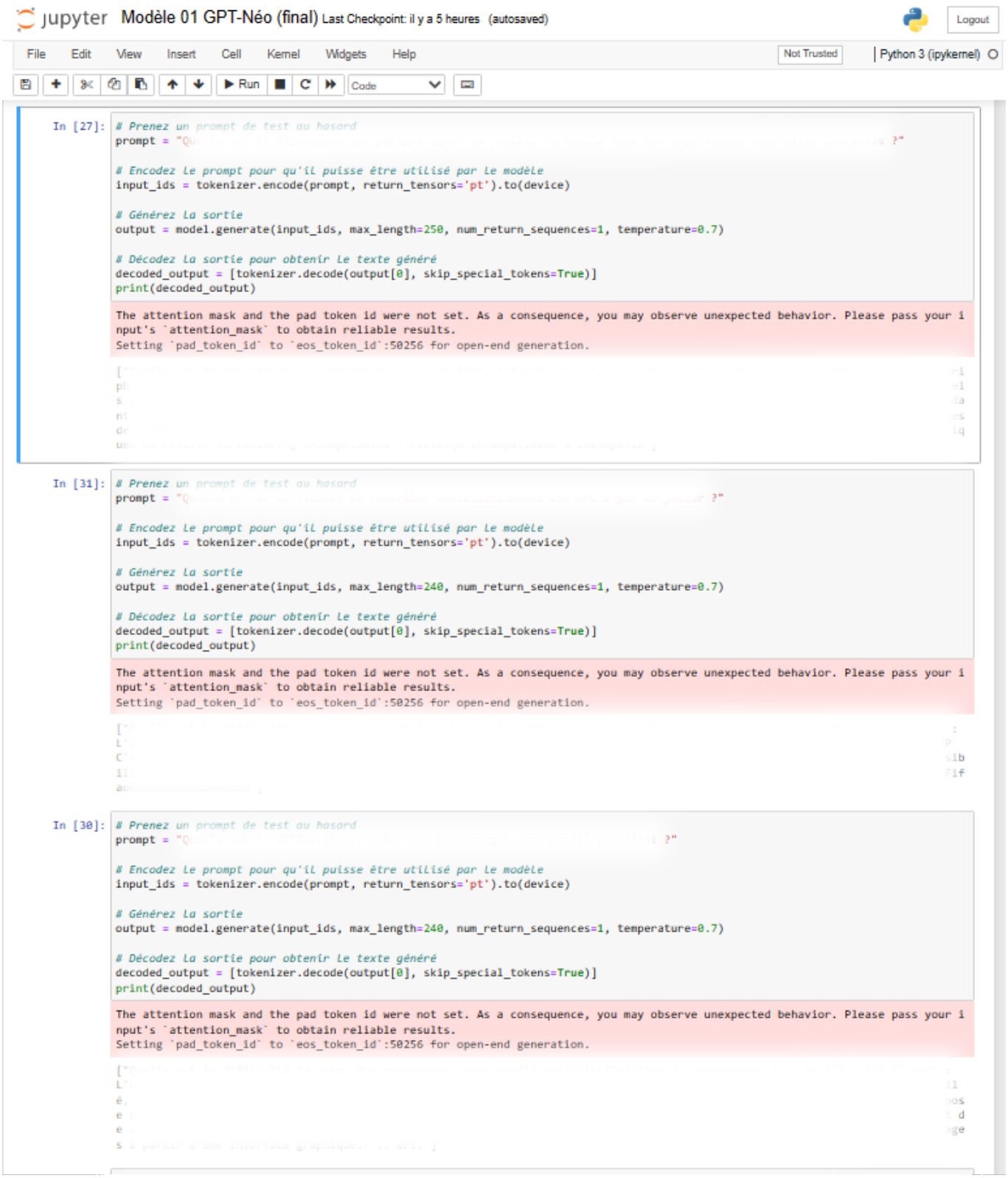
```
{'eval_loss': 1.5427607297897339, 'eval_runtime': 75.3414, 'eval_samples_per_second': 0.81, 'eval_steps_per_second': 0.053, 'ep
och': 10.0}
```

In [51]:
```
import math

eval_results = trainer.evaluate()
print(f"Perplexity: {math.exp(eval_results['eval_loss']):.2f}")
```

```
Perplexity: 4.68
```

In [22]:
```
import matplotlib.pyplot as plt

# Training and validation loss values
train_losses = [1.580473, 1.919100, 1.919100, 1.196700, 0.845400, 0.845400, 0.483300, 0.208700, 0.208700, 0.132400]
val_losses = [1.580473, 1.331327, 1.259176, 1.239716, 1.277813, 1.368354, 1.423842, 1.487547, 1.513566, 1.542761]

# Create x-axis values (epochs)
epochs = range(1, len(train_losses) + 1)

# Plot the training and validation losses
plt.plot(epochs, train_losses, label='Training Loss')
plt.plot(epochs, val_losses, label='Validation Loss')

# Add labels and title
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.title('Training and Validation Loss')

# Add legend
plt.legend()

# Show the plot
plt.show()
```




jupyter Modèle 01 GPT-Néo (final) Last Checkpoint: il y a 5 heures (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [27]:
# Prenez un prompt de test au hasard
prompt = "Qu
?"
# Encodez le prompt pour qu'il puisse être utilisé par le modèle
input_ids = tokenizer.encode(prompt, return_tensors='pt').to(device)
# Générez la sortie
output = model.generate(input_ids, max_length=250, num_return_sequences=1, temperature=0.7)
# Décodez la sortie pour obtenir le texte généré
decoded_output = [tokenizer.decode(output[0], skip_special_tokens=True)]
print(decoded_output)
The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your i
nput's `attention_mask` to obtain reliable results.
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
In [31]:
# Prenez un prompt de test au hasard
prompt = "Qu
?"
# Encodez le prompt pour qu'il puisse être utilisé par le modèle
input_ids = tokenizer.encode(prompt, return_tensors='pt').to(device)
# Générez la sortie
output = model.generate(input_ids, max_length=240, num_return_sequences=1, temperature=0.7)
# Décodez la sortie pour obtenir le texte généré
decoded_output = [tokenizer.decode(output[0], skip_special_tokens=True)]
print(decoded_output)
The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your i
nput's `attention_mask` to obtain reliable results.
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
In [30]:
# Prenez un prompt de test au hasard
prompt = "Qu
?"
# Encodez le prompt pour qu'il puisse être utilisé par le modèle
input_ids = tokenizer.encode(prompt, return_tensors='pt').to(device)
# Générez la sortie
output = model.generate(input_ids, max_length=240, num_return_sequences=1, temperature=0.7)
# Décodez la sortie pour obtenir le texte généré
decoded_output = [tokenizer.decode(output[0], skip_special_tokens=True)]
print(decoded_output)
The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your i
nput's `attention_mask` to obtain reliable results.
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

In [101]:
```python
#Analyse de la similarité de texte :

from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity

# Charger le modèle BERT pré-entraîné
model = SentenceTransformer('bert-base-uncased')

# Textes à comparer
text_generated = "D
text_reel = "D

# Obtenir les vecteurs d'embedding pour chaque texte
embeddings = model.encode([text_generated, text_reel])

# Calculer la similarité cosinus entre les paires de vecteurs
similarity_matrix = cosine_similarity(embeddings)

# Afficher la matrice de similarité
print(similarity_matrix)
```

```
No sentence-transformers model found with name C:\Users\DESKTOP/.cache\torch\sentence_transformers\bert-base-uncased. Creating
a new one with MEAN pooling.
Some weights of the model checkpoint at C:\Users\DESKTOP/.cache\torch\sentence_transformers\bert-base-uncased were not used whe
n initializing BertModel: ['cls.predictions.transform.dense.bias', 'cls.seq_relationship.weight', 'cls.predictions.transform.de
nse.weight', 'cls.predictions.transform.LayerNorm.bias', 'cls.predictions.bias', 'cls.seq_relationship.bias', 'cls.predictions.
decoder.weight', 'cls.predictions.transform.LayerNorm.weight']
- This IS expected if you are initializing BertModel from the checkpoint of a model trained on another task or with another arc
hitecture (e.g. initializing a BertForSequenceClassification model from a BertForPreTraining model).
- This IS NOT expected if you are initializing BertModel from the checkpoint of a model that you expect to be exactly identical
(initializing a BertForSequenceClassification model from a BertForSequenceClassification model).
```

```
[[1.0000002 0.960011 ]
 [0.960011  1.       ]]
```

In [102]:
```python
#Évaluation de la diversité des prédictions :

import textdistance

predictions = [text1, text2, text3]

diversity_scores = []
for i in range(len(predictions)):
    for j in range(i + 1, len(predictions)):
        similarity = 1 - textdistance.levenshtein.normalized_similarity(predictions[i], predictions[j])
        diversity_scores.append(similarity)

diversity = sum(diversity_scores) / len(diversity_scores)
print("Diversity score:", diversity)
```

```
Diversity score: 0.707459009429955
```

D.1.2 Modèle 02 : BART

In [1]:
```
pip install torch
```

```
Requirement already satisfied: torch in c:\users\desktop\anaconda3\lib\site-packages (2.0.1)
Requirement already satisfied: filelock in c:\users\desktop\anaconda3\lib\site-packages (from torch) (3.3.1)
Requirement already satisfied: sympy in c:\users\desktop\anaconda3\lib\site-packages (from torch) (1.9)
Requirement already satisfied: jinja2 in c:\users\desktop\anaconda3\lib\site-packages (from torch) (2.11.3)
Requirement already satisfied: networkx in c:\users\desktop\anaconda3\lib\site-packages (from torch) (2.6.3)
Requirement already satisfied: typing-extensions in c:\users\desktop\anaconda3\lib\site-packages (from torch) (3.10.0.2)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\desktop\anaconda3\lib\site-packages (from jinja2->torch) (1.1.1)
Requirement already satisfied: mpmath>=0.19 in c:\users\desktop\anaconda3\lib\site-packages (from sympy->torch) (1.2.1)
Note: you may need to restart the kernel to use updated packages.
```

```
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
```

In [2]:
```
pip install transformers
```

```
Requirement already satisfied: transformers in c:\users\desktop\anaconda3\lib\site-packages (4.29.2)
Requirement already satisfied: numpy>=1.17 in c:\users\desktop\anaconda3\lib\site-packages (from transformers) (1.22.4)
Requirement already satisfied: tokenizers!=0.11.3,<0.14,>=0.11.1 in c:\users\desktop\anaconda3\lib\site-packages (from transfor
mers) (0.13.3)
Requirement already satisfied: requests in c:\users\desktop\anaconda3\lib\site-packages (from transformers) (2.26.0)
Requirement already satisfied: pyyaml>=5.1 in c:\users\desktop\anaconda3\lib\site-packages (from transformers) (6.0)
Requirement already satisfied: regex!=2019.12.17 in c:\users\desktop\anaconda3\lib\site-packages (from transformers) (2021.8.3)
Requirement already satisfied: filelock in c:\users\desktop\anaconda3\lib\site-packages (from transformers) (3.3.1)
Requirement already satisfied: packaging>=20.0 in c:\users\desktop\anaconda3\lib\site-packages (from transformers) (21.0)
Requirement already satisfied: tqdm>=4.27 in c:\users\desktop\anaconda3\lib\site-packages (from transformers) (4.62.3)
Requirement already satisfied: huggingface-hub<1.0,>=0.14.1 in c:\users\desktop\anaconda3\lib\site-packages (from transformers)
(0.14.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in c:\users\desktop\anaconda3\lib\site-packages (from huggingface-hub
<1.0,>=0.14.1->transformers) (3.10.0.2)
Requirement already satisfied: fsspec in c:\users\desktop\anaconda3\lib\site-packages (from huggingface-hub<1.0,>=0.14.1->trans
formers) (2023.5.0)
Requirement already satisfied: pyparsing>=2.0.2 in c:\users\desktop\anaconda3\lib\site-packages (from packaging>=20.0->transfor
mers) (3.0.4)
Requirement already satisfied: colorama in c:\users\desktop\anaconda3\lib\site-packages (from tqdm>=4.27->transformers) (0.4.4)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\desktop\anaconda3\lib\site-packages (from requests->transf
ormers) (2.0.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\desktop\anaconda3\lib\site-packages (from requests->transforme
rs) (1.26.7)
Requirement already satisfied: idna<4,>=2.5 in c:\users\desktop\anaconda3\lib\site-packages (from requests->transformers) (3.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\desktop\anaconda3\lib\site-packages (from requests->transformers)
(2021.10.8)
Note: you may need to restart the kernel to use updated packages.
```

```
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
```

In [3]:
```
pip install pandas
```

jupyter Modèle 02 BART (final) Last Checkpoint: il y a 11 heures (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted | Python 3 (ipykernel)

Run Code

In [3]:
```
pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\desktop\anaconda3\lib\site-packages (1.3.4)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\desktop\anaconda3\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in c:\users\desktop\anaconda3\lib\site-packages (from pandas) (2021.3)
Requirement already satisfied: numpy>=1.17.3 in c:\users\desktop\anaconda3\lib\site-packages (from pandas) (1.22.4)
Requirement already satisfied: six>=1.5 in c:\users\desktop\anaconda3\lib\site-packages (from python-dateutil>=2.7.3->pandas)
(1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

```
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -ordcloud (c:\users\desktop\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib (c:\users\desktop\anaconda3\lib\site-packages)
```

In [4]:
```
import torch
from sklearn.model_selection import train_test_split
import pandas as pd
from transformers import BartTokenizer, BartForConditionalGeneration, Trainer, TrainingArguments
```

In [5]:
```
# Vérifiez si CUDA est disponible et réglez l'appareil sur GPU si c'est le cas
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
```

In [6]:
```
# 2. Load the model and tokenizer
model = BartForConditionalGeneration.from_pretrained('facebook/bart-large')
tokenizer = BartTokenizer.from_pretrained('facebook/bart-large')
```

In [7]:
```
# Chargez votre base de données
df = pd.read_csv('BDD2.csv')
```

In [8]:
```
# Séparez les données en données d'entraînement et de validation
train_df, val_df = train_test_split(df, test_size=0.2)
```

In [9]:
```
# Préparez les données d'entraînement
train_encodings = tokenizer(train_df['Prompt'].tolist(), truncation=True, padding=True)
train_labels = tokenizer(train_df['Target'].tolist(), truncation=True, padding=True)

# Préparez les données de validation
val_encodings = tokenizer(val_df['Prompt'].tolist(), truncation=True, padding=True)
val_labels = tokenizer(val_df['Target'].tolist(), truncation=True, padding=True)
```

In [10]:
```
class UncertaintyDataset(torch.utils.data.Dataset):
    def __init__(self, encodings, labels):
        self.encodings = encodings
        self.labels = labels

    def __getitem__(self, idx):
        item = {key: torch.tensor(val[idx]) for key, val in self.encodings.items()}
        item['labels'] = torch.tensor(self.labels['input_ids'][idx])
        return item

    def __len__(self):
        return len(self.encodings.input_ids)
```

In [11]:
```
# Créez les ensembles de données d'entraînement et de validation
train_dataset = UncertaintyDataset(train_encodings, train_labels)
val_dataset = UncertaintyDataset(val_encodings, val_labels)
```

jupyter Modèle 02 BART (final) Last Checkpoint: il y a 11 heures (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [11]:
```
# Créez les ensembles de données d'entraînement et de validation
train_dataset = UncertaintyDataset(train_encodings, train_labels)
val_dataset = UncertaintyDataset(val_encodings, val_labels)
```

In [12]:
```
training_args = TrainingArguments(
    output_dir='./results',
    num_train_epochs=10,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=64,
    warmup_steps=500,
    weight_decay=0.01,
    logging_dir='./logs',
    logging_steps=10,
    evaluation_strategy="epoch", # ajoutez cette ligne
)
```

In [13]:
```
# Créez le formateur et démarrez l'entraînement
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=val_dataset
)
```

In [14]:
```
trainer.train()
```

```
C:\Users\DESKTOP\anaconda3\lib\site-packages\transformers\optimization.py:407: FutureWarning: This implementation of AdamW is d
eprecated and will be removed in a future version. Use the PyTorch implementation torch.optim.AdamW instead, or set `no_depreca
tion_warning=True` to disable this warning
  warnings.warn(
```

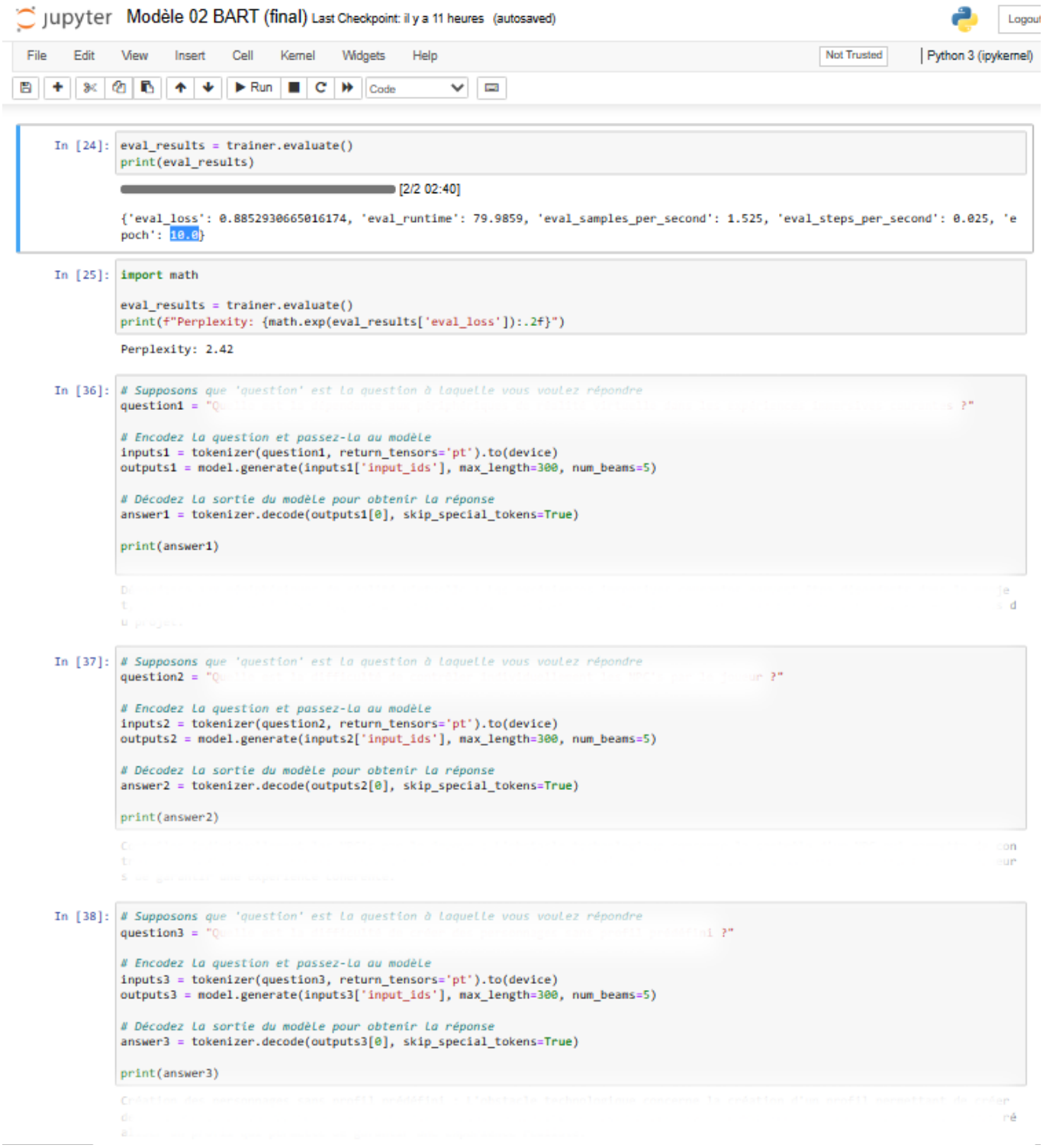
[310/310 3:29:26, Epoch 10/10]

Epoch	Training Loss	Validation Loss
1	11.482100	10.091373
2	8.296100	7.356726
3	6.574100	5.017141
4	4.877200	3.932638
5	4.141600	3.329330
6	3.375200	2.597049
7	2.492400	1.720202
8	1.603800	1.090688
9	1.028500	0.925022
10	0.963300	0.885293

Out[14]:
```
TrainOutput(global_step=310, training_loss=4.906545934369487, metrics={'train_runtime': 12606.0985, 'train_samples_per_second':
0.385, 'train_steps_per_second': 0.025, 'total_flos': 913506544435200.0, 'train_loss': 4.906545934369487, 'epoch': 10.0})
```

In [22]:
```
pip install --upgrade matplotlib
```

```
Requirement already satisfied: matplotlib in c:\users\desktop\anaconda3\lib\site-packages (3.7.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (9.5.0)
Requirement already satisfied: packaging>=20.0 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (21.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (2.8.
2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (1.0.7)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (3.0.4)
Requirement already satisfied: cycler>=0.10 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (1.3.1)
Requirement already satisfied: numpy>=1.20 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib) (1.22.4)
Requirement already satisfied: importlib-resources>=3.2.0 in c:\users\desktop\anaconda3\lib\site-packages (from matplotlib)
```


jupyter Modèle 02 BART (final) Last Checkpoint: il y a 11 heures (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [24]:
eval_results = trainer.evaluate()
print(eval_results)
[2/2 02:40]
{'eval_loss': 0.8852930665016174, 'eval_runtime': 79.9859, 'eval_samples_per_second': 1.525, 'eval_steps_per_second': 0.025, 'epoch': 10.0}
In [25]:
import math
eval_results = trainer.evaluate()
print(f"Perplexity: {math.exp(eval_results['eval_loss']):.2f}")
Perplexity: 2.42
In [36]:
# Supposons que 'question' est la question à laquelle vous voulez répondre
question1 = "
?"
# Encodez la question et passez-la au modèle
inputs1 = tokenizer(question1, return_tensors='pt').to(device)
outputs1 = model.generate(inputs1['input_ids'], max_length=300, num_beams=5)
# Décodez la sortie du modèle pour obtenir la réponse
answer1 = tokenizer.decode(outputs1[0], skip_special_tokens=True)
print(answer1)
In [37]:
# Supposons que 'question' est la question à laquelle vous voulez répondre
question2 = "
?"
# Encodez la question et passez-la au modèle
inputs2 = tokenizer(question2, return_tensors='pt').to(device)
outputs2 = model.generate(inputs2['input_ids'], max_length=300, num_beams=5)
# Décodez la sortie du modèle pour obtenir la réponse
answer2 = tokenizer.decode(outputs2[0], skip_special_tokens=True)
print(answer2)
In [38]:
# Supposons que 'question' est la question à laquelle vous voulez répondre
question3 = "
?"
# Encodez la question et passez-la au modèle
inputs3 = tokenizer(question3, return_tensors='pt').to(device)
outputs3 = model.generate(inputs3['input_ids'], max_length=300, num_beams=5)
# Décodez la sortie du modèle pour obtenir la réponse
answer3 = tokenizer.decode(outputs3[0], skip_special_tokens=True)
print(answer3)

In [40]:
```python
#Analyse de la similarité de texte :

from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity

# Charger le modèle BERT pré-entraîné
model = SentenceTransformer('bert-base-uncased')

# Textes à comparer
text_generated =
text_reel =

# Obtenir les vecteurs d'embedding pour chaque texte
embeddings = model.encode([text_generated, text_reel])

# Calculer la similarité cosinus entre les paires de vecteurs
similarity_matrix = cosine_similarity(embeddings)

# Afficher la matrice de similarité
print(similarity_matrix)
```

```
No sentence-transformers model found with name C:\Users\DESKTOP/.cache\torch\sentence_transformers\bert-base-uncased. Creating
a new one with MEAN pooling.
Some weights of the model checkpoint at C:\Users\DESKTOP/.cache\torch\sentence_transformers\bert-base-uncased were not used whe
n initializing BertModel: ['cls.predictions.transform.LayerNorm.weight', 'cls.seq_relationship.bias', 'cls.predictions.transfor
m.dense.bias', 'cls.predictions.decoder.weight', 'cls.predictions.bias', 'cls.predictions.transform.dense.weight', 'cls.seq_rel
ationship.weight', 'cls.predictions.transform.LayerNorm.bias']
- This IS expected if you are initializing BertModel from the checkpoint of a model trained on another task or with another arc
hitecture (e.g. initializing a BertForSequenceClassification model from a BertForPreTraining model).
- This IS NOT expected if you are initializing BertModel from the checkpoint of a model that you expect to be exactly identical
(initializing a BertForSequenceClassification model from a BertForSequenceClassification model).
```

```
[[1.0000002 0.9557236]
 [0.9557236 1.       ]]
```

In [41]:
```python
#Évaluation de la diversité des prédictions :

import textdistance

predictions = [text1, text2, text3]

diversity_scores = []
for i in range(len(predictions)):
    for j in range(i + 1, len(predictions)):
        similarity = 1 - textdistance.levenshtein.normalized_similarity(predictions[i], predictions[j])
        diversity_scores.append(similarity)

diversity = sum(diversity_scores) / len(diversity_scores)
print("Diversity score:", diversity)
```

```
Diversity score: 0.6775597269624574
```

D.2 Développement de l'API

D.2.1 Modèle 01 : GPT-Neo

In [2]:
```
from flask import Flask, request, jsonify
from transformers import GPTNeoForCausalLM, GPT2Tokenizer
import torch
app = Flask(__name__)

# Spécifiez la taille du modèle à utiliser
model_size = '1.3B'

# Chargez le modèle et le tokenizer
model = GPTNeoForCausalLM.from_pretrained(f"EleutherAI/gpt-neo-{model_size}")
tokenizer = GPT2Tokenizer.from_pretrained(f"EleutherAI/gpt-neo-{model_size}")

# Vérifiez si un GPU est disponible et, dans l'affirmative, déplacez le modèle vers le GPU
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
model.to(device)

@app.route('/predict', methods=['POST'])
def predict():
    data = request.get_json(force=True)
    prompt = data['prompt']

    # Encodez le prompt pour qu'il puisse être utilisé par le modèle
    input_ids = tokenizer.encode(prompt, return_tensors='pt').to(device)

    # Générez la sortie
    output = model.generate(input_ids, max_length=250, num_return_sequences=1, temperature=0.7)

    # Décodage de la sortie
    generated_text = tokenizer.decode(output[0], skip_special_tokens=True)

    # Retournez la sortie générée sous forme de réponse JSON
    return jsonify({'generated_text': generated_text})

# Lancez l'application
if __name__ == '__main__':
    app.run(port=5000, debug=True)
```

```
 * Serving Flask app "__main__" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with watchdog (windowsapi)
An exception has occurred, use %tb to see the full traceback.

SystemExit: 1


C:\Users\DESKTOP\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3452: UserWarning: To exit: use 'exit', 'quit', o
r Ctrl-D.
  warn("To exit: use 'exit', 'quit', or Ctrl-D.", stacklevel=1)
```

In [5]:
```
import threading
from werkzeug.serving import run_simple
def run_server():
    app.run(port=5000, use_reloader=False)

server_thread = threading.Thread(target=run_server)
server_thread.start()
```

```
 * Serving Flask app "__main__" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

D.2.2 Modèle 02 : BART

In [15]:
```
from flask import Flask, request, jsonify
from transformers import BartTokenizer, BartForConditionalGeneration
import torch

# Initialisation du modèle et du tokenizer
model = BartForConditionalGeneration.from_pretrained('facebook/bart-large')
tokenizer = BartTokenizer.from_pretrained('facebook/bart-large')

app = Flask(__name__)

@app.route('/generate', methods=['POST'])
def generate():
    data = request.get_json()
    prompt = data.get('prompt', '')

    inputs = tokenizer([prompt], max_length=1024, return_tensors='pt')

    # Générez la sortie
    output = model.generate(inputs['input_ids'], max_length=250, num_return_sequences=1, temperature=0.7)
    generated_text = tokenizer.decode(output[0], skip_special_tokens=True)

    return jsonify({'generated_text': generated_text})
if __name__ == '__main__':
    app.run(port=5000, debug=True)
```

```
 * Serving Flask app "__main__" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with watchdog (windowsapi)
An exception has occurred, use %tb to see the full traceback.

SystemExit: 1


C:\Users\DESKTOP\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3452: UserWarning: To exit: use 'exit', 'quit', o
r Ctrl-D.
  warn("To exit: use 'exit', 'quit', or Ctrl-D.", stacklevel=1)
```

In [16]:
```
import threading
from werkzeug.serving import run_simple

def run_server():
    app.run(port=5000, use_reloader=False)

server_thread = threading.Thread(target=run_server)
server_thread.start()
```

```
 * Serving Flask app "__main__" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [17/Jun/2023 11:41:36] "GET / HTTP/1.1" 404 -
127.0.0.1 - - [17/Jun/2023 11:41:36] "GET /favicon.ico HTTP/1.1" 404 -
```