



FACULTÉ DES SCIENCES INFORMATIQUES
Réseaux et Infrastructures

**Conception et développement d'un chatbot basé sur
un LLM comme support de vulgarisation au système
juridique Congolais**

*Travail de fin d'études présenté et défendu en vue de l'obtention
du grade de licencié en sciences informatiques.*

Présenté par : TSHABU NGANDU Bernard

Juillet 2024



FACULTÉ DES SCIENCES INFORMATIQUES
Réseaux et Infrastructures

**Conception et développement d'un chatbot basé sur
un LLM comme support de vulgarisation au système
juridique Congolais**

*Travail de fin d'études présenté et défendu en vue de l'obtention
du grade de licencié en sciences informatiques.*

*Présenté par : TSHABU NGANDU Bernard
Dirigé par : Prof. BAGULA Antoine PhD.
Co-dirigé par : Ass. MBALE Landry*

Juillet 2024

Dédié à la mémoire de Ngalula Tuadila Catherine.
1960 – 2018

RÉSUMÉ

Dans un environnement où l'accès à l'information juridique est souvent entravé, et où la complexité du vocabulaire juridique est un obstacle majeur à la compréhension du système juridique de la République démocratique du Congo, ce projet vise à créer et à déployer un chatbot innovant, basé sur un Large Language Model (LLM). L'objectif de cette initiative est de faciliter l'accès à la connaissance juridique. En utilisant des techniques d'intelligence artificielle, ce chatbot vise à réinterpréter et à simplifier le langage juridique, le rendant ainsi intelligible pour tous. Ce travail se concentre sur l'affinement des réponses générées par les LLM en utilisant une architecture de type Retrieval-Augmented Generation (RAG) et une application web (Juro) servant d'interface utilisateur.

Notre contribution majeure comprend la création d'une architecture de scraper web pour la collecte systématique de documents juridiques, formant ainsi le premier ensemble de données structuré de documents juridiques en République Démocratique du Congo. Nous avons également développé une architecture RAG flexible permettant l'interchangeabilité des modèles d'intégration et de langage, offrant ainsi la possibilité de tester et d'intégrer différents modèles afin d'optimiser les performances du chatbot. De plus, nous avons mis en place un mécanisme de citation des sources dans les réponses générées, garantissant la fiabilité et la traçabilité de l'information. Enfin, la conception de l'architecture est adaptable à d'autres domaines que le droit, ce qui démontre sa polyvalence et son potentiel d'application dans divers contextes nécessitant la vulgarisation et l'accessibilité d'informations complexes.

Les résultats obtenus montrent que les réponses générées par notre modèle sont globalement pertinentes et précises dans le contexte juridique congolais par rapport aux modèles d'entreprise, bien que certaines réponses nécessitent une compréhension plus approfondie des nuances juridiques et contextuelles. Les évaluations qualitatives via Google Form ont indiqué que les utilisateurs ont trouvé les réponses utiles, malgré quelques incohérences mineures. Le temps nécessaire pour recevoir le premier jeton après l'envoi de la requête API a révélé une latence plus élevée que les modèles d'entreprise, principalement en raison de notre infrastructure de serveur moins optimisée.

Mots clés : Chatbot, Large Language Model (LLM), Système Juridique Congolais, Vulgarisation, Accessibilité à l'Information, Intelligence Artificielle, Retrieval-augmented Generation

ABSTRACT

In an environment where access to legal information is frequently hampered, and where the complexity of legal vocabulary is a major obstacle to understanding the legal system of the Democratic Republic of Congo, this project aims to create and deploy an innovative chatbot, based on a Large Language Model (LLM). The aim of this initiative is to facilitate access to legal knowledge. Using artificial intelligence techniques, this chatbot aims to reinterpret and simplify legal language, making it intelligible to all. This work focuses on fine-tuning the responses generated by LLMs using a Retrieval-Augmented Generation (RAG) architecture and a web application (Juro) serving as the user interface.

Our major contribution includes the creation of a web scraper architecture for the systematic collection of legal documents, thus forming the first structured dataset of legal documents in the Democratic Republic of Congo. We also developed a flexible RAG architecture enabling interchangeability of embedding and language models, thus offering the possibility of testing and integrating various models to optimise chatbot performance. In addition, we have implemented a mechanism for quoting sources in the responses generated, guaranteeing the reliability and traceability of the information. Finally, the architecture design is adaptable to other fields outside law, demonstrating its versatility and potential for application in various contexts requiring the popularisation and accessibility of complex information.

The results obtained show that the answers generated by our model are globally relevant and accurate in the Congolese legal context than enterprise models, although some answers require a deeper understanding of legal and contextual nuances. Qualitative evaluations via Google Form indicated that users found the answers useful, despite some minor inconsistencies. The time taken to receive the first token after sending the API request revealed a higher latency compared to enterprise models, mainly due to our less optimised server infrastructure.

Key words : Chatbot, Large Language Model (LLM), Congolese Legal System, Popularization, Information Accessibility, Artificial Intelligence, Retrieval-augmented generation

REMERCIEMENTS

Arrivé à la fin de ce travail, je me trouve dans l'obligation et l'honneur de présenter un témoignage de gratitude envers toutes les personnes qui ont, de près ou de loin, contribué à l'aboutissement de ce mémoire. Ce parcours a été jalonné de défis et de découvertes, et je ne serais pas parvenu à ce stade sans le soutien et les encouragements constants de ceux qui m'ont entouré.

En premier lieu, je remercie chaleureusement le **Professeur Antoine Bagula, PhD**, directeur de ce travail, pour ses précieux conseils, ses orientations et sa disponibilité. Son encadrement a été d'une aide inestimable, nous accordant une liberté de recherche nécessaire à l'exploration de ce domaine tout en veillant à ce que nous restions sur la bonne voie. Son expertise et sa bienveillance ont été des piliers tout au long de ce parcours académique.

Je souhaite également exprimer ma reconnaissance à l'**Assistant Landry Mbale**, co-directeur de ce travail, pour les séances de brainstorming inspirantes, sa disponibilité et ses retours constants à chaque étape du projet. Ses commentaires pertinents et son soutien continu ont grandement contribué à la qualité et à la rigueur de ce mémoire.

À ma famille, je dois une reconnaissance éternelle pour leur soutien émotionnel, spirituel et financier. Mon père, **John Kalolo Ngandu**, sans qui mes études n'auraient pas été possibles. Ma sœur, **Marguerite Tshiambi Kalolo**, et mon frère, **Junior Tshilenge Ngandu**, ont été des sources constantes d'encouragement et d'inspiration. Leur présence et leur amour ont été un soutien indéfectible.

Mes remerciements vont aussi à **Ketsia Kamotela Heri**, pour son soutien tant émotionnel que professionnel. Son expertise en droit congolais, ses orientations techniques et l'inspiration qu'elle a apportée au sujet de ce mémoire ont été déterminantes. Son soutien a été d'une grande aide tout au long de ce travail.

Enfin, je tiens à remercier mes amis et frères, **Méschac Irung**, **Arthur Kaza**, **Martin Wakenge**, **Joyce Banza**, **Amaury Cansa**, et **Rusade Kakanga**, pour leur soutien indéfectible. Leur amitié, leurs encouragements et leur confiance en moi ont été une source de motivation et de force tout au long de cette aventure.

À tous ceux qui ont contribué, de près ou de loin, à la réalisation de ce travail, je vous exprime ma plus profonde gratitude. Merci.

TABLE DES MATIÈRES

Dédicace	iii
Résumé	iv
Abstract	v
Remerciements	vi
Table des matières	vii
Table des figures	ix
Liste des tableaux	xi
Liste des abréviations, sigle et acronymes	xii
0 INTRODUCTION GÉNÉRALE	1
0.1 Aperçu général et problématique	1
0.2 Objectifs	3
0.3 Limitations	3
0.4 Division du travail	4
1 ÉTAT DE L'ART ET FONDEMENTS THÉORIQUES	5
1.1 Généralités sur l'intelligence artificielle	5
1.1.1 Le modèle	6
1.1.2 L'apprentissage	7
1.1.3 Le processus de l'apprentissage automatique	9
1.1.4 L'apprentissage profond	10
1.1.5 Les Modèles de Langage à Grande Échelle (LLMs)	13
1.1.6 l'architecture Transformer [13, 59]	16
1.1.7 Résumé de l'évolution des chatbots	21
1.2 Généralités sur le Droit Congolais	22
1.2.1 Le système juridique Congolais [26]	22
1.2.2 Aperçu du Cadre Législatif Congolais [27]	23
1.3 Intersection entre Intelligence artificielle et Droit	26
1.3.1 Dans la recherche	27
1.3.2 Dans l'industrie	28
1.4 Résumé du chapitre	28
2 CONCEPTION ET DÉVELOPPEMENT	29
2.1 Les données	29
2.1.1 Les sources d'informations	29
2.1.2 Collecte des données	33
2.1.3 Pré-traitement et Formalisations des données	40
2.2 Du texte à la création d'Embeddings	41
2.2.1 La tokenisation	42
2.2.2 les Embeddings	42
2.3 Retrieval-Augmented Generation (RAG)	45
2.3.1 Le modèle d'embedding	46
2.3.2 Base de données vectorielle (Vector Store)	46
2.3.3 Le contexte	47
2.4 Conception de l'application Web	48

2.4.1	Le frontend	48
2.4.2	Le backend - API	53
2.5	Déploiement et mis en production	56
2.5.1	Configuration du pare-feu	56
2.5.2	Configuration du nom de domaine, DNS	57
2.5.3	Configuration du reserve proxy	58
2.6	Résumé du chapitre	59
3	ÉVALUATION ET FEEDBACK HUMAIN	60
3.1	Critères et méthodes d'évaluations	60
3.2	Évaluation des modèles existants	63
3.2.1	Évaluation qualitative	64
3.2.2	Évaluation quantitative	65
3.3	Évaluation de notre modèle	67
3.3.1	Évaluation qualitative	68
3.3.2	Évaluation quantitative	69
3.4	Résultats et perspectives	70
4	CONCLUSION	71
	Annexes	72
	BIBLIOGRAPHIE	82

TABLE DES FIGURES

Figure 1.1	Représentation de la valeur trois dans l'ensemble de données MNIST et sa matrice équivalente [9].	5
Figure 1.2	Procédure de subdivision du dataset. Image : Michael Galarnyk	7
Figure 1.3	Le processus de l'apprentissage automatique [52].	9
Figure 1.4	Volume de données/informations créées, capturées, copiées et consommées dans le monde de 2010 à 2020, avec des prévisions de 2021 à 2025. (Voir code 17) [60].	10
Figure 1.5	Le perceptron	11
Figure 1.6	XOR est considéré comme non-linéaire du fait qu'aucune ligne droite ne peut séparer les sorties de 0 et de 1 dans un espace bidimensionnel où les axes représentent les entrées. [29]	12
Figure 1.7	Un réseaux de neurones	12
Figure 1.8	Modèle uni-gramme, bi-gramme et tri-gramme. [1]	13
Figure 1.9	Architecture Transformer. [59]	16
Figure 1.10	(gauche) Scaled Dot-Product Attention. (droite) Multi-Head Attention consists of several attention layers running in parallel. [59]	18
Figure 1.11	La pyramide de Kelsen permet de visualiser la hiérarchie des normes. [71]	24
Figure 2.1	Résultat de la recherche via L'API Google Custom Search	33
Figure 2.2	Architecture du web crawler [le nôtre]	35
Figure 2.3	Documents sur Google Drive après téléchargement	38
Figure 2.4	Téléchargement en cours, approche itérative	40
Figure 2.5	Téléchargement en cours, approche récursive	40
Figure 2.6	Exemple de tokenisation avec Open AI tokenizer.	42
Figure 2.7	Représentation vectorielle d'un texte [3]	42
Figure 2.8	Architecture d'un système RAG inspiré par [44]	45
Figure 2.9	Création de la base de connaissance	46
Figure 2.10	Architecture du chatbot	48
Figure 2.11	Diagramme de séquence pour l'inscription	49
Figure 2.12	Diagramme de séquence pour la connexion	50
Figure 2.13	Diagramme de séquence pour la création d'un chat	51
Figure 2.14	Diagramme de séquence pour l'envoi d'un message	51
Figure 2.15	Diagramme de séquence pour modifier du chat	52
Figure 2.16	Diagramme de séquence pour la suppression du chat	52
Figure 2.17	Architecture du backend et services tiers	53
Figure 2.18	Documentation de l'API générer par API Platform	54
Figure 2.19	Diagramme relationnel d'entité	55
Figure 2.20	Architecture de production	56
Figure 2.21	Configuration Firewall	57
Figure 2.22	Configuration Domain Name System (DNS)	58
Figure 2.23	Achitecture	59
Figure 3.1	Extrait des évaluations reçues via Google Form	64

Figure 3.2	Résultats après évaluation des différents modèles ,en pourcentage. (voir Code 21)	64
Figure 3.3	Délai de réception du premier token, en secondes, après l'envoi de la demande d'Application Programming Interface (API). . . .	65
Figure 3.4	Temps nécessaire pour recevoir une réponse de 100 tokens. Estimation basée sur la latence (temps de réception du premier morceau) et la vitesse de sortie (nombre de tokens par seconde). . . .	65
Figure 3.5	Prix par token, représenté en USD par million de jetons. Le prix est un mélange des prix des tokens d'entrée et de sortie (ratio 3 1). . .	66
Figure 3.6	Juro : Réponse avec citation	67
Figure 3.7	Résultats après évaluation de Juro par rapport aux modèles existants, en pourcentage.	68
Figure 3.8	Temps moyen nécessaire pour recevoir une réponse.	69
Figure .1	Capture d'écran page de connexion	73
Figure .2	Capture d'écran page d'inscription	73
Figure .3	Capture d'écran page de chat	74
Figure .4	Capture d'écran page lecture de message	74
Figure .5	Capture d'écran modifier le chat	75
Figure .6	Capture d'écran supprimer le chat	75

LISTE DES TABLEAUX

Table 1	Association entre types d'apprentissage, tâches et algorithmes en Apprentissage Automatique (ML)	9
Table 2	Association entre tâches, architectures de modèles NN et type d'architectures	13
Table 3	Principaux modèles Open source	15
Table 4	Principaux modèles Entreprise	15
Table 5	Sortes de Lois dans le Système Juridique de la RDC	24
Table 6	Résumé des articles sur l'Intelligence Artificielle (IA) pour des applications juridiques	27
Table 7	Résumé des services et applications web d'Intelligence Artificielle (IA) pour des applications juridiques	28
Table 8	Sources d'information dans le système juridique Congolais	30
Table 9	Comparaison des méthodes d'embedding de mots	43
Table 10	Comparatif des modèles d'embedding disponibles	44
Table 11	Cas d'utilisation et leur importance	48
Table 12	Questions du test de magistrature Congolais 2022	62

LISTE DES ABRÉVIATIONS, SIGLE ET ACRONYMES

ML	Apprentissage Automatique
DL	Apprentissage Profond
NLP	Natural language processing
IA	Intelligence Artificielle
LLM	Modèle de Langage à Grande Échelle
LM	Modèle de Langage
SVM	Support Vector Machine
k-NN	K-Nearest Neighbour
PCA	Principal Component Analysis
t-SNE	t-Distributed Stochastic Neighbor Embedding
DQN	Deep Q Network
DBSCAN	Density-based Spatial Clustering of Applications with Noise
GD	Descente de Gradient
SGD	Descente de Gradient Stochastique
Adam	Adaptive Moment Estimation
RMSprop	Root Mean Squared Propagation
Adagrad	Adaptive Gradient Algorithm
RAG	Retrieval-Augmented Generation
FT	Fine-tuning
URL	Uniform Resource Locator
API	Application Programming Interface
JSON	JavaScript Object Notation
PDF	Portable Document Format

CSV	Comma-Separated Values
HTML	Hyper Text Markup Language
WWW	World Wide Web
XML	Extensible Markup Language
DOM	Document Object Model
MD5	Message-Digest algorithm 5
OCR	Optical Character Recognition
SQL	Structured Query Langage
ORM	Object-Relational Mapping
HTTP	Hypertext Transfert Protocol
HTTPS	Hypertext Transfert Protocol Secure
REST	Representational State Transfer
PHP	Hypertext Processor
IP	Internet Protocol
DNS	Domain Name System
RDC	République Démocratique du Congo



INTRODUCTION GÉNÉRALE

0.1 APERÇU GÉNÉRAL ET PROBLÉMATIQUE

Le Droit imprègne l'existence humaine, se manifestant comme une toile tissée à travers le spectre entier des interactions sociales. Il transcende les simples cadres institutionnels pour s'insérer dans les fibres mêmes de la vie quotidienne, régulant non seulement les transactions économiques et les relations étatiques, mais s'étendant également aux sphères les plus intimes des rapports humains. Des liens conjugaux, où il encadre des aspects aussi personnels que la fidélité ou le soutien dans la maladie, aux liens parentaux, où il définit la filiation, l'autorité parentale et les obligations mutuelles, le Droit se révèle omniprésent. Cette ubiquité du Droit souligne son caractère fondamental au sein de toute société organisée, où il émerge naturellement pour ordonner les comportements dès que des individus cohabitent. Le Droit, en tant que phénomène dynamique, reflète et s'adapte à l'évolution constante des normes sociales et des interactions humaines, rendant sa nature intrinsèquement complexe. [5]

Dans le contexte de la République Démocratique du Congo (RDC), le Droit s'articule autour d'un cadre juridique qui, tout en partageant des similitudes avec d'autres systèmes juridiques notamment le système juridique Français et Belge, se distingue par ses particularités inhérentes aux réalités historiques, culturelles et sociales du pays [39]. La RDC, héritière d'un système juridique mixte influencé par la tradition du droit civil, s'efforce de réconcilier les normes légales formelles avec les coutumes locales et les réalités socio-politiques propres à une société diverse et en mutation [12]. La structuration du Droit Congolais reflète ainsi un équilibre délicat entre les principes universels de justice et les spécificités locales, nécessitant une approche nuancée pour sa compréhension et son application (voir Section 1.2).

Seulement, l'accès à l'information juridique en RDC est exacerbé non seulement par la rareté des ressources numériques, mais également par la complexité intrinsèque du Droit qui rend sa compréhension ardue pour les non-initiés. Cette situation est d'autant plus critique que l'information juridique, essentielle à l'exercice des droits et à la participation citoyenne, demeure souvent confinée dans des textes légaux d'accès et de lecture complexes. La compréhension du Droit, avec ses termes techniques et ses concepts abstraits, requiert une médiation pédagogique pour être rendue accessible au grand public. Bien que des sites internet dédiés au Droit existent déjà et que des avocats soient disponibles pour fournir des conseils, ces ressources présentent certaines limites. Les sites internet, sans une compréhension approfondie des textes juridiques, ne parviennent pas toujours à résoudre efficacement les problématiques des utilisateurs. D'autre part, l'assistance d'un avocat, bien qu'utile, peut s'avérer coûteuse et donc inaccessible pour une grande partie de la population.

Conscients de ces limitations, nous nous tournons vers les nouvelles technologies, et plus spécifiquement vers l'IA et les Modèles de Langage à Grande Échelle (LLMs) tels que GPT4 [40], Gemini [55], (voir Table 3), etc... pour apporter une solution novatrice ; Ces modèles ont révolutionné la manière dont l'information est recherchée et assimilée.

Un LLM (voir Section 1.1.5) est une forme avancée d'intelligence artificielle spécialisée dans la compréhension et la génération de langage naturel, entraînée sur de vastes ensembles de données textuelles. Ces modèles utilisent des architectures complexes pour déchiffrer, interpréter et produire du texte de manière cohérente et contextuellement pertinente. Parmi ces architectures, celle du «transformer» [49] (voir Section 1.1.6), elle repose sur des mécanismes d'attention qui permettent au modèle de pondérer l'importance de différentes parties d'un texte lors de la génération ou de la compréhension de langage, rendant les LLM particulièrement efficaces pour une variété de tâches linguistiques complexes [59].

Toutefois, malgré leur puissance et leur polyvalence, ces modèles ont des limites, notamment en termes de contextualisation. Les LLM sont souvent entraînés sur des données majoritairement issues de sources globales, principalement en anglais, ce qui entraîne un manque de représentativité des contextes spécifiques tels que celui de la RDC. Cette lacune se manifeste par une compréhension insuffisante des nuances culturelles, légales et linguistiques propres au contexte Congolais, limitant ainsi leur efficacité à servir de médiateurs fiables pour la vulgarisation du Droit Congolais auprès du grand public [11].

Devant l'impératif de lever les barrières qui limitent l'accès et la compréhension de l'information juridique en RDC, notre mémoire vise à mettre au point un chatbot innovant. Ce chatbot, s'appuyant sur les technologies avancées des LLM, est conçu pour assimiler et refléter les nuances linguistiques et culturelles spécifiques au contexte juridique Congolais.

Un chatbot est un agent logiciel capable de simuler une conversation avec les utilisateurs en langage naturel à travers des applications de messagerie, des sites web, des applications mobiles ou par téléphone [43]. Inspiré par les fonctionnalités de ChatGPT, un modèle reconnu pour son interaction fluide et naturelle en langage humain, notre chatbot vise à devenir un outil efficace pour démocratiser l'accès au Droit, facilitant ainsi la compréhension et l'accessibilité de l'information juridique pour tous les Congolais.

Cette démarche soulève plusieurs questions de recherche fondamentales :

1. De quelle manière pouvons-nous ajuster des LLM pour qu'ils épousent fidèlement les nuances linguistiques et les spécificités culturelles du cadre juridique en RDC, assurant ainsi une pertinence et une efficacité maximales dans le contexte local ?
2. Comment concevoir un chatbot capable de simplifier et de traduire des concepts juridiques complexes de manière précise tout en les rendant accessibles à un public non spécialisé, sans compromettre l'exactitude et la fidélité des informations fournies ?
3. Quelles méthodologies devrions-nous mettre en place pour évaluer de manière rigoureuse la fiabilité et l'exactitude des données fournies par le chatbot, afin de

s'assurer que les utilisateurs reçoivent des explications justes et précises des textes de loi et des procédures juridiques ?

La réponse à ces interrogations constituera le socle de notre démarche, visant à rendre le Droit plus accessible et à favoriser une meilleure compréhension du système juridique Congolais parmi la population.

0.2 OBJECTIFS

Ce mémoire vise à créer un chatbot fondé sur un LLM pour rendre le système juridique Congolais plus accessible. Notre principal objectif est de simplifier les informations juridiques complexes, permettant ainsi à un public plus large de comprendre les tenants et aboutissants du Droit Congolais. Pour ce faire, nous nous concentrons sur plusieurs axes majeurs.

Tout d'abord, notre objectif est de concevoir une interface utilisateur conviviale pour le chatbot. Une interface intuitive permettra aux utilisateurs d'interagir plus facilement avec le chatbot, simplifiant ainsi la recherche et la compréhension des informations juridiques. Cette convivialité est un pilier essentiel pour rendre les concepts juridiques compréhensibles à un public non-initié.

Ensuite, notre démarche consiste à raffiner un LLM spécifiquement adapté à cette tâche. Ce processus comprend une personnalisation approfondie du modèle, notamment par le biais du fine-tuning et de l'analyse des similarités sémantiques, pour assurer une transmission précise et fiable des informations juridiques. L'accent est mis sur le développement d'un chatbot qui non seulement répond aux exigences de précision et de fiabilité, mais qui est également sensible aux nuances culturelles congolaises.

Enfin, une composante cruciale de ce mémoire est l'évaluation humaine. Nous prévoyons de mettre en place un système d'évaluation impliquant des utilisateurs réels. Cela nous permettra de vérifier la qualité des informations transmises par le chatbot et de nous assurer que ces informations restent fidèles aux concepts juridiques originaux, tout en étant compréhensibles pour le public ciblé. Cette évaluation continue guidera l'amélioration constante du chatbot pour garantir son efficacité et sa pertinence dans la vulgarisation du système juridique Congolais.

0.3 LIMITATIONS

Ce mémoire reconnaît plusieurs contraintes inhérentes à la conception d'un chatbot destiné à la vulgarisation du Droit Congolais. Premièrement, la richesse et la complexité du système juridique Congolais constituent un obstacle significatif. La législation et les procédures sont non seulement vastes mais aussi sujettes à des modifications fréquentes, ce qui complique l'intégration exhaustive de toutes les informations pertinentes dans le chatbot. De plus, l'obligation de mettre régulièrement à jour le système pour refléter l'évolution du Droit exige une vigilance et des ressources constantes.

Par ailleurs, la mise en place d'une base de données exhaustive est limitée par la disponibilité restreinte des ressources numériques. L'accès à des informations juridiques complètes, actuelles et fiables est freiné par la numérisation insuffisante des textes de loi, des décisions de justice, des traités internationaux, ainsi que des analyses doctrinales. De plus, les restrictions d'accès à des bases de données juridiques spécialisées, aux archives des avis d'experts et aux résumés de jurisprudence constituent un obstacle majeur. Ces données essentielles sont cruciales pour assurer l'efficacité et la fiabilité du chatbot dans le domaine juridique.

Malgré l'étendue des capacités des Modèles de Langage à Grande Échelle (LLMs), notre projet se concentrera principalement sur trois fonctionnalités clés adaptées aux besoins spécifiques de notre public cible : la réponse aux questions (Question Answering), la synthèse d'informations (Summarisation). Cette focalisation permettra de maximiser l'efficacité du chatbot dans le contexte juridique Congolais, tout en tenant compte des limitations précitées.

0.4 DIVISION DU TRAVAIL

En dehors de l'introduction, la partie conclusive et l'annexe, ce travail est organisé en trois chapitres comme suit :

Chapitre 1 Ce chapitre offrira une vue d'ensemble approfondie des [LLM](#), en explorant les avancées récentes, les applications marquantes et les défis associés, particulièrement en lien avec les applications juridiques et les chatbots. Cette section établira un cadre théorique et contextuel essentiel pour appréhender l'innovation et la pertinence du mémoire.

Chapitre 2 Ici, la discussion portera sur la conception technique et le développement du chatbot. Les choix technologiques, les architectures logicielles, ainsi que les stratégies d'ajustement fin et d'adaptation du [LLM](#) au contexte juridique Congolais seront décrits. Cette partie mettra en avant les aspects pratiques et techniques du mémoire, illustrant comment le chatbot a été concrètement mis en œuvre.

Chapitre 3 Le dernier chapitre se concentrera sur l'évaluation des performances du chatbot, en présentant les méthodologies d'évaluation, l'analyse des données recueillies et les retours des utilisateurs. L'objectif sera de mesurer l'efficacité du chatbot dans la simplification des informations juridiques et d'identifier des pistes d'amélioration pour en augmenter la pertinence et l'utilité.

ÉTAT DE L'ART ET FONDEMENTS THÉORIQUES

1.1 GÉNÉRALITÉS SUR L'INTELLIGENCE ARTIFICIELLE

L'Intelligence Artificielle (IA) peut être définie de diverses manières, mais une définition largement acceptée la décrit comme la création de systèmes informatiques capables d'accomplir des tâches qui nécessitent normalement l'intelligence humaine, telles que la reconnaissance de formes, la compréhension du langage, l'apprentissage et le raisonnement [13, 24].

Cette capacité à simuler des processus cognitifs humains est rendue possible grâce à une branche spécifique appelée "Apprentissage Automatique (ML)". Celle-ci permet aux ordinateurs d'apprendre à partir de données c'est à dire : d'identifier des motifs et de prendre des décisions.

Le ML repose donc sur la conception d'algorithmes ¹ capables d'accéder à des données et de les utiliser pour se former eux-mêmes à accomplir des tâches spécifiques. Cette conception table sur une solide base mathématique et statistique, qui fournit les outils nécessaires pour modéliser et comprendre les motifs complexes dans les données. Les données traitées sont souvent représentées sous forme de matrices. Par exemple, une image peut être représentée comme une matrice de pixels, où chaque élément de la matrice correspond à la valeur d'un pixel (voir Figure 1.1).

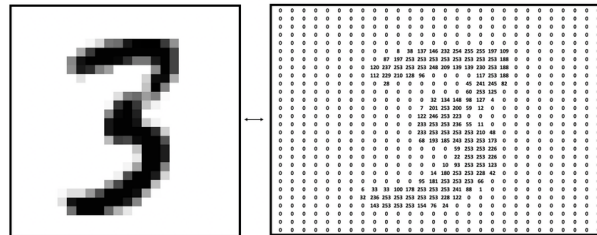


FIGURE 1.1 : Représentation de la valeur trois dans l'ensemble de données MNIST et sa matrice équivalente [9].

De même, les opérations sur les données, telles que les transformations appliquées par les couches d'un réseau de neurones, sont effectuées à l'aide d'opérations d'algèbre linéaire sur ces matrices.

¹ "Un algorithme est une suite finie et non ambiguë d'instructions et d'opérations permettant de résoudre une classe de problèmes." [68]

L'inférence ² statistique quant à elle permet aux algorithmes de faire des prédictions sur des données non vues auparavant, en se basant sur les probabilités extraites de l'ensemble de données d'apprentissage. Cela signifie que si un modèle est entraîné avec suffisamment de données représentatives, il peut inférer des résultats pour de nouvelles données basées sur les motifs appris [17].

En combinant l'inférence statistique avec l'algèbre linéaire, les algorithmes de ML peuvent donc apprendre à partir de données complexes, identifier des motifs subtils et faire des prédictions précises.

1.1.1 Le modèle

Dans ce contexte un modèle peut être défini comme une abstraction mathématique, un ensemble d'équations, ou un algorithme conçu pour effectuer une tâche spécifique. Ce modèle est formé à partir de données et apprend à effectuer cette tâche en identifiant des motifs, des relations, ou des structures au sein des données. Les modèles peuvent varier grandement en complexité, depuis des modèles linéaires simples jusqu'à des réseaux de neurones profonds très complexes [17].

Un modèle de ML est composé de trois composants principaux :

- **L'architecture du modèle** : C'est la structure sous-jacente du modèle, qui définit la manière dont les données d'entrée sont transformées en sorties. Par exemple, dans un réseau de neurones, l'architecture inclurait le nombre de couches cachées, le nombre de neurones dans chaque couche, le type d'activation utilisé, etc.
- **Les paramètres du modèle** : Ce sont les éléments du modèle qui sont ajustés au cours de l'entraînement pour minimiser la fonction d'erreur. Dans un réseau de neurones, les paramètres sont les poids et les biais associés à chaque connexion entre les neurones. Dans une régression linéaire, les paramètres seraient le coefficient de pente et l'ordonnée à l'origine. Par exemple :

$$y = mx + b \tag{1}$$

Où :

- y est la variable dépendante ou la sortie que l'on souhaite de prédire ou d'expliquer.
- x est la variable indépendante ou l'entrée pour faire des prédictions.
- m est le coefficient de pente, qui représente la variation attendue dans y pour une variation d'une unité dans x. En d'autres termes, il quantifie l'effet de la variable indépendante sur la variable dépendante.

² "Opération logique par laquelle on admet une proposition en vertu de sa liaison avec d'autres propositions déjà tenues pour vraies." - Le Robert

- b est l'ordonnée à l'origine, qui représente la valeur de y lorsque x est égal à zéro. C'est là où la ligne de régression coupe l'axe des ordonnées.

m et b sont les paramètres du modèle de régression linéaire. Pendant le processus d'entraînement, ces paramètres sont ajustés pour minimiser la différence entre les valeurs prédites par le modèle et les valeurs réelles des données d'entraînement. Cette différence est souvent mesurée en utilisant la somme des carrés des résidus ou une autre fonction de perte similaire [8].

- **L'algorithme d'apprentissage** : C'est la méthode utilisée pour ajuster les paramètres du modèle en fonction des données d'entrée et de la fonction d'erreur. L'algorithme d'apprentissage spécifie comment le modèle est entraîné, par exemple, en utilisant la descente de gradient pour minimiser la fonction d'erreur.

L'objectif d'un modèle est de généraliser à partir des données d'entraînement, c'est-à-dire d'être capable de faire des prédictions précises ou de prendre des décisions judicieuses sur de nouvelles données, jamais vues auparavant. La capacité d'un modèle à bien généraliser est essentielle pour son efficacité dans des applications réelles. La généralisation est souvent évaluée en utilisant un ensemble de données de test distinct de l'ensemble de données d'entraînement, permettant d'estimer la performance du modèle dans le monde réel [8, 17].

1.1.2 L'apprentissage

L'apprentissage est un processus itératif qui se fait au travers de ce qu'on appelle des "epochs" - chaque epoch ³ représentant un cycle complet de passage de l'ensemble des données d'apprentissage à travers le modèle. Pour évaluer et affiner la performance du modèle, les données sont généralement subdivisées en deux ou trois catégories : un ensemble d'entraînement (train), un ensemble de test, et parfois un ensemble de validation.

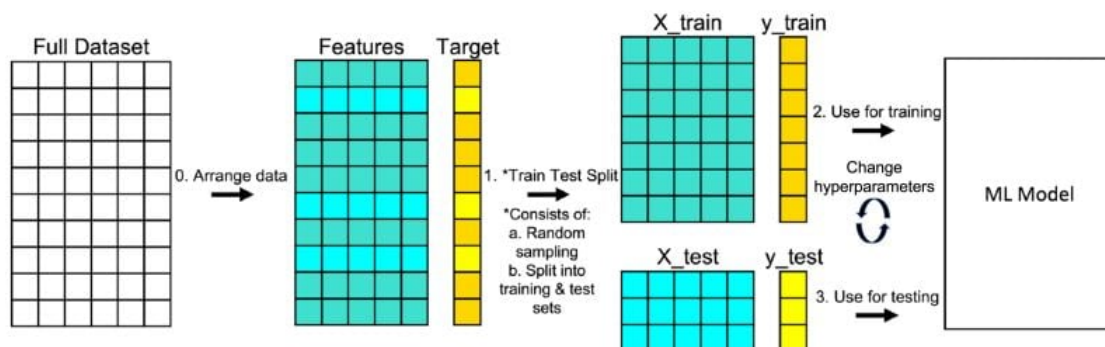


FIGURE 1.2 : Procédure de subdivision du dataset. | Image : Michael Galarnyk

³ "le nombre de passages d'un dataset d'entraînement par un algorithme. Un passage équivaut à un aller-retour. Le nombre d'epochs peut atteindre plusieurs milliers, car la procédure se répète indéfiniment jusqu'à ce que le taux d'erreurs du modèle soit suffisamment réduit." [33]

L'ensemble d'entraînement sert à ajuster les paramètres du modèle, tandis que l'ensemble de test est utilisé pour évaluer sa performance. L'ensemble de validation, lorsqu'il est présent, aide à affiner les hyperparamètres ⁴ du modèle et à éviter le surapprentissage ⁵ en fournissant une couche d'évaluation supplémentaire avant le test final. Tout au long de ce processus, tant les hyperparamètres que les paramètres ⁶ sont ajustés et optimisés dans le but de minimiser l'erreur de prédiction du modèle. Cet ajustement méthodique assure une amélioration continue de la performance du modèle, le rendant de plus en plus précis dans ses prédictions ou classifications sur des données non vues auparavant.

Les types d'apprentissage en ML sont généralement classés en trois grandes catégories :

- **Apprentissage supervisé** : Dans cette approche, le modèle apprend à partir d'un ensemble de données étiqueté, où chaque exemple d'entraînement comprend des entrées et les sorties correspondantes. Le but est de permettre au modèle de prédire la sortie associée à de nouvelles entrées inédites. Des exemples classiques incluent la régression linéaire et la classification.
- **Apprentissage non supervisé** : Ici, le modèle travaille sur des données non étiquetées, apprenant à identifier les structures et les motifs inhérents sans aucune indication de sortie désirée. Les algorithmes d'apprentissage non supervisé, tels que le clustering ou la réduction de dimensionnalité, visent à découvrir des groupements naturels dans les données ou à simplifier les données tout en conservant leur structure essentielle.
- **Apprentissage par renforcement** : Dans l'apprentissage par renforcement, un agent apprend à prendre des décisions en exécutant des actions dans un environnement afin de maximiser une certaine notion de récompense cumulative. C'est une approche dynamique où l'apprentissage est guidé par les interactions de l'agent avec l'environnement et les retours (récompenses) reçus pour ses actions.

Ces différents types d'apprentissage sont adaptés à diverses tâches. Ces tâches peuvent être regroupées en plusieurs catégories principales, chacune répondant à des objectifs spécifiques et utilisant des types de modèles appropriés. Voici une énumération de certaines des tâches les plus communes, comme détaillé par Giuseppe Bonaccorso dans [8].

Chacune de ces tâches exploite des principes et des algorithmes d'apprentissage automatique pour interpréter les données et faire des prédictions ou des classifications. Le choix de la tâche et de l'algorithme dépend largement du problème spécifique à résoudre et de la nature des données disponibles [22, 35].

Ce tableau n'est pas exhaustif mais donne un aperçu général des associations entre les types d'apprentissage, les tâches spécifiques, et les algorithmes.

⁴ Paramètres qui définissent la structure du modèle et son comportement, comme le taux d'apprentissage

⁵ "Le surajustement est un comportement indésirable d'apprentissage automatique qui se produit lorsque le modèle d'apprentissage automatique fournit des prédictions précises pour les données d'entraînement mais pas pour les nouvelles données." [45]

⁶ Poids et biais ajustés au cours de l'apprentissage

Type d'apprentissage	Tâche	Algorithmes
Supervisé	Régression	Régression linéaire, Forêts aléatoires
	Classification	SVM , k-NN , Réseaux de neurones
Non supervisé	Clustering	K-means, Clustering hiérarchique
	Réduction de dimensionnalité	PCA , t-SNE
	Détection d'anomalies	Isolation Forest, One-Class SVM
Par renforcement	Prise de décision	Q-learning, DQN

TABLE 1 : Association entre types d'apprentissage, tâches et algorithmes en [ML](#)

1.1.3 Le processus de l'apprentissage automatique

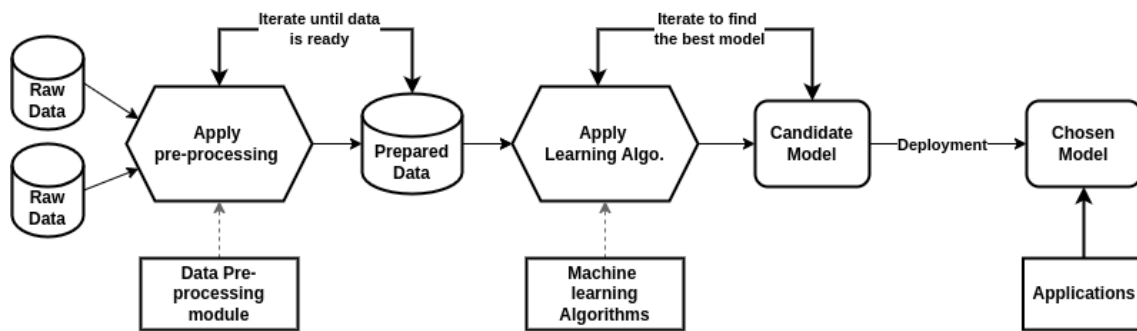


FIGURE 1.3 : Le processus de l'apprentissage automatique [52].

L'apprentissage est dit "**automatique**" en raison de la capacité des modèles à apprendre à partir de données sans intervention humaine explicite après leur programmation initiale. Comme nous pouvons le voir sur la figure 1.3, le processus de développement d'un modèle d'apprentissage automatique est présenté comme un flux de travail structuré en plusieurs étapes clés :

1. Tout commence avec les données brutes, ces données peuvent provenir de différentes sources et sont souvent hétérogènes, non structurées et peuvent contenir des erreurs ou des valeurs manquantes.
2. Les données brutes subissent un processus de pré-traitement, qui peut inclure le nettoyage, la normalisation, la transformation et la sélection des caractéristiques. Cette étape est cruciale pour préparer les données de manière à ce qu'elles puissent être utilisées efficacement par les algorithmes d'apprentissage automatique. Le pré-traitement est souvent un processus itératif, comme l'indique la flèche retournant vers la boîte de pré-traitement.
3. Après le pré-traitement, les données sont prêtes à être utilisées dans des modèles d'apprentissage automatique. Ces données préparées sont généralement plus propres, pertinentes et formatées de manière appropriée pour l'entraînement des modèles.

4. Les données préparées sont ensuite utilisées pour entraîner différents modèles à l'aide de divers algorithmes d'apprentissage automatique. Cette étape peut impliquer l'utilisation de techniques d'apprentissage supervisé, non supervisé ou par renforcement, selon la nature de la tâche à accomplir.
5. Plusieurs modèles candidats sont généralement produits et évalués pour déterminer lequel performe le mieux selon les critères de succès définis pour le projet. Ce processus peut nécessiter de nombreuses itérations pour affiner et optimiser les modèles.
6. Le modèle qui présente les meilleures performances est sélectionné comme le modèle final. C'est ce modèle qui sera déployé dans une application réelle.
7. Une fois le modèle choisi, il est déployé dans l'environnement de production où il peut être utilisé pour effectuer la tâche pour laquelle il a été conçu, comme faire des prédictions, classer des données ou automatiser des décisions.
8. Finalement, le modèle déployé est intégré à des applications ou systèmes plus larges, où il peut fournir une valeur ajoutée, comme améliorer l'expérience utilisateur, augmenter l'efficacité opérationnelle ou générer des perspectives à partir des données.

1.1.4 L'apprentissage profond

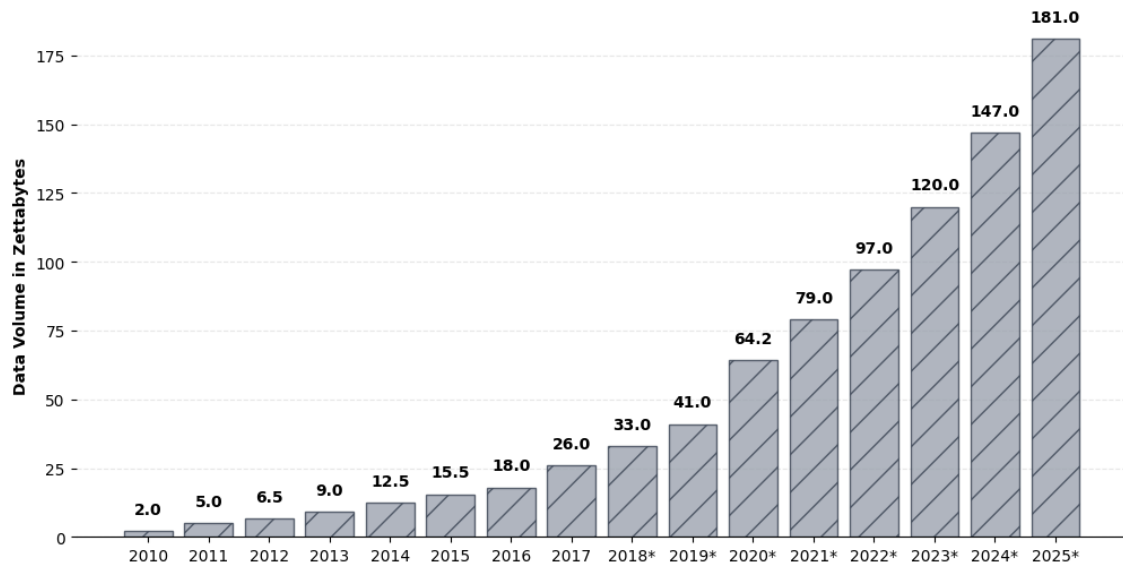


FIGURE 1.4 : Volume de données/informations créées, capturées, copiées et consommées dans le monde de 2010 à 2020, avec des prévisions de 2021 à 2025. (Voir code 17) [60].

L'avènement d'Internet a entraîné une explosion de la quantité de données générées, connue sous le nom de "**big data**". Le big data désigne de vastes ensembles de données si complexes et volumineux qu'ils sont difficiles à traiter avec des outils de gestion de base de données traditionnels [47].

Ces données, provenant de diverses sources telles que les médias sociaux, les transactions en ligne, les capteurs et les dispositifs connectés, représentent une mine d'or pour l'entraînement des modèles de **ML**, offrant une richesse d'informations pour améliorer la précision des prédictions et des décisions.

Cependant, l'apprentissage automatique traditionnel s'est avéré insuffisant pour traiter efficacement le volume, la variété et la vitesse du big data.

Cela a conduit à l'émergence de l'Apprentissage Profond (**DL**), un sous-ensemble de l'apprentissage automatique qui utilise des réseaux de neurones artificiels profonds. Le **DL** s'inspire de la structure et du fonctionnement du cerveau humain, permettant aux modèles de traiter des niveaux de complexité et de subtilité dans les données bien au-delà de la portée des approches traditionnelles de **ML**. Grâce à sa capacité à apprendre des caractéristiques hiérarchiques dans les données, le deep learning s'est avéré particulièrement efficace pour des tâches complexes telles que la reconnaissance d'images, la compréhension du langage naturel et la génération de contenu créatif [23].

Ces réseaux forment le cœur des algorithmes d'apprentissage profond et sont responsables de leur capacité à apprendre des représentations complexes dans d'immenses ensembles de données.

A. Le perceptron

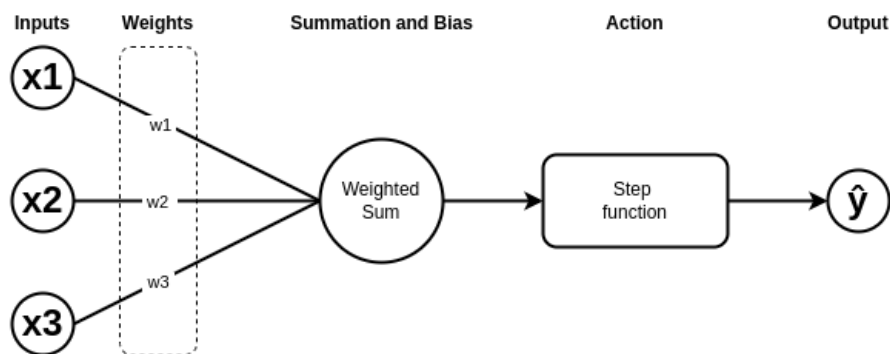


FIGURE 1.5 : Le perceptron

Le perceptron peut être considéré comme la brique élémentaire des réseaux de neurones. Développé dans les années 1950 par Frank Rosenblatt ⁷, il représente un modèle simplifié d'un neurone biologique. Un perceptron reçoit plusieurs entrées, chacune d'elles étant pondérée par un poids spécifique. Ces entrées pondérées sont ensuite sommées, et si la somme dépasse un certain seuil, le perceptron émet un signal de sortie. Cette sortie est déterminée par une fonction d'activation, qui, dans le cas du perceptron initial, est typiquement une fonction d'étape. Le perceptron ajuste ses poids à travers un proces-

⁷ "Frank Rosenblatt (né à New Rochelle le 11 juillet 1928 - mort le 11 juillet 1971) est un psychologue américain qui travaille sur l'intelligence artificielle. Principal représentant du « courant neuronal », qui voulait construire celle-ci à partir de la conception du réseau neuronal humain, il développe sur ce modèle le perceptron en 1957 à l'Université Cornell." [69]

sus d'apprentissage basé sur les erreurs commises, raffinant ainsi sa capacité à classer correctement les entrées [61].

Bien que le perceptron ait marqué un tournant dans le développement des algorithmes d'apprentissage automatique, il présente des limitations, notamment son incapacité à traiter des problèmes non linéairement séparables, comme l'opérateur logique XOR (voir Figure 1.6). Cette limitation a conduit à l'exploration de structures plus complexes, ouvrant la voie aux réseaux de neurones [38].

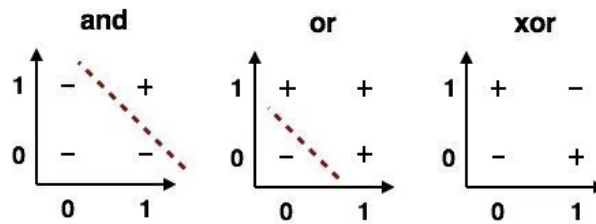


FIGURE 1.6 : XOR est considéré comme non-linéaire du fait qu'aucune ligne droite ne peut séparer les sorties de 0 et de 1 dans un espace bidimensionnel où les axes représentent les entrées. [29]

B. Les réseaux de neurones

Un réseau de neurones, dans son essence, est une collection de perceptrons (ou neurones artificiels) organisés en couches. Un réseau typique comprend une couche d'entrée, une ou plusieurs couches cachées, et une couche de sortie. Chaque neurone dans une couche est connecté à plusieurs neurones dans la couche suivante, permettant ainsi au réseau de traiter l'information de manière hiérarchique. Cette conception permet aux réseaux de neurones d'apprendre des relations complexes et non linéaires dans les données [29, 38].

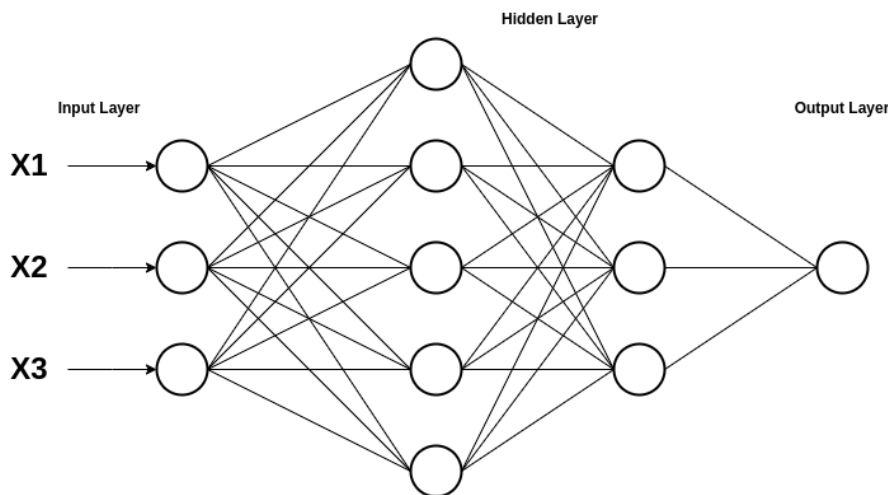


FIGURE 1.7 : Un réseaux de neurones

Les réseaux de neurones sont entraînés à l'aide d'algorithmes d'optimisation, comme la descente de gradient, qui ajustent les poids des connexions neuronales afin de minimiser l'erreur entre les prédictions du réseau et les véritables valeurs [4].

Ce tableau n'est pas exhaustif mais donne un aperçu général des associations entre les types d'apprentissage, les tâches spécifiques, et architectures.

Tâche	Architecture NN	Type d'apprentissage
Classification	MLP, CNN, RNN, LSTM, Transformers	Supervisé
Régression	MLP, CNN	Supervisé
Clustering	Auto-encodeurs, SOM	Non supervisé
Réduction de dimension	Auto-encodeurs, PCA Neural Network	Non supervisé
Détection d'anomalies	Auto-encodeurs, One-Class NN	Non supervisé
Prise de décision	DQN, Policy Gradient, Actor-Critic	Par renforcement

TABLE 2 : Association entre tâches, architectures de modèles NN et type d'apprentissage

1.1.5 Les Modèles de Langage à Grande Échelle (LLMs)

Un Modèle de Langage (LM) est un type de modèle utilisé dans le domaine du Natural language processing (NLP) pour comprendre, générer, et manipuler le langage humain. À son essence, un LM apprend la structure et les règles d'une langue à partir de grandes quantités de texte, permettant ainsi de prédire le mot suivant dans une phrase donnée, de générer du texte cohérent, ou d'accomplir d'autres tâches liées au langage [31].

Les modèles de langage sont entraînés à comprendre les probabilités d'apparition des mots ou des séquences de mots dans une langue. Initialement, des approches plus simples comme les modèles n-grammes étaient utilisées, où la prédiction du mot suivant se basait uniquement sur les $n-1$ mots précédents, sans tenir compte du contexte global ou des nuances sémantiques plus larges [1, 31].

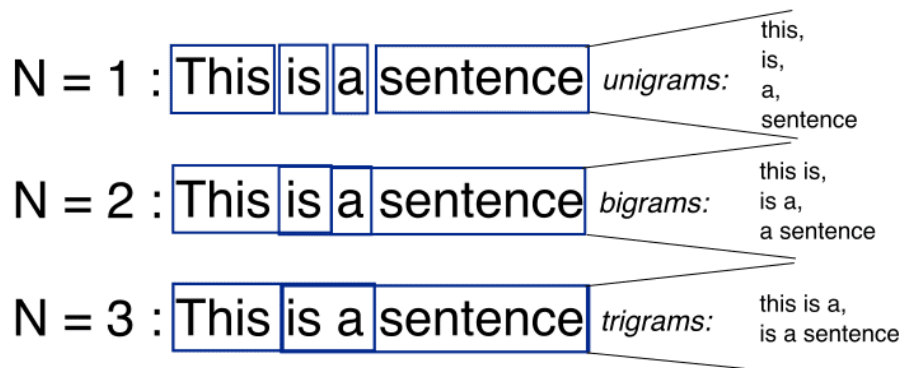


FIGURE 1.8 : Modèle uni-gramme, bi-gramme et tri-gramme. [1]

Prenons l'exemple d'un modèle trigramme (un cas spécifique où $n = 3$) pour illustrer comment cela fonctionne, Considérons la phrase "Le chat mange". Si nous voulons pré-

dire le mot suivant après "chat mange", un modèle tri-gramme regardera les occurrences de "chat mange" dans le corpus⁸ d'entraînement et calculera la probabilité du mot suivant chaque occurrence.

La probabilité d'un mot w_n sachant les deux mots précédents w_{n-2} et w_{n-1} dans un modèle trigramme est donnée par :

$$P(w_n | w_{n-2}, w_{n-1}) = \frac{C(w_{n-2}, w_{n-1}, w_n)}{C(w_{n-2}, w_{n-1})} \quad (2)$$

où :

- $P(w_n | w_{n-2}, w_{n-1})$ est la probabilité conditionnelle du mot w_n sachant les deux mots précédents w_{n-2} et w_{n-1} .
- $C(w_{n-2}, w_{n-1}, w_n)$ est le nombre d'occurrences de la séquence de trois mots w_{n-2}, w_{n-1}, w_n dans le corpus d'entraînement.
- $C(w_{n-2}, w_{n-1})$ est le nombre d'occurrences de la séquence de deux mots w_{n-2}, w_{n-1} dans le corpus.

Les mathématiques sous-jacentes du n-gramme ont été proposées pour la première fois par Markov (1913) dans [36].

En contraste, les Modèles de Langage à Grande Échelle (LLMs) utilisent des réseaux de neurones profonds pour apprendre à partir de vastes ensembles de données textuelles, leur permettant de capturer des nuances linguistiques et des dépendances complexes. Au cœur de ces modèles se trouve souvent l'architecture Transformer (voir Section 1.1.6), qui, grâce à son mécanisme d'attention, permet de considérer l'intégralité du texte en entrée, offrant une compréhension bien plus riche du contexte et des relations à longue distance.

Ces modèles sont dits "à grande échelle" parce qu'ils sont entraînés sur d'immenses volumes de données, ce qui leur permet de prédire le mot suivant en considérant tout le contexte donné. Leur performance dépend également du nombre de leurs paramètres, souvent en milliards, reflétant leur capacité à mémoriser et générer du langage. Plus un modèle a de paramètres, mieux il peut gérer un contexte étendu, ce qui améliore sa compréhension du langage et sa capacité à saisir les nuances [13].

A. Les modèles open-source

Après l'entraînement, les poids de ces modèles sont sauvegardés et souvent hébergés sur des plateformes de partage et de collaboration comme Hugging Face⁹, où ils deviennent accessibles à tous.

⁸ "Un corpus est un ensemble de documents, artistiques ou non, regroupés dans une optique précise. On peut utiliser des corpus dans plusieurs domaines : études littéraires, linguistiques, scientifiques, philosophie, etc." [65]

⁹ <https://huggingface.co/>

Les poids des modèles open-source peuvent être exécutés par quiconque en utilisant des outils et des bibliothèques adaptés, tels que Ollama ¹⁰ ou llama.cpp¹¹. Ces outils facilitent l'implémentation des modèles de langage dans divers environnements et applications, permettant une grande variété d'usages.

Modèle	Description
gemma	Gemma est une famille de modèles ouverts légers et de pointe construits par Google DeepMind [56].
llama2	Llama 2 est une collection de modèles linguistiques de base allant des paramètres 7B à 70B [58].
mistral	Le modèle 7B publié par Mistral AI, mis à jour à la version 0.2 [30].
Vicuna	Modèle de chat à usage général basé sur Llama et Llama 2 avec des contextes de 2K à 16K [15].
Starling	formé par apprentissage par renforcement à partir du retour d'information de L'IA, qui vise à améliorer l'utilité des chatbots. [73].
mixtral	Un modèle de mélange d'experts (MoE) de haute qualité avec des poids ouverts par Mistral AI.
llava	LLaVA est un nouveau modèle multimodal entraîné de bout en bout qui combine un encodeur de vision et Vicuna

TABLE 3 : Principaux modèles Open source

B. Les modèles entreprise

D'autre part, les modèles d'entreprise sont développés comme des services disponibles via des API payantes ou pas, ce qui représente un modèle d'accès différent par rapport aux modèles open-source. Ces API offrent aux entreprises la possibilité d'intégrer des capacités d'intelligence artificielle avancées dans leurs propres systèmes et applications, sans avoir à supporter directement les coûts et la complexité liés à l'entraînement, au déploiement, et à la maintenance de ces modèles.

Modèle	Description
GPT-4	La suite de GPT-3, GPT-4 améliore les capacités de génération de texte et de compréhension du modèle avec un nombre encore plus grand de paramètres [40].
Gemini	Le modèle créé par Google avec 137B paramètres [55].

TABLE 4 : Principaux modèles Entreprise

¹⁰ <https://github.com/ollama/>

¹¹ <https://github.com/ggerganov/llama.cpp>

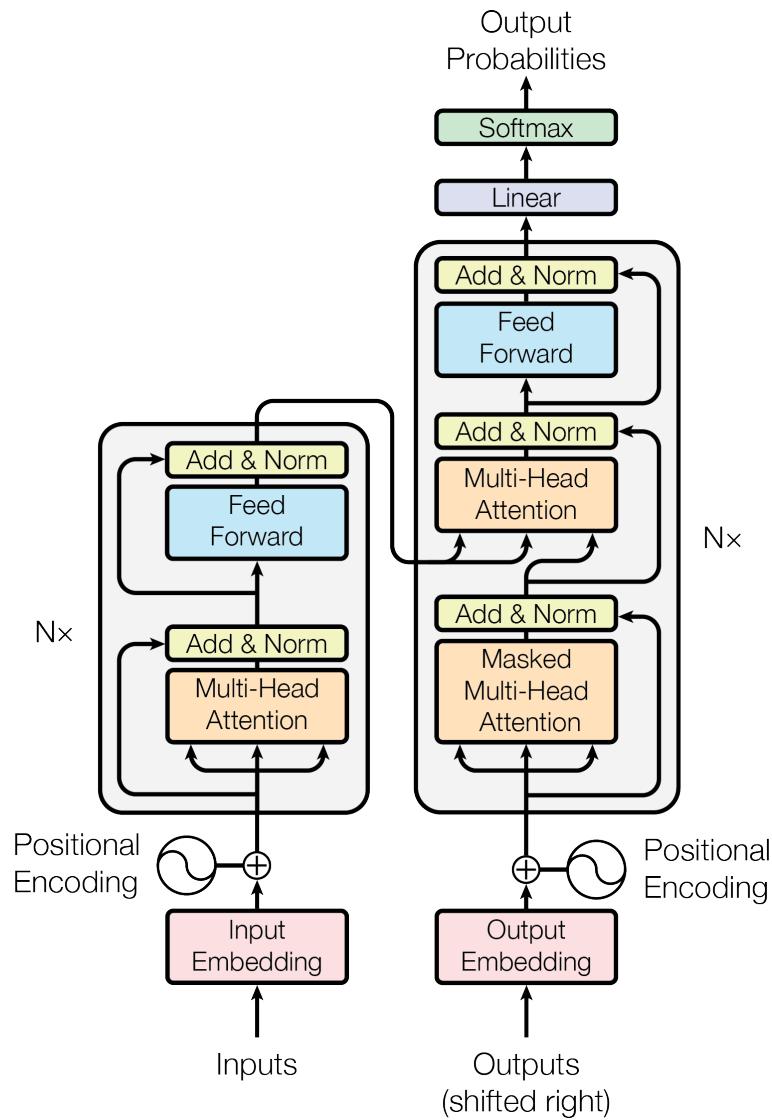
1.1.6 *l'architecture Transformer* [13, 59]

FIGURE 1.9 : Architecture Transformer. [59]

L'architecture Transformer, introduite dans l'article intitulé "Attention is All You Need" par Vaswani et al. en 2017, représente une avancée significative dans le domaine du NLP et du ML. Cette architecture se distingue par son mécanisme d'attention, permettant aux modèles de pondérer différemment les parties d'une séquence d'entrée lors de la prédiction de la séquence de sortie. Nous aborderons ici les composants clés de l'architecture Transformer ainsi que les formules associées, en nous efforçant de détailler chaque élément avec précision.

A. Encodage et Décodage

ENCODEUR

l'encodeur est responsable de comprendre le texte d'entrée. Il commence par convertir le texte en une série de vecteurs, qui sont des représentations numériques des mots. Ces vecteurs traversent plusieurs couches identiques, chacune comprenant deux sous-couches principales :

- **la self-attention** et le réseau de neurones feed-forward. La self-attention permet à chaque mot de prêter attention à tous les autres mots de la phrase, aidant ainsi à comprendre le contexte global.
- **Le réseau de neurones feed-forward** affine ensuite ces représentations. À la sortie, l'encodeur produit une série de vecteurs contextuels enrichis qui représentent la compréhension du texte d'entrée.

DÉCODEUR

Le décodeur, quant à lui, est responsable de générer le texte de sortie basé sur la compréhension fournie par l'encodeur. Il prend deux types d'entrées : les vecteurs produits par l'encodeur et le texte généré jusqu'à présent. Comme l'encodeur, le décodeur se compose de plusieurs couches identiques, mais avec des sous-couches supplémentaires. La sous-couche de masked self-attention masque les mots futurs pour empêcher la prédiction de regarder les réponses futures. La sous-couche d'attention croisée permet au décodeur de prêter attention aux vecteurs de l'encodeur, liant ainsi les informations d'entrée et de sortie.

Enfin, le réseau de neurones feed-forward affine les représentations après l'attention croisée. Le décodeur génère le texte de sortie, un mot à la fois, en se basant sur les vecteurs contextuels et les mots générés précédemment.

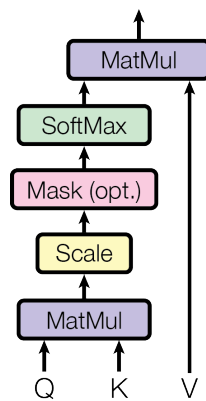
B. L'attention

L'attention est un mécanisme central dans l'architecture Transformer, permettant au modèle de se concentrer sur différentes parties de la séquence d'entrée de manière flexible et efficace. Voici comment cela fonctionne :

Le mécanisme d'attention, en particulier la self-attention, permet à chaque mot d'une séquence de texte de prêter attention à tous les autres mots de cette séquence. Cela signifie que pour chaque mot, le modèle peut déterminer quels autres mots sont les plus importants pour comprendre son contexte. Pour chaque paire de mots, le modèle calcule un score d'attention, qui indique l'importance relative de chaque mot par rapport aux autres.

Ces scores d'attention sont utilisés pour pondérer les représentations des mots, permettant au modèle de créer une nouvelle représentation pour chaque mot, intégrant le

Scaled Dot-Product Attention



Multi-Head Attention

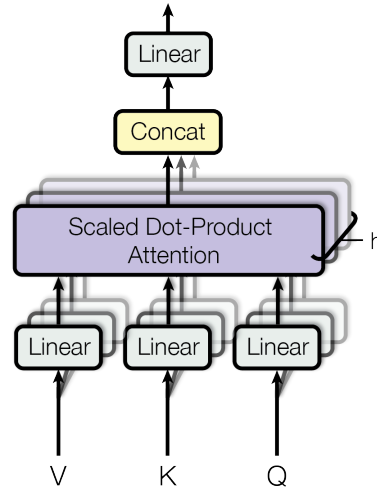


FIGURE 1.10 : (gauche) Scaled Dot-Product Attention. (droite) Multi-Head Attention consists of several attention layers running in parallel. [59]

contexte des mots pertinents. Cette capacité à capturer les relations contextuelles entre les mots est cruciale pour comprendre des structures complexes et des dépendances longues dans le texte.

L'attention est calculée en utilisant trois matrices : les matrices de requêtes (queries), de clés (keys) et de valeurs (values). Chaque mot est projeté en une requête, une clé et une valeur. Les scores d'attention sont obtenus en calculant le produit scalaire entre les requêtes et les clés, suivi d'une normalisation par softmax pour obtenir des poids de probabilité. Ces poids sont ensuite utilisés pour pondérer les valeurs, produisant ainsi une représentation contextuelle pondérée pour chaque mot.

SCALED DOT-PRODUCT ATTENTION

Trois matrices sont utilisées : requêtes (queries), clés (keys) et valeurs (values). Ces matrices sont dérivées des représentations des mots. Les scores d'attention sont obtenus en calculant le produit scalaire entre chaque requête et toutes les clés. Matriciellement, cela se fait en multipliant la matrice des requêtes par la transposée de la matrice des clés. Les scores obtenus sont divisés par la racine carrée de la dimension des clés ($\sqrt{d_k}$).

Cette étape de mise à l'échelle (scaling) aide à stabiliser les gradients lors de l'entraînement. Les scores mis à l'échelle sont ensuite normalisés à l'aide d'une fonction softmax pour obtenir des poids de probabilité. Cela convertit les scores en poids relatifs qui s'additionnent à 1.

Les poids de probabilité sont utilisés pour pondérer les valeurs correspondantes. La sortie de l'attention est alors une somme pondérée des valeurs, reflétant les parties importantes de la séquence d'entrée.

La matrice de sortie est alors obtenue par la formule suivante :

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3)$$

MULTI-HEAD ATTENTION

L'attention multi-têtes améliore le mécanisme de scaled dot-product attention en permettant au modèle de se concentrer sur différentes parties de la séquence de manière simultanée

Les entrées (requêtes, clés et valeurs) sont projetées en plusieurs sous-espaces de dimension inférieure par des couches linéaires distinctes. Si nous avons h têtes d'attention, chaque projection crée des versions plus petites des requêtes, clés et valeurs. Pour chaque tête, la scaled dot-product attention est calculée indépendamment. Cela permet à chaque tête de se concentrer sur différentes parties de la séquence ou de capturer différentes relations entre les mots.

Les sorties des h têtes d'attention sont concaténées pour former une seule matrice. Cette matrice représente une combinaison de différentes perspectives sur les relations de la séquence. La matrice concaténée passe par une dernière couche linéaire pour produire la sortie finale de la multi-head attention.

La formule générale pour l'attention multi-têtes est définie comme suit :

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (4)$$

où chaque head_i correspond à :

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (5)$$

Le principal avantage de la multi-head attention est sa capacité à capturer diverses caractéristiques des mots et de leurs relations contextuelles en parallèle. Cela permet au modèle d'extraire plus d'informations riches et variées par rapport à une seule tête d'attention..

c. Position-wise Feed-Forward Networks

Sont des réseaux de neurones appliqués de manière indépendante à chaque position de la séquence de texte. Ils consistent en deux couches linéaires séparées par une fonction d'activation ReLU. Ces réseaux permettent des transformations non linéaires des représentations des mots, enrichissant ainsi les capacités de capture des relations complexes dans le texte par le modèle Transformer.

La fonction du réseau de neurones peut être décrite par la formule suivante :

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (6)$$

où $\max(0, z)$ représente la fonction d'activation ReLU appliquée élément par élément au vecteur z .

D. *Embeddings et Softmax*

EMBEDDINGS

Les embeddings sont des représentations vectorielles des mots dans un espace continu de dimension inférieure. Ils sont utilisés pour convertir des mots en vecteurs numériques, ce qui permet au modèle Transformer de traiter des données textuelles de manière plus efficace. Chaque mot du vocabulaire est représenté par un vecteur dense de dimension fixe. Par exemple, un mot pourrait être représenté par un vecteur de 512 dimensions.

Les embeddings capturent les similarités sémantiques entre les mots. Par exemple, les mots "roi" et "reine" auront des vecteurs similaires. Dans l'architecture Transformer, les embeddings sont utilisés comme les premières représentations des mots. Ils sont enrichis par les mécanismes d'attention et les réseaux feed-forward dans les couches suivantes.

SOFTMAX

La fonction Softmax est utilisée à la fin du modèle Transformer, en particulier dans la tâche de génération de texte, pour convertir les scores bruts (logits) en probabilités. Softmax prend un vecteur de scores réels (logits) et les transforme en un vecteur de probabilités. Chaque score est exponentié, et les résultats sont normalisés pour que la somme des probabilités soit égale à 1.

Après que le décodeur ait produit les logits pour chaque mot potentiel, la fonction Softmax est appliquée pour obtenir une distribution de probabilité sur tous les mots possibles du vocabulaire. Le mot avec la plus haute probabilité est choisi comme le mot suivant dans la séquence générée.

E. *Positional Encoding*

Les Positional Encodings permettent aux modèles Transformer de capturer l'information de position dans les séquences de texte. Calculés à l'aide de fonctions sinusoïdales et cosinus, ils sont ajoutés aux embeddings des mots pour fournir une notion d'ordre. Cette technique permet aux Transformeurs de traiter des séquences de manière non séquentielle tout en conservant l'information de position essentielle pour comprendre le contexte et le sens des phrases.

1.1.7 Résumé de l'évolution des chatbots

L'aventure du Natural language processing (NLP) et de l'Intelligence Artificielle (IA) dans le domaine des chatbots est une histoire captivante de progrès technologique et d'innovation humaine. Le terme "chatbot" lui-même, une contraction de "chat" et "robot", trouve ses origines dans sa fonction initiale de système de dialogue textuel simulant le langage humain. Ces premières versions, essentiellement des programmes informatiques, utilisaient des interfaces d'entrée et de sortie pour simuler une expérience utilisateur mobile évoquant une conversation en temps réel. Toutefois, l'évolution des chatbots a largement dépassé cette interaction textuelle de base [2].

L'histoire du développement des chatbots et de l'intelligence artificielle (IA) dans la communication entre humains et machines est une saga captivante, jalonnée d'innovations technologiques et d'avancées majeures qui ont révolutionné notre interaction avec les machines. Dès 1906, les chaînes de Markov [25], inventées par le mathématicien russe Andrey Markov, ont jeté les bases de la prédiction de séquences aléatoires, permettant d'enseigner aux machines la création de données nouvelles, une première étape essentielle vers l'exploration du potentiel des chatbots.

L'aventure s'accélère en 1950 avec la proposition du test de Turing par Alan Turing [54], évaluant l'intelligence d'une machine à travers sa capacité à imiter le langage humain, un tournant décisif inspirant des décennies de recherche. L'apparition d'ELIZA [62] en 1966, conçue par Joseph Weizenbaum, marque la naissance des premiers chatbots, malgré une compréhension et une interaction encore limitées. L'évolution continue avec PARRY en 1972, simulant un patient schizophrène paranoïaque, offrant une nouvelle perspective sur les interactions conversationnelles. Les décennies suivantes voient la création de Racter et Jabberwacky, explorant respectivement la génération aléatoire de texte et l'apprentissage à partir d'interactions en ligne, des pas de géant vers des chatbots plus interactifs et personnalisés [2, 63, 72].

Le champ du ML a connu une avancée significative au début des années 2000 avec l'introduction de l'apprentissage profond et l'arrivée d'Internet, permettant aux ordinateurs de comprendre et d'interpréter l'information sous diverses formes. Cette période a vu les grandes entreprises technologiques pousser le progrès de l'intelligence artificielle, en utilisant la puissance computationnelle pour relever des défis complexes, avec des projets comme CALO qui pavent la voie à des assistants virtuels sophistiqués tels que Siri [57], marquant l'aube d'une nouvelle ère dans L'IA conversationnelle.

L'introduction de Modèles de Langage à Grande Échelle (LLMs) par OpenAI avec GPT et ses versions successives, ainsi que l'annonce de Google Bard (aujourd'hui Gemini), témoignent d'une évolution rapide vers des chatbots dotés d'une compréhension contextuelle et d'une capacité de raisonnement avancées, reflétant l'incroyable voyage depuis les premières techniques de correspondance de motifs à l'utilisation des architectures neurales Transformer (voir Section 1.1.6).

1.2 GÉNÉRALITÉS SUR LE DROIT CONGOLAIS

Le droit, tel qu'il est envisagé par le Professeur Ilunga Kabululu, est compris comme une règle de conduite sociale dont le respect est garanti par l'autorité publique. Cette définition met en lumière le rôle fondamental du Droit dans l'organisation de la société, en établissant des normes de comportement obligatoires pour ses membres. Le Droit existe dès l'instant où il y a société, soulignant l'adage « Ubi societas, ibi ius » (Là où il y a une société, il y a du Droit).

Cette perspective révèle le Droit non seulement comme un ensemble de règles coercitives mais aussi comme un phénomène vivant, qui naît, évolue et meurt au gré des changements sociétaux. Par conséquent, le Droit régit non seulement les rapports économiques et les interactions entre individus et État, mais s'étend aussi à des domaines aussi personnels que les relations familiales, illustrant sa présence omniprésente et son influence sur tous les aspects de la vie humaine [21].

le Droit en RDC, ainsi que dans toute société, s'avère être une structure complexe et dynamique, adaptative aux évolutions des rapports humains et aux exigences de justice et d'ordre social [21].

1.2.1 *Le système juridique Congolais* [26]

Pour aborder le thème du système juridique congolais, nous examinerons d'abord ce qu'est un système juridique, comment il est constitué, et comment cela s'applique spécifiquement à la République Démocratique du Congo (RDC).

Un système juridique est l'ensemble organisé des règles de droit applicables dans un espace géographique donné, régissant les rapports entre les personnes ainsi que ceux entre les personnes et l'État. Il comprend la structure institutionnelle chargée de créer, interpréter, et appliquer ces règles. Les systèmes juridiques se classifient généralement en différentes familles de droit, en fonction de leurs origines historiques, de leurs caractéristiques communes et de leurs principes fondateurs.

Il est évident que la distinction entre **le Droit objectif** et **le droit subjectif** est fondamentale. Le Droit objectif désigne l'ensemble des règles de Droit qui régissent les rapports sociaux et s'appliquent de manière impersonnelle, tandis que les droits subjectifs représentent les prérogatives spécifiques accordées aux individus ou entités, leur permettant de faire valoir leurs droits dans des situations concrètes.

Au niveau des sources formelles du Droit en RDC, la loi et la coutume apparaissent comme des piliers fondamentaux. La loi, dans son sens le plus large, englobe divers actes normatifs allant de la constitution aux ordonnances et décrets, caractérisés par leur impersonnalité, leur obligatorité, et leur permanence. La coutume, quant à elle, bien qu'elle doive céder devant la loi écrite, joue encore un rôle important, notamment dans des domaines moins réglementés par le Droit écrit.

La notion de droits subjectifs en Droit congolais est explorée à travers leur classification et leur preuve. Ces droits, attachés à l'individu, peuvent être patrimoniaux ou extrapatrimoniaux, reflétant respectivement des intérêts évaluables en argent et ceux qui ne le sont pas. Les droits patrimoniaux englobent, entre autres, les droits réels (comme la propriété) et les droits de créance, tandis que les droits extrapatrimoniaux incluent des droits tels que le droit à la vie, à la liberté, et au respect de la vie privée.

Le système juridique de la RDC s'inscrit dans la famille du droit civil, héritage du colonialisme belge. Cette influence se manifeste dans la prédominance du droit écrit et dans l'organisation judiciaire du pays. Néanmoins, la coutume occupe toujours une place significative, surtout en matière de droit de la famille et de droit foncier, où elle coexiste avec les règlements nationaux et internationaux. Cette coexistence illustre la complexité du système juridique Congolais, où les sources formelles et matérielles du droit, ainsi que les droits subjectifs et leur application, reflètent un mélange d'influences traditionnelles et modernes.

L'application de ce système juridique en RDC montre un équilibre entre le respect des normes internationales et la reconnaissance des spécificités locales. Cela est particulièrement visible dans le domaine des droits de l'homme, où les engagements internationaux de la RDC se conjuguent avec les réalités et pratiques locales pour façonner l'application des droits subjectifs dans le pays.

1.2.2 *Aperçu du Cadre Législatif Congolais [27]*

Pour approfondir l'aperçu du cadre législatif congolais, nous allons définir ce qu'est une loi, présenter les différentes sortes de lois existantes dans le système juridique de la RDC et expliquer comment une loi y est élaborée.

Une loi est définie comme une règle de conduite sociale, obligatoire, qui émane de l'autorité publique. Elle est d'application générale et permanente, pouvant être impérative ou supplétive. La loi au sens strict émane du Parlement, tandis que la loi au sens large inclut également les règlements, les contrats, et d'autres formes de textes normatifs .

Le système juridique congolais comprend plusieurs sortes de lois, chacune ayant sa spécificité et sa hiérarchie au sein de l'ordre juridique. Voici un tableau récapitulatif des sortes de lois, adaptées à la RDC :

Cet ajustement présente le même contenu mais dans un format légèrement différent, qui peut être plus familier ou approprié pour certains usages documentaires ou de présentation. Les différentes catégories de lois sont conçues pour structurer l'organisation de l'État, réguler la vie en société, et fournir un cadre pour l'administration publique.

ÉLABORATION D'UNE LOI

L'élaboration d'une loi en RDC suit un processus bien défini, impliquant différentes étapes et acteurs du système juridique et politique. Ce processus commence généralement par l'initiative législative, qui peut être prise par les membres du Parlement, le

Type de Loi	Description
Loi Constitutionnelle	La charte ou loi fondamentale de l'État, suprême sur le reste du droit
Lois Organiques	Relatives à l'organisation et fonctionnement des pouvoirs publics
Lois Ordinaires	Établies par le Parlement dans son domaine de compétence
Lois-Cadre	Définissent les grands principes d'une réforme, détaillée ensuite par le pouvoir réglementaire
Règlements	Actes de portée générale et impersonnelle édictés par l'exécutif

TABLE 5 : Sortes de Lois dans le Système Juridique de la RDC

gouvernement, ou par voie de pétition populaire dans certains cas ¹². Une fois proposée, une loi doit passer par diverses étapes de discussion, d'amendement, et d'approbation dans les deux chambres du Parlement congolais.

Après l'approbation par le Parlement, la loi est soumise au Président de la République pour promulgation et publication au Journal Officiel, devenant ainsi exécutoire. Ce processus d'élaboration législative est conçu pour assurer la participation démocratique, le débat ouvert, et la réflexion approfondie sur les propositions législatives avant qu'elles ne deviennent des lois [27, 28].

LA PYRAMIDE DE Kelsen [32, 46]

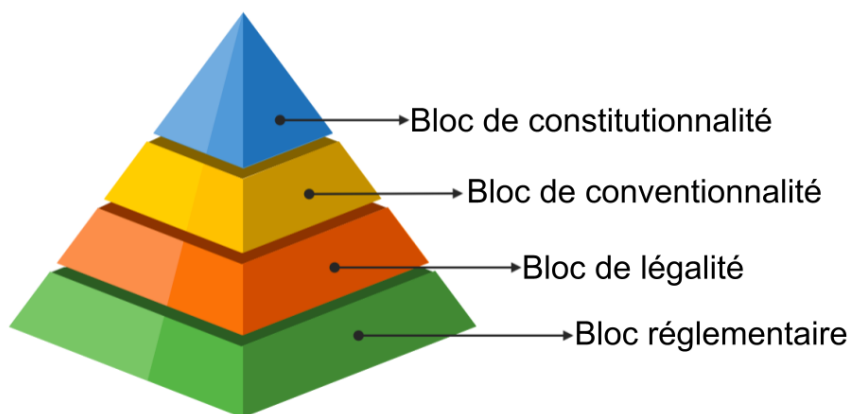


FIGURE 1.11 : La pyramide de Kelsen permet de visualiser la hiérarchie des normes. [71]

La pyramide de Kelsen est un concept essentiel pour comprendre la structure et la hiérarchie des normes juridiques dans un système juridique, y compris celui de la République

¹² <https://www.memoireonline.com/12/12/6552/Problematiche-du-droit-de-la-petition-dans-la-constitution-du-18-fevrier-2006-de-la-RDC.html>

Démocratique du Congo. Développée par le juriste Hans Kelsen ¹³, cette théorie illustre comment les différentes règles de droit s'ordonnent et se subordonnent les unes aux autres, formant une structure pyramidale.

Au sommet de cette pyramide se trouve la Constitution, qui est la norme suprême d'un pays. Toutes les autres normes juridiques dérivent de la Constitution et doivent lui être conformes. En RDC, comme dans de nombreux autres systèmes juridiques, la Constitution détermine les principes fondamentaux de l'État, les droits et libertés des citoyens, ainsi que l'organisation et le fonctionnement des pouvoirs publics.

Sous la Constitution, on trouve les lois organiques qui ont une valeur supérieure aux lois ordinaires. Les lois organiques précisent l'organisation et le fonctionnement des institutions prévues par la Constitution. Elles sont suivies des lois ordinaires qui régissent les aspects spécifiques de la vie sociale, économique, et juridique.

Les règlements exécutifs, comprenant les décrets et les arrêtés, viennent ensuite. Ces actes sont émis par le pouvoir exécutif pour assurer l'application des lois. Bien qu'inférieurs aux lois, ils jouent un rôle crucial dans la mise en œuvre des politiques publiques et des législations.

Enfin, à la base de la pyramide, se trouvent les normes réglementaires locales et les décisions de jurisprudence. Ces dernières, bien que ne constituant pas une source de droit formelle dans tous les systèmes juridiques, influencent significativement l'application et l'interprétation du droit.

La pyramide de Kelsen met en lumière la notion d'ordre juridique, où chaque norme tire sa validité de la norme immédiatement supérieure, jusqu'à la Constitution qui est la source ultime de toute validité juridique. Ce principe de hiérarchie des normes assure la cohérence et la systématique du droit, permettant ainsi de prévenir les conflits normatifs et de garantir une certaine prévisibilité et stabilité dans l'ordre juridique.

¹³ "Hans Kelsen, né le 11 octobre 1881 à Prague et mort le 19 avril 1973 à Orinda, est un juriste austro-américain, fils d'une famille juive de Bohême et de Galicie. Théoricien du droit, il est l'auteur de la « Théorie pure du droit », œuvre phare de la discipline." [70]

1.3 INTERSECTION ENTRE INTELLIGENCE ARTIFICIELLE ET DROIT

L'intersection entre l'Intelligence Artificielle (IA) et le droit représente un domaine d'étude fascinant et en constante évolution, reflétant l'intégration croissante des technologies avancées dans tous les aspects de la société. Alors que l'IA continue de progresser à un rythme exponentiel, son impact sur le domaine juridique devient de plus en plus significatif, soulevant à la fois des opportunités et des défis uniques pour les praticiens, les législateurs et les institutions juridiques.

L'application de l'IA dans le droit ouvre la voie à une multitude d'innovations, allant de l'automatisation des tâches documentaires à la prédiction des issues judiciaires, en passant par l'amélioration de l'accès à la justice. Ces technologies promettent d'accroître l'efficacité des processus légaux, de réduire les coûts et d'améliorer la qualité des services juridiques. Par exemple, les chatbots juridiques peuvent fournir des consultations préliminaires, tandis que les systèmes de traitement du langage naturel permettent d'analyser rapidement d'importants volumes de textes juridiques pour extraire des informations pertinentes.

Toutefois, l'adoption de l'IA dans le secteur juridique n'est pas sans soulever des questions complexes relatives à l'éthique, à la responsabilité, à la transparence et à la protection des données personnelles. Les systèmes d'IA, bien qu'incroyablement puissants, peuvent présenter des biais, des erreurs et des limitations qui, s'ils ne sont pas correctement adressés, pourraient compromettre les droits et libertés fondamentales des individus. En outre, l'émergence de l'IA générative et des modèles LLM pose de nouvelles interrogations sur les droits de propriété intellectuelle, notamment la paternité et la titularité des œuvres créées par ou avec l'aide de l'IA.

Face à ces enjeux, le monde juridique est appelé à réfléchir et à adapter ses cadres législatifs pour encadrer l'utilisation de l'IA, garantissant ainsi que son déploiement se fasse dans le respect des principes de justice, d'équité et de respect de la vie privée. Cette tâche implique une collaboration étroite entre juristes, technologues, chercheurs et décideurs politiques pour élaborer des normes et des régulations qui équilibrent les avantages de l'IA avec la protection des droits fondamentaux.

L'IA et du droit a conduit à une variété d'applications tant dans la recherche que dans l'industrie, notamment à travers le développement et le déploiement de services web innovants. Ces applications illustrent comment l'IA peut transformer les pratiques juridiques, offrir de nouveaux outils pour l'analyse et la gestion des données légales, et même réinventer l'accès à la justice pour le grand public.

1.3.1 Dans la recherche

Article	Résumé
Development of a Legal Document AI-Chatbot	Cet article présente le développement d'un chatbot AI pour la documentation juridique, exploitant le potentiel de L'IA, notamment des chatbots, pour simplifier la gestion des documents légaux. Le chatbot fonctionne en traitant les requêtes dans le contexte des documents chargés sur le serveur et en fournissant des réponses pertinentes [18].
ChatLaw : Open-Source Legal Large Language Model with Integrated External Knowledge Bases	ChatLaw est un modèle de langage large open-source conçu pour le domaine juridique chinois, visant à résoudre le problème des hallucinations du modèle lors de la récupération de données de référence. Le modèle intègre une base de connaissances externe pour améliorer la précision des réponses et est ouvert à la contribution communautaire [16].
Generative AI and US Intellectual Property Law	Cet article aborde les implications juridiques et éthiques de L'IA générative sur les droits d'auteur, la production de contenu, la collecte de données, la précision de l'information et les droits de propriété intellectuelle, en se concentrant sur les défis posés par les systèmes d'IA générative aux droits des créateurs de contenu humains dans le contexte du droit de la propriété intellectuelle aux États-Unis [42].

TABLE 6 : Résumé des articles sur l'Intelligence Artificielle (IA) pour des applications juridiques

Chaque jour, de nouvelles publications émergent, poussant les frontières de ce que ces technologies peuvent accomplir dans le secteur juridique.

1.3.2 Dans l'industrie

Service	Description
LegalMasterAI	Plateforme d'automatisation des tâches juridiques basée sur L'IA
AI Lawyer	Assistant juridique personnel basé sur L'IA pour les recherches et formalités administratives
LegalAI	Plateforme mondiale d'intelligence juridique avec accès à une vaste collection d'informations juridiques
vLex	Collection d'informations juridiques provenant de 100 pays
Harvey	Modèles linguistiques juridiques personnalisés pour les cabinets d'avocats
LawChatGPT	Outil d'IA pour des réponses juridiques en langage naturel
AI.law	Solutions d'automatisation juridique basées sur L'IA
LexisNexis	Solutions d'IA pour la recherche juridique et l'analyse des litiges
Paxton	Plateforme d'automatisation des contrats pour accélérer le processus de rédaction et d'examen
Luminance	Analyse de documents juridiques basée sur L'IA
RobinAI	Automatisation des contrats
Spellbook Legal	Automatisation des contrats avec recherche de clauses contractuelles
SuperLegalAI	Automatisation des tâches juridiques avec recherche de jurisprudence et rédaction de contrats

TABLE 7 : Résumé des services et applications web d'Intelligence Artificielle (IA) pour des applications juridiques

1.4 RÉSUMÉ DU CHAPITRE

Ce chapitre a fourni une vue d'ensemble approfondie des modèles LLM, en mettant en lumière les avancées récentes, les applications marquantes et les défis associés, avec un accent particulier sur les applications juridiques et les chatbots. Grâce à cette exploration, nous avons établi un cadre théorique et contextuel essentiel pour comprendre l'innovation et la pertinence de ce mémoire dans le domaine de l'intelligence artificielle appliquée au droit.

En conclusion, les perspectives offertes par les LLM dans le domaine juridique sont vastes et prometteuses, marquant le début d'une nouvelle ère dans laquelle l'intelligence artificielle jouera un rôle central dans la transformation et l'amélioration des services juridiques. Ce mémoire, en explorant ces technologies et leurs applications, aspire à contribuer à cette évolution, en mettant en évidence les opportunités et en naviguant à travers les défis inhérents à cette intégration pionnière.

CONCEPTION ET DÉVELOPPEMENT

La conception et le développement présentés dans ce mémoire s'articulent autour de deux axes principaux. D'une part, nous nous concentrerons sur le modèle [LLM](#), en suivant toutes les étapes de conception et de développement détaillées dans la section [1.1.3](#). Ce processus englobe la collecte initiale des données brutes jusqu'au déploiement final du modèle sélectionné dans un environnement de production. D'autre part, l'attention sera également portée sur le développement du chatbot, qui agit en tant qu'application web. Cette interface utilisateur servira de pont pour accéder au modèle [LLM](#), facilitant ainsi l'interaction entre les utilisateurs et le système juridique Congolais à travers le chatbot.

Ce dernier ne représente pas seulement un outil d'accès, mais aussi une manière intuitive et efficace de mettre en application les capacités du modèle [LLM](#), permettant aux utilisateurs d'obtenir des réponses et des informations juridiques pertinentes de manière interactive.

2.1 LES DONNÉES

Compte tenu de nos contraintes et objectifs (voir Section [0.3](#)), les données n'ont pas besoin d'être structurées selon un format particulier, à condition qu'elles soient disponibles sous forme textuelle. Cette flexibilité permet d'exploiter une large variété de sources d'information juridique sans nécessiter de processus de prétraitement complexe pour adapter les données à un format spécifique.

Notre jeu de données sera principalement constitué de documents juridiques provenant de diverses sources officielles et spécialisées, afin d'englober une vaste étendue de la législation et de la doctrine juridique congolaise.

Il est à noter que, bien que les données ne requièrent pas un format spécifique, leur qualité textuelle est essentielle. Cela implique un travail de vérification pour s'assurer de la fiabilité, de la pertinence et de l'actualité des informations collectées. Ce processus permettra de minimiser les erreurs et les ambiguïtés dans les réponses fournies par le chatbot, assurant ainsi une assistance juridique de qualité aux utilisateurs.

2.1.1 *Les sources d'informations*

La numérisation et l'adoption croissante d'Internet en République Démocratique du Congo (RDC) ont considérablement facilité l'accès aux documents légaux officiels. Dé-

Source	Description
Journal Officiel	Publications officielles qui contiennent les nouvelles lois, décrets, et annonces légales, offrant une source à jour des évolutions législatives.
Lois et Décrets	Textes législatifs et réglementaires qui forment la base du système juridique congolais, essentiels pour comprendre le cadre légal en vigueur.
Jurisprudences	Décisions de justice issues des tribunaux, fournissant des exemples concrets d'application des lois et des interprétations juridiques.
Articles d'Actualité Juridique	Articles publiés par des spécialistes et des médias juridiques, offrant des analyses et des commentaires sur les évolutions récentes du droit et les cas d'intérêt.
Doctrine	Contributions d'experts dans le domaine juridique, y compris des analyses détaillées, des critiques, et des interprétations de divers aspects du droit.

TABLE 8 : Sources d'information dans le système juridique Congolais

sormais, un nombre croissant de ces documents est accessible librement en ligne, offrant une opportunité sans précédent pour la recherche et l'analyse juridique. Des plateformes telles que Leganet.cd ¹ jouent un rôle crucial dans l'agrégation et la diffusion de ces documents, constituant ainsi une ressource inestimable pour les praticiens du droit, les chercheurs, et le grand public intéressé par le droit congolais.

Dans ce cadre nous envisageons d'explorer ces diverses sources en ligne. L'objectif est de collecter un large éventail de documents afin de capturer l'étendue et la profondeur du cadre juridique congolais. Cette démarche vise non seulement à fournir à notre modèle une richesse de connaissances et de perspectives sur le droit congolais mais aussi à garantir que les réponses générées soient à la fois informées et pertinentes, reflétant fidèlement les principes et les pratiques juridiques actuels.

Pour faciliter la découverte et l'exploration de ces ressources en ligne, nous envisageons d'utiliser l'API Google Custom Search ². Cet outil nous permettra d'automatiser la recherche et de visualiser rapidement les résultats, identifiant ainsi les sites les plus pertinents et fiables où les documents juridiques Congolais sont disponibles. Cette approche automatisée nous aidera à optimiser le processus de collecte de données, en assurant une couverture des sources d'information juridique pertinentes pour notre modèle.

¹ <https://leganet.cd>

² <https://developers.google.com/custom-search/v1/introduction>

LES MOTS CLÉS

Les mots-clés jouent un rôle crucial dans le processus de recherche, particulièrement lorsqu'il s'agit de collecter des données spécifiques à un domaine tel que le Droit Congolais. Ils servent de fondement pour affiner les requêtes de recherche et accéder efficacement à l'information pertinente.

```
keywords = [
    "Constitution", "civil", "Code pénal",
    "Code de travail", "Code foncier ", "Jurisprudence",
    "Droits humains", "Justice transitionnelle", "Droit minier",
    "l'environnement", "commercial", "sociétés",
    "Propriété intellectuelle", "famille", "Violence sexuelle et droit",
    "international humanitaire", "Institutions judiciaires congolaises",
    "Réforme judiciaire",
    "Lutte contre la corruption", "affaires", "Arbitrage et médiation",
    "bancaire et financier congolais", "fiscal congolais",
    "Contrats et obligations en congolais",
    "assurances", "santé", "l'éducation",
    "technologies de l'information", "humanitaire",
    "Participation politique et droit"
]
```

Listing 1 : Liste des mots clés à utiliser pour la recherche.

Cette liste englobe une gamme étendue de domaines juridiques, allant du cadre constitutionnel et législatif général à des domaines plus spécifiques tels que le droit minier, le droit fiscal, et le droit des affaires. Elle inclut également des aspects liés à la jurisprudence, aux publications officielles, et aux analyses d'experts.

Après avoir établi notre liste de mots-clés, nous pouvons créer une fonction qui interagit avec l'[API](#) Google Custom Search. Cette fonction se sert de la bibliothèque **Requests**³ pour lancer une requête GET vers l'[URL](#) de l'[API](#), en incorporant les paramètres que nous avons spécifiés. Les données renvoyées par l'[API](#) nous parviennent sous forme de JavaScript Object Notation ([JSON](#)).

```
params = {
    'q': query,
    'orTerms': ' '.join(keywords).lower(),
    'start': start,
    'key': "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
    'cx': 'xxxxxxxxxxxxxxxxxxxxxxxx',
    'lr': 'lang_fr',
    'fileType': 'pdf',
    'num': 10
}
```

Listing 2 : Dictionnaire des paramètres utile à l'utilisation de l'[API](#) Google Custom Search

³ <https://pypi.org/project/requests/>

Le dictionnaire **params** contient plusieurs paramètres configurés pour la requête de recherche, des explications détaillées sur l'utilisation des paramètres sont disponible sur la documentation ⁴ :

q : Le terme principal de la recherche.

orTerms : Une chaîne de caractères contenant tous les mots-clés joints par un espace, servant à élargir la recherche à ces termes connexes.

fileType : Restreint les résultats aux fichiers d'un type spécifique, ici des [PDF](#).

lr : Limite la recherche aux documents dans la langue spécifiée, ici le français.

num : Détermine le nombre de résultats de recherche à retourner, ici 10 résultats.

La fonction est conçue pour récupérer les résultats d'une seule page à la fois. Pour explorer un ensemble plus large de résultats, nous allons nous appuyer sur la constante **MAX PAGES**. En procédant à une itération, nous collecterons les résultats de plusieurs pages jusqu'à atteindre la limite fixée par **MAX PAGES**.

AGRÉGATION DES SOURCES

```
import requests
import json
import pickle

MAX_PAGES = 10

def search(start=1):
    url = 'https://www.googleapis.com/customsearch/v1'
    query = 'droit congolais'
    response = requests.get(url, params=params)
    return response.json()

try:
    websites = []
    for page in range(1, MAX_PAGES + 1):
        next_page = ((page - 1) * 10) + 1
        results = search(next_page)
        websites.extend(results.get("items", []))

    with open('data.pickle', 'wb') as f:
        pickle.dump(websites, f)
except Exception as e:
    raise e
```

Listing 3 : Fonction de recherche via l'API Google Custom Search

Après avoir recueilli les résultats, nous exploitons la bibliothèque **Pandas** ⁵, qui facilite à la fois la visualisation des données et leur enregistrement dans un fichier au format Comma-Separated Values ([CSV](#)).

⁴ <https://developers.google.com/custom-search/v1/reference/rest/v1/cse/list>

⁵ <https://pandas.pydata.org/>

```
import pandas as pd

rows = []
for item in websites:
    row = {
        'Title': item.get('title'),
        'Link': item.get('link'),
        'Snippet': item.get('snippet')
    }
    rows.append(row)

df = pd.DataFrame(rows)
df.head(100)
```

Listing 4 : Visualisation et exportation avec Pandas

	Title	Link	Snippet
0	Annuaire congolais de droit international ~ Ou...	https://www.larcier-intersentia.com/fr/annuaire...	Jan 12, 2024 ... L'Annuaire congolais de droit...
1	Reparation des dommages causes par les trouble...	https://lawcat.berkeley.edu/record/287141	Reparation des dommages causes par les trouble...
2	DROIT DE LA RÉPUBLIQUE DÉMOCRATIQUE DU CONGO ...	https://droitcongolais.info/	Le but de la vie de tout être humain est d'arr...
3	République démocratique du Congo : la situatio...	https://www.amnesty.org/fr/location/africa/eas...	Le droit à l'éducation a été bafoué. Contexte...
4	Barnes and Noble Les fondamentaux du droit pén...	https://www.hamiltonplace.com/products/product...	Le présent ouvrage intitulé Les fondamentaux d...
...
95	Le statut juridique des provinces dans la cons...	https://catalogue.leidenuniv.nl/UBL_V1:lib_asc...	Le statut juridique des provinces dans la cons...
96	RDC. Ituri, zone de non-droit - MSF-UREPH	https://msf-ureph.ch/publications/rdc-ituri-zo...	Atterrir à Bunia, ville principale de l'Ituri,...
97	Mampuya Kanunk'a Tshiabo Auguste, Les sanction...	https://journals.co.za/doi/abs/10.10520/EJC-69...	Dec 30, 2016 ... congolais violent le droi...
98	Editeur - Harmattan Congo - Librairie Mollat B...	https://www.mollat.com/Recherche/Editeur/0-773...	Un état des lieux du développement de l'Afriq...
99	République démocratique du Congo	https://www.justice-en-ligne.be/+Republique-d...	Le droit humanitaire et la Justice internation...

100 rows × 3 columns

FIGURE 2.1 : Résultat de la recherche via L'API Google Custom Search

Suite à l'emploi de l'API, nous avons réussi à récolter plus de 100 résultats distincts, couvrant une variété de sites et de plate-formes dédiés au domaine juridique, et plus spécifiquement au contexte Congolais. Ces résultats nous ouvre la voie vers l'étape suivante de notre processus de collecte de données.

2.1.2 Collecte des données

La collecte de données peut être effectuée via diverses méthodes, dépendant de la nature des données visées et du contexte d'utilisation. Parmi les techniques les plus courantes, on trouve les enquêtes et sondages, l'analyse documentaire, l'observation, et l'exploration de données en ligne. Chacune de ces méthodes a ses propres avantages et inconvénients en termes de coût, de temps, de précision et de couverture des données.

Dans le cadre de notre projet, nous avons opté pour le web scraping comme technique principale de collecte de données. Le web scraping est une méthode d'extraction automatique de données à partir de sites web. Cette technique nous permet de récupérer

efficacement un grand volume de documents, en particulier des fichiers [PDF](#) liés au droit congolais, disponibles sur divers sites et plateformes spécialisés.

Le web scraping repose sur l'utilisation de scripts ou de programmes qui envoient des requêtes aux serveurs web, analysent le [HTML](#) de la page pour identifier et extraire les informations nécessaires, et sauvegardent ces informations dans un format structuré pour une utilisation ultérieure. Dans notre cas, cette méthode est particulièrement pertinente pour télécharger automatiquement une multitude de documents juridiques, ce qui constitue une ressource inestimable pour notre base de données [14].

A. Conception d'un web crawler

Un web crawler, également connu sous le nom de web scraper ou spider, est un logiciel conçu pour parcourir automatiquement le World Wide Web ([www](#)) en suivant les liens entre les pages web. Il est principalement utilisé pour indexer le contenu des sites web, permettant aux moteurs de recherche de collecter, classer et servir les informations recherchées par les utilisateurs. Le processus implique la visite d'une page web, la lecture de son contenu, l'extraction des liens, puis le suivi de ces liens vers d'autres pages, et ainsi de suite, formant une toile étendue de données recueillies [10, 14, 67].

Le fonctionnement d'un web crawler, tel qu'initialement conceptualisé par Larry Page, repose sur un principe fondamental relativement simple, mais puissant dans sa capacité à organiser l'information sur le web. En substance, le processus peut être décrit par quelques étapes de base codées : le crawler commence par visiter une page web spécifiée, extrait tous les liens présents sur cette page, les enregistre pour un suivi ultérieur, puis répète ce processus de manière récursive pour chaque nouveau lien découvert. Ce mécanisme permet au crawler de naviguer à travers le vaste réseau du web, en cataloguant les ressources trouvées chemin faisant.

B. Notre approche

Dans notre cas, l'objectif s'affine vers une recherche plus ciblée : nous visons spécifiquement à localiser et récupérer des documents au format Portable Document Format ([PDF](#)). Ce choix implique une adaptation du processus de crawl standard pour filtrer les liens et ne retenir que ceux qui mènent directement à des fichiers [PDF](#). En d'autres termes, notre crawler est conçu pour non seulement suivre les liens, mais aussi pour identifier et sauvegarder les chemins vers des documents [PDF](#), en ignorant les autres types de fichiers ou de contenu web. Cette spécialisation permet une collecte de données plus efficace et pertinente pour nos besoins de recherche juridique, garantissant que seuls les documents correspondant à nos critères soient téléchargés et stockés pour une analyse ultérieure.

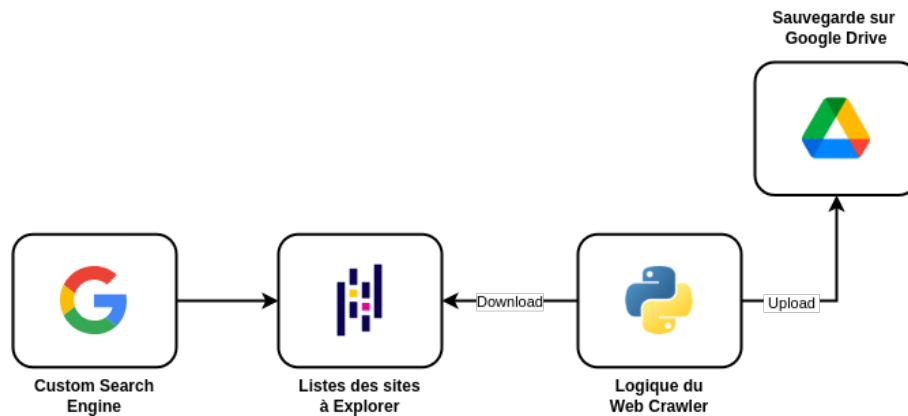


FIGURE 2.2 : Architecture du web crawler [le nôtre]

Comme illustré dans la figure 2.3, notre méthodologie s'aligne étroitement sur le modèle éprouvé par l'architecture de Google [10]. Le point d'entrée de notre processus est l'utilisation de l'API Google Custom Search pour générer un inventaire de sites web spécialisés dans le droit congolais. Cette première récolte de données est ensuite convertie en un DataFrame⁶ Pandas, ce qui facilite la visualisation et le filtrage des données pour affiner notre recherche. Après avoir épuré cette liste, un script Python prend le relais, parcourant méthodiquement les sites pour y télécharger les fichiers PDF disponibles. Enfin, dans la dernière étape de notre flux de travail, les PDF ainsi collectés sont stockés sur Google Drive⁷, assurant leur sauvegarde dans le cloud et permettant un accès aisé pour des analyses futures.

IMPLÉMENTATION ITÉRATIVE

```

import requests
from bs4 import BeautifulSoup

def crawl_link(url, from_root = False):
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'html.parser')
    links = soup.find_all('a', href=True)

    for link in links:
        href = link.get('href')
        if from_root:
            base = urlparse(url)
            href = f'{url}{href}'
        if href and href.endswith('.pdf'):
            download_pdf(href, DOWNLOAD_PATH)
            print(f'Downloaded {href}')
  
```

Listing 5 : Implémentation itérative du crawler

⁶ <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html>

⁷ <https://developers.google.com/drive/api/guides/about-sdk>

Le script (implémentation itérative) principal de notre crawler est bâti sur l'usage de deux puissantes bibliothèques Python : requests et BeautifulSoup ⁸. requests est la porte d'entrée pour accéder aux contenus des sites web en récupérant leur code Hyper Text Markup Language (HTML).

Une fois le contenu HTML obtenu, c'est là qu'intervient **BeautifulSoup**, une bibliothèque qui se distingue par sa capacité à analyser et à extraire des informations à partir de documents HTML et Extensible Markup Language (XML). Grâce à BeautifulSoup, nous pouvons naviguer dans la structure de la page web, parcourir les différents éléments du Document Object Model (DOM), et extraire précisément les données qui nous intéressent. Elle offre une variété de méthodes pour filtrer les éléments de la page, tels que les liens, en fonction de leur balisage et de leurs attributs [64].

Dans la pratique, notre script exploite BeautifulSoup pour identifier spécifiquement les liens de téléchargement de fichiers PDF. Nous pouvons isoler ces éléments grâce à leurs attributs distinctifs (par exemple, href se terminant par .pdf) et récupérer les URLs nécessaires pour lancer le téléchargement.

TÉLÉCHARGEMENTS DES FICHIERS

L'automatisation du téléchargement d'un volume conséquent de fichiers présente le risque inhérent de doublons, ce qui pourrait entraver l'efficacité de notre base de données et compliquer les analyses futures. Pour pallier ce problème et garantir l'unicité de chaque document, nous avons recours à une stratégie de renommage judicieuse : attribuer à chaque fichier un nom basé sur un hachage Message-Digest algorithm 5 (MD5) ⁹ de son contenu.

```
import os
import hashlib

def calculate_md5(file_path):
    hash_md5 = hashlib.md5()
    with open(file_path, 'rb') as f:
        for chunk in iter(lambda: f.read(4096), b''):
            hash_md5.update(chunk)
    return hash_md5.hexdigest()
```

Listing 6 : Fonction de hashage MD5 des fichiers

Même un changement minime dans le document entraînera la création d'un hachage radicalement différent. Par conséquent, si deux fichiers distincts portent le même nom mais ont des contenus différents, leur hachage MD5 révélera leur individualité. À l'inverse, si deux fichiers de noms différents présentent un contenu identique, ils seront assignés au même hachage MD5, ce qui nous permet de détecter et de gérer les doublons [48].

⁸ <https://pypi.org/project/beautifulsoup4/>

⁹ "MD5 est une fonction de hachage cryptographique qui calcule, à partir d'un fichier numérique, son empreinte numérique (en l'occurrence une séquence de 128 bits ou 32 caractères en notation hexadécimale) avec une probabilité très forte que deux fichiers différents donnent deux empreintes différentes." [66]

En renommant chaque fichier téléchargé avec son hachage MD5, nous établissons un système de dénomination unique et invariant, indépendant des noms de fichier originaux qui peuvent être arbitraires ou redondants. Cette méthode de renommage par hachage assure donc que chaque fichier dans notre collection est véritablement distinct et facilite l'organisation et la recherche dans la base de données, en éliminant les redondances et en optimisant l'espace de stockage.

```
import requests
import os
from urllib.parse import urlparse

def download_pdf(url, path):
    try:
        response = requests.get(url, stream=True)
        response.raise_for_status()
        filename = os.path.join(path, os.path.basename(urlparse(url).path))

        with open(filename, 'wb') as pdf_file:
            for chunk in response.iter_content(chunk_size=8192):
                pdf_file.write(chunk)

        hash = calculate_md5(filename)
        hashedfile = os.path.join(path, hash + '.pdf')
        os.rename(filename, hashedfile)
        print(f'Fichier téléchargé {filename} => {hashedfile}')
    except Exception as e:
        print(f'Erreur lors du téléchargement de {url}: {e}')
```

Listing 7 : Fonction de téléchargement des fichiers

Avec une stratégie de nommage optimisée désormais en place, nous sommes en mesure de procéder au téléchargement des fichiers en toute confiance, en alimentant simplement notre fonction avec l'Uniform Resource Locator (URL) correspondante. Nous exploitons l'environnement Google Colab ¹⁰ pour mener à bien ces opérations, bénéficiant de ses capacités de calcul et de sa compatibilité avec d'autres services Google. L'un des avantages notables de Colab est sa capacité à intégrer un disque virtuel Google Drive ¹¹, que nous pouvons monter et utiliser comme s'il s'agissait d'un répertoire local.

Les fichiers téléchargés peuvent être directement enregistrés sur le Drive monté sans nécessiter de transferts ultérieurs, permettant ainsi un accès immédiat et une organisation structurée. En utilisant le système de fichiers virtuel de Google Drive, nous simplifions considérablement le processus de sauvegarde et de partage des documents, tout en tirant parti de la robustesse et de la redondance des infrastructures de stockage cloud de Google.

¹⁰ <https://colab.research.google.com/>

¹¹ <https://colab.research.google.com/notebooks/io.ipynb>



















Name ↑	Owner	Last modified ▼	File size
 Oa2f92275508ed2ab6c16bd5e12a5670.pdf 	 google-drive@congo-la...	Mar 28, 2024 google-drive...	362 KB
 Oa6a60b711e882b2ae0e8a3bae9821fd.pdf 	 google-drive@congo-la...	Mar 28, 2024 google-drive...	384 KB
 Oa8c5045c7d2518d4cb5435acb040fcb.pdf 	 google-drive@congo-la...	Mar 28, 2024 google-drive...	35 KB
 Oa31e6396e2f1291126aa77ee13bd001.pdf 	 google-drive@congo-la...	Mar 28, 2024 google-drive...	264 KB
 Oa72f26186351a794e38f01eed93b6f1.pdf 	 google-drive@congo-la...	Mar 28, 2024 google-drive...	2.2 MB
 Oa76aec794af1604d143e6fae1b7e9f0.pdf 	 google-drive@congo-la...	Mar 28, 2024 google-drive...	149 KB

FIGURE 2.3 : Documents sur Google Drive après téléchargement

IMPLÉMENTATION RÉCURSIVE

Avec l'ajout d'une implémentation récursive à notre arsenal, notre crawler transcende les limitations précédentes et s'aventure désormais au-delà d'une page unique. Cette version avancée de notre crawler dispose désormais de la capacité de naviguer méthodiquement à travers un site entier, identifiant et téléchargeant non seulement les documents disponibles sur l'[URL](#) de départ mais aussi en suivant les liens vers d'autres pages pour récupérer les fichiers [PDF](#) associés.

```
from urllib.parse import urljoin, urlparse, urlunparse
from urllib.request import urlretrieve

def remove_fragment(url):
    parsed_url = urlparse(url)
    clean_parsed_url = parsed_url._replace(fragment="")
    return urlunparse(clean_parsed_url)
```

Listing 8 : Fonction de nettoyage d'[URL](#)

Le processus débute avec la fonction `remove_fragment`, qui purifie l'[URL](#) de toute portion inutile, permettant ainsi de concentrer le crawl sur les contenus essentiels des pages. Puis, `recursive_crawl` prend le relais, visitant chaque [URL](#) unique seulement une fois grâce à l'ensemble `visited_urls` qui assure une trace des pages déjà explorées et prévient les boucles infinies.

Lorsqu'une page est visitée, toute [URL](#) se terminant par `.pdf` est immédiatement transmise à `download_pdf`, qui procède au téléchargement du fichier et le sauvegarde dans un emplacement spécifique sur Google Drive. Tous les autres liens sont examinés et suivis s'ils appartiennent au même domaine que l'[URL](#) de départ, et ce processus se poursuit de façon récursive. C'est une exploration en profondeur, déployant une toile d'araignée qui s'étend sur l'intégralité du site.

Cela transforme notre crawler en un outil dynamique et exhaustif, capable d'extraire méthodiquement une richesse d'informations et de ressources souvent enfouies dans la structure des sites web. En exploitant `recursive_crawl`, nous pouvons cartographier

```

def recursive_crawl(start_url, url, visited_urls):
    url = remove_fragment(url)
    if url in visited_urls:
        return

    visited_urls.add(url)
    print(url)

    try:
        response = requests.get(url)
        response.raise_for_status()
        soup = BeautifulSoup(response.text, 'html.parser')

        if url.lower().endswith('.pdf'):
            download_pdf(url, '/content/drive/MyDrive/DATA/LawLLM/PDF/')
            print(f"File-Downloaded: {url}")

        for link in soup.find_all('a', href=True):
            href = link['href']

            if not urlparse(href).scheme:
                next_url = urljoin(url, href)
                recursive_crawl(start_url, next_url, visited_urls)

            elif urlparse(href).hostname == urlparse(start_url).hostname:
                recursive_crawl(start_url, href, visited_urls)

    except Exception as e:
        print(f"Error while processing {url}: {e}")

def recursive_crawl_link(start_url, destination):
    visited_urls = set()
    recursive_crawl(start_url, start_url, visited_urls)

```

Listing 9 : Implémentation récursive du crawler

un domaine entier, saisissant non seulement la superficie mais aussi plongeant dans les couches les plus profondes de contenu.

L'implémentation récursive signifie également que notre crawler peut fonctionner avec une autonomie et une efficacité élevées, nécessitant une supervision minimale et permettant aux chercheurs et aux professionnels de se concentrer sur l'analyse et l'exploitation des données collectées plutôt que sur le processus de collecte lui-même.

Cette méthode enrichit considérablement notre ensemble de données, non seulement en volume mais aussi en variété, fournissant une image complète et nuancée du domaine juridique congolais, qui est notre objectif premier. Elle souligne l'efficacité de la programmation récursive dans des applications pratiques telles que le web scraping,

démontrant comment une approche algorithmique bien pensée peut considérablement simplifier des tâches autrement ardues et complexes.

```
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/Ohada-Acte-Uniforme-2010-droit-commercial.pdf => 27ff5f0a8c5a1669db89036b5c38f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Arrete-2006-409-reglementation-des-emballages.pdf => e244c8e25fb96bf1cfcf
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Arrete-2006-409-reglementation-des-emballages.pdf => e244c8e25fb96bf1cfcf
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Arrete-2006-409-reglementation-des-emballages.pdf => e244c8e25fb96bf1cfcf
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Loi-1973-09-portant-sur-le-commerce.pdf => 84c9b9bf58c5dcfda602b59673d394f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Loi-2017-01-sous-traitance-secteur-privé.pdf => 9de87a59b7d7b958b8263b67e1
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Ordonnance-loi-1979-21-Reglementation-du-petit-commerce.pdf => af35819a6e
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Arrete-2019-16-application-droit-auteur.pdf => d825f598346c659a8d6b2666bd
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Arrete-2019-21-bareme-tarifaire-droits-auteur.pdf => 293c7cdf42f293666427f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Arrete-2007-17-taux-redevance-pour-copie-privée.pdf => 982d2bd8d1ba1082d5
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Arrete-2007-19-execution-des-oeuvres-musicales-et-dramatiques.pdf => bc77f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Loi-1982-01-propriete-industrielle.pdf => 7835e3787ee5f66bc023c7df537722f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/RDC-Ordonnance-loi-1986-33-Droits-auteur-et-droits-voisins.pdf => 7142df77af7f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/OHADA-Acte-uniforme-2015-Procédures-collectives.pdf => 931a408ed5ac7286897255f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/Ohada-Acte-Uniforme-2000-Comptabilite.pdf => 59f28c6f4710b6a19ee8715b9402dd32
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/Ohada-Acte-Uniforme-2000-Comptabilite-annexes.pdf => 575929bb3630743222f42c34f
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/Ohada-Plan-comptable-2017.pdf => 29caaa65f6872109bc492ba5b5dfeef
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/Ohada-Acte-Uniforme-2010-suretes.pdf => 44b3ac7e627d961b86b54df516ba8836
[Downloaded] /content/drive/MyDrive/DATA/LawLLM/PDF/Mauritanie-Decret-2021-22-registre-commerce-suretes-mobilieres.pdf => 50ab700f
```

FIGURE 2.4 : Téléchargement en cours, approche itérative

```
crawl('https://www.leganet.cd/legislation.htm', destination=DOWNLOAD_PATH)

[Crawling] https://www.leganet.cd/legislation.htm
[Crawling] https://www.leganet.cd/index.htm
[Crawling] https://www.leganet.cd/contact.htm
[Crawling] https://www.leganet.cd/Soutien.htm
[Crawling] https://www.leganet.cd/Modeles.htm
[Crawling] https://www.leganet.cd/nos_partenaires.htm
[Crawling] https://www.leganet.cd/J0.htm
[Crawling] https://www.leganet.cd/jurisprudence.htm
[Crawling] https://www.leganet.cd/doctrine.htm
[Crawling] https://www.leganet.cd/Doctrine.textes/generalites/Recension%20de%20l'ouvrage%20Guide%20Kandolo%20par%20Brozeck%20Kandolo.pdf
[Downloaded] /content/drive/MyDrive/DATA/PDF/Recension%20de%20l'ouvrage%20Guide%20Kandolo%20par%20Brozeck%20Kandolo.pdf => b
[Crawling] https://www.leganet.cd/Doctrine.textes/généralité/Intro.ilunga.2012.pdf
[Downloaded] /content/drive/MyDrive/DATA/PDF/Intro.ilunga.2012.pdf => a459b5dc74a92242b32dba870bd93f43
[Crawling] https://www.leganet.cd/Doctrine.textes/Decon/RELATIONS%20INTERNATIONALES%20AFRICAINES.2020.pdf
[Downloaded] /content/drive/MyDrive/DATA/PDF/RELATIONS%20INTERNATIONALES%20AFRICAINES.2020.pdf => 5356f7136c073beb63f670d736f
[Crawling] https://www.leganet.cd/Doctrine.textes/generalites/projet%20de%20politique%20nationale%20COMITE%20SCIENTIFIQUE%20R
[Downloaded] /content/drive/MyDrive/DATA/PDF/projet%20de%20politique%20nationale%20COMITE%20SCIENTIFIQUE%20RAPPORT%20FINAL%20
[Crawling] https://www.leganet.cd/Doctrine.textes/DroitPublic/Adm.territoire.solbena.pdf
[Downloaded] /content/drive/MyDrive/DATA/PDF/Adm.territoire.solbena.pdf => 56e29d654c70f610eb3783677bd314ba
[Crawling] https://www.leganet.cd/Doctrine.textes/Dadministratif/Notions.Ilunga.Etienne.pdf
[Downloaded] /content/drive/MyDrive/DATA/PDF/Notions.Ilunga.Etienne.pdf => 7578462659461047945b9802cf9231f6
[Crawling] https://www.leganet.cd/Doctrine.textes/Dadministratif/Decentralisation%20et%20Emplacement%20nouveau%20chef%20lieu%20territoire%20de%20
[Downloaded] /content/drive/MyDrive/DATA/PDF/Decentralisation%20et%20Emplacement%20nouveau%20chef%20lieu%20territoire%20de%20
[Crawling] https://www.leganet.cd/Doctrine.textes/DroitCiv/bail/Delaresiliationducontratdelocation2022.pdf
[Downloaded] /content/drive/MyDrive/DATA/PDF/Delaresiliationducontratdelocation2022.pdf => fbaf92f699ce6cda0b682470bbca875c
[Crawling] https://www.leganet.cd/Doctrine.textes/DroitCiv/Droitdelafamille/protection.fe.enf.succession.pdf
[Downloaded] /content/drive/MyDrive/DATA/PDF/protection.fe.enf.succession.pdf => 0eee45c211550b0867b2a661ef0c2d44
[Crawling] https://www.leganet.cd/Doctrine.textes/DroitCiv/Droitdelafamille/GUIDE%20POUR%20L'ENFANT%20ET%20L'ELEVE%20EN%20RDC
```

FIGURE 2.5 : Téléchargement en cours, approche récursive

2.1.3 Pré-traitement et Formalisations des données

Après avoir mené à bien le téléchargement de plus de 4000 documents, il est essentiel de reconnaître que bien que ces fichiers soient une mine d'informations, leur forme brute n'est pas directement utilisable. Par conséquent, l'étape de pré-traitement devient cruciale. Dans notre contexte, le pré-traitement consiste principalement à convertir le contenu de ces documents PDF en texte exploitable.

Le processus d'extraction de texte des PDF est un défi en soi, car il implique de décoder les diverses manières dont le contenu est encapsulé dans un document PDF. Cela peut inclure la gestion de la mise en page complexe, l'extraction de texte à partir d'images incorporées par Optical Character Recognition (OCR) et la préservation de la structure sémantique des informations.

EXTRACTION DU TEXTE

L'extraction de texte à partir de documents PDF est une opération qui peut être réalisée avec efficacité en utilisant la bibliothèque Fitz, également connue sous le nom de PyMuPDF ¹².

```
def extract_text(pdf_path):
    doc = fitz.open(pdf_path)
    text_content = []

    for page_num in range(len(doc)):
        page = doc.load_page(page_num)
        text = page.get_text("text")
        text_content.append({
            'page': page_num + 1,
            'type': 'text',
            'content': text
        })

    # extract text in image

    doc.close()
    return text_content
```

Listing 10 : Extraction du contenu d'un document PDF

EXTRACTION DU TEXTE SUR IMAGE

Pour extraire du texte à partir d'images, telles que des documents scannés ou numérisés, nous faisons appel à Tesseract. Tesseract est un moteur d'OCR open source, considéré comme l'un des plus précis disponibles. L'OCR est une technologie qui permet de convertir différents types de documents, tels que des images numérisées de texte imprimé, des captures d'écran ou des images contenant du texte, en données de texte modifiables et recherchables (voir Code 16) [51].

2.2 DU TEXTE À LA CRÉATION D'EMBEDDINGS

Dans le domaine de l'IA, et plus particulièrement dans celui des LLM, la compréhension directe du texte en langage naturel par un modèle est "impossible". Au lieu de cela, le texte doit être converti en une forme que le modèle peut traiter : une représentation mathématique. Ce processus est essentiel pour que les modèles apprennent et génèrent du langage humain de manière efficace [19].

¹² <https://pymupdf.readthedocs.io/en/latest/>

2.2.1 La tokenisation

Le premier pas vers la transformation du texte en données compréhensibles par une machine est la tokenisation. Ce processus consiste à découper le texte en morceaux plus petits, appelés tokens, qui peuvent être des mots, des phrases, ou même des parties de mots. Ces tokens servent de base pour la construction des représentations numériques.

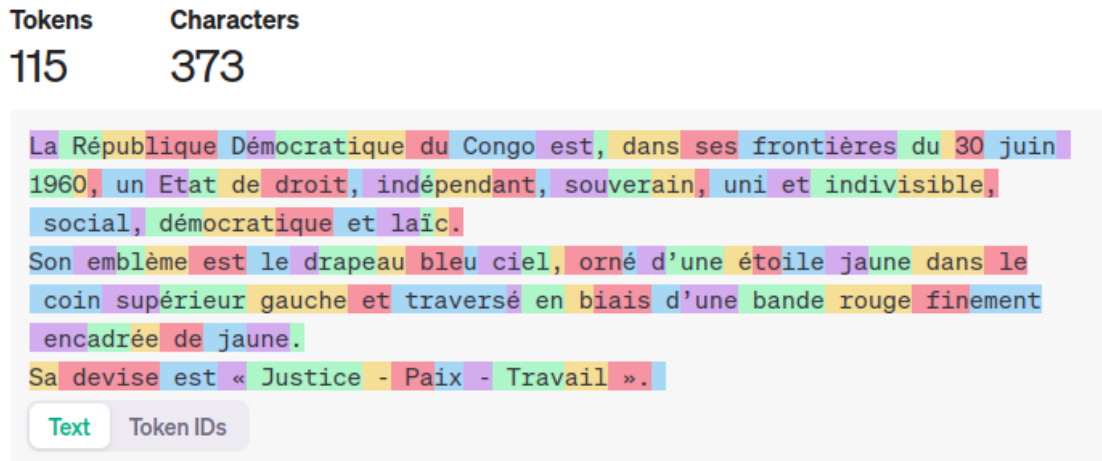


FIGURE 2.6 : Exemple de tokenisation avec Open AI tokenizer.

2.2.2 les Embeddings

Après la tokenisation, l'ensemble de tokens sont convertis en un vecteur via un processus appelé embedding. Les embeddings sont des représentations vectorielles denses de mots dans un espace continu de dimensions réduites, par rapport au vocabulaire total. Ces vecteurs capturent non seulement l'identité des mots mais aussi les aspects sémantiques et syntaxiques de ceux-ci, ce qui permet au modèle d'établir des liens entre les mots en fonction de leur contexte et de leur usage (voir section [d.](#)).

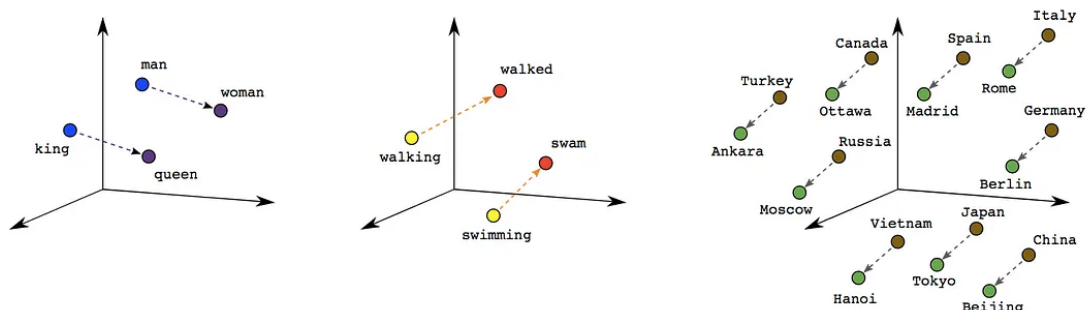


FIGURE 2.7 : Représentation vectorielle d'un texte [3]

MÉTHODES D'EMBEDDINGS

Méthode	Description
One-hot Enco- ding	Chaque mot du vocabulaire est représenté par un vecteur où un seul élément est "1" et tous les autres sont "0". Cette méthode est simple mais inefficace pour les grands vocabulaires et ne capture pas les relations sémantiques [50].
Word2Vec	Développé par Google, Word2Vec utilise un réseau de neurones pour apprendre des vecteurs de mots à partir de grands ensembles de données textuelles. Il existe deux architectures pour Word2Vec : Skip-gram et CBOW (Continuous Bag of Words). Word2Vec est efficace pour capturer les contextes des mots et les relations sémantiques entre eux [37].
GloVe	Une autre méthode populaire qui combine les avantages de la factorisation de la matrice des co-occurrences de mots et des méthodes contextuelles comme Word2Vec. GloVe est particulièrement bon pour saisir des relations subtiles entre les mots [41].
FastText	Proposé par Facebook, FastText étend Word2Vec pour considérer non seulement les mots mais aussi les séquences de caractères, ce qui le rend efficace pour gérer les mots rares ou mal orthographiés [7].
Transformers et BERT	Plus récemment, les modèles basés sur l'architecture Transformer, tels que BERT (Bidirectional Encoder Representations from Transformers), utilisent des embeddings contextuels, où la représentation d'un mot peut changer en fonction des mots qui l'entourent dans une phrase. Cela permet une compréhension plus nuancée et dynamique du texte [20].

TABLE 9 : Comparaison des méthodes d'embedding de mots

Il est important de noter que la qualité des embeddings a un impact direct sur la performance du modèle, rendant leur choix et leur implémentation des aspects critiques de la conception des systèmes de [NLP](#).

Il est également essentiel de souligner que la dimension du vecteur généré en fonction des données joue un rôle significatif. En effet, plus la dimension du vecteur est grande, plus il est capable de capturer des nuances fines et complexes du langage. Cependant, il faut également prendre en compte que des vecteurs de plus grandes dimensions requièrent plus de temps pour être générés et peuvent augmenter considérablement la charge computationnelle lors du traitement des données.

Modèle	Dimension	Disponibilité
Mistral-embed	1024	API payante
text-embedding-3-small	1536	API payante
text-embedding-3-large	3072	API payante
voyage-2	1024	API payante
Ollama embedding	1024	Gratuit

TABLE 10 : Comparatif des modèles d'embedding disponibles

Dans ce mémoire, nous avons opté pour des embeddings de dimension **1024**. Ce choix est stratégiquement motivé par deux facteurs clés :

- Premièrement, cette dimension offre une flexibilité optimale en termes d'interchangeabilité des modèles générant lesdits embeddings, permettant ainsi une intégration aisée de diverses technologies d'embedding sans compromettre la compatibilité ou la performance.
- Deuxièmement, en considérant les ressources financières à notre disposition, la dimension de 1024 est suffisamment large pour capturer une quantité significative de nuances sémantiques tout en restant gérable d'un point de vue computationnel, ce qui est crucial pour limiter les coûts liés à la puissance de calcul et au stockage.

Les différents modèles d'embedding imposent généralement une limite sur la taille du texte pouvant être traité en une seule fois, souvent désignée comme la limite de contexte. Pour contourner cette contrainte et optimiser le traitement des données, nous adopterons une approche où tous nos documents seront subdivisés en segments de 500 caractères chacun. Cette méthode de segmentation nous permet de gérer efficacement les limitations de taille d'entrée tout en préservant le contexte nécessaire à une analyse sémantique significative.

Une fois ces segments générés, nous procéderons à leur embedding avant de les stocker dans une base de données PostgreSQL ¹³. Cette base de données sera configurée avec l'extension "pgvector" ¹⁴, spécialement conçue pour supporter les types de données vectorielles. L'utilisation de PostgreSQL, combinée avec cette extension, nous offre une solution robuste et performante pour la gestion de grands volumes de données vectorielles.

¹³ <https://www.postgresql.org/>

¹⁴ <https://github.com/pgvector/pgvector>

2.3 RETRIEVAL-AUGMENTED GENERATION (RAG)

De prime abord, il est important de souligner que la création d'un LLM est un processus coûteux et exigeant en termes de données. Les modèles de langage existants, tels que GPT-4 [40], ont été développés avec une quantité massive de données et une puissance de calcul considérable. Cependant, pour adapter ces modèles au contexte juridique congolais, il serait peu pratique, voire impossible, de construire un LLM à partir de zéro en raison de contraintes de ressources.

Inspiré par Heydar Soudani [53], notre approche consiste à utiliser un modèle de langage existant comme point de départ (voir Table 3). En prenant un modèle pré-entraîné, nous pouvons bénéficier des connaissances et des capacités linguistiques déjà intégrées dans le modèle. Nous chercherons ensuite à affiner ce modèle pré-entraîné pour le rendre spécifique au système juridique congolais.

L'approche du RAG, combine la génération de texte avec des techniques de recherche d'informations. Le RAG permet au chatbot d'accéder à une base de connaissances juridiques étendue et de récupérer des informations pertinentes en réponse aux requêtes des utilisateurs. En utilisant des index de recherche efficaces et des algorithmes de récupération d'informations, le chatbot peut fournir des réponses bien informées en s'appuyant sur une grande variété de sources [34].

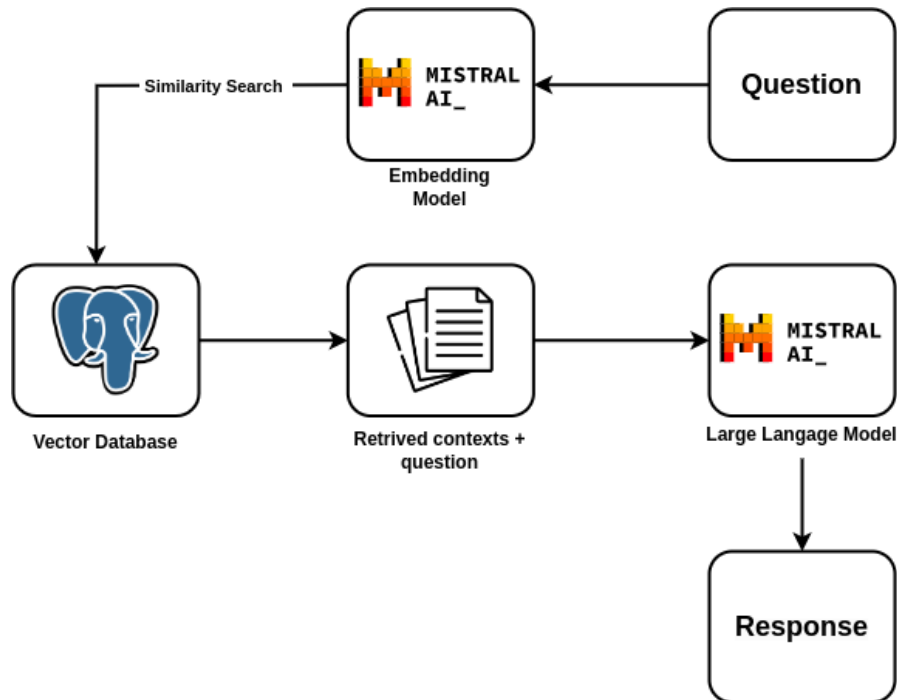


FIGURE 2.8 : Architecture d'un système RAG inspiré par [44]

L'illustration ci-dessus présente l'architecture de notre approche adaptée pour améliorer la pertinence et la précision des réponses générées par un LLM dans le domaine juridique congolais.

Le processus débute lorsqu'une question est posée par l'utilisateur, cette question sert de point de départ pour l'ensemble du pipeline :

2.3.1 Le modèle d'embedding

La question est d'abord transformée en une représentation vectorielle à l'aide d'un modèle d'embedding (voir Table 10). L'embedding de la question est ensuite utilisé pour effectuer une recherche par similarité dans une base de données vectorielle.

2.3.2 Base de données vectorielle (Vector Store)

Cette base de données, alimentée par *PostgreSQL* et l'extension *pgvector*, contient des embeddings de documents récoltés précédemment.

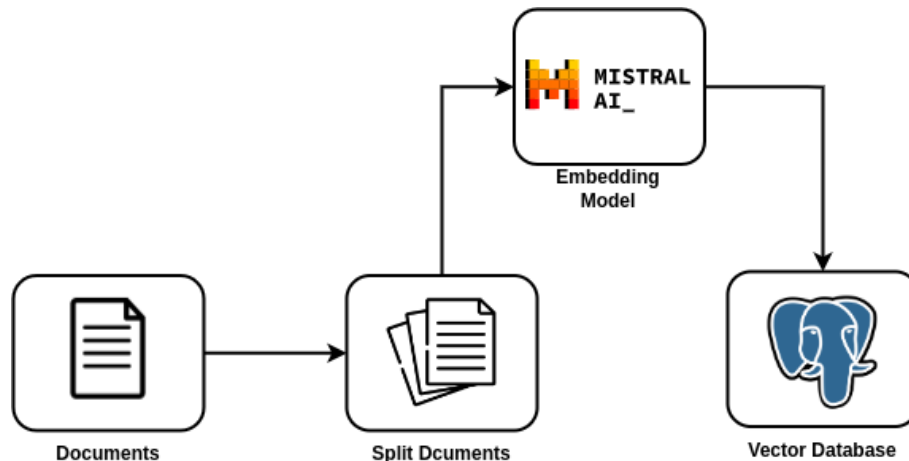


FIGURE 2.9 : Création de la base de connaissance

Les documents initiaux sont divisés en segments plus petits. Cette segmentation facilite le traitement et l'analyse des textes en morceaux gérables, ici en sous-parties de 500 caractères et chaque segment de texte résultant de la division est ensuite envoyé à un modèle d'embedding et enfin stocké.

Pour mesurer la similarité entre l'embedding de la question et les embeddings des documents stockés, diverses fonctions de recherche de similarité peuvent être employées. Parmi celles-ci, les plus couramment utilisées sont :

<-> **Distance Euclidienne (L2)** : La distance euclidienne est une mesure de la distance directe entre deux points dans un espace vectoriel.

<#> **Produit Scalaire** : Le produit scalaire mesure la similarité en calculant la somme des produits des composantes correspondantes de deux vecteurs.

<=> **Distance Cosinus** : La distance cosinus mesure la similarité entre deux vecteurs en calculant le cosinus de l'angle entre eux.

<+> **Distance de Manhattan (L1)** : La distance L1, également connue sous le nom de distance de Manhattan, mesure la somme des valeurs absolues des différences entre les composantes correspondantes de deux vecteurs.

Par exemple nous pouvons récupérer le contenu des 5 documents les plus proches de la question de l'utilisateur avec les requêtes Structured Query Language (SQL) suivantes :

```
SELECT * FROM documents ORDER BY embedding <-> '[3,1,2]' LIMIT 5;
SELECT * FROM documents ORDER BY (embedding <#> '[3,1,2]') * -1 LIMIT 5;
SELECT * FROM documents ORDER BY 1 - (embedding <=> '[3,1,2]') LIMIT 5;
```

Listing 11 : Exemple de requêtes SQL sur les distances entre vecteurs

2.3.3 Le contexte

```
public string $systemMessageTemplate = <<<TEMPLATE
    Ton nom est Juro crée par Bernard Ngandu,
    Tu es un expert en Droit Congolais (RDC), ton objectif est de vulgariser
    le droit congolais et de répondre aux questions en utilisant
    les éléments de CONTEXT suivants. si aucun CONTEXT ne t'es fourni,
    précise que tu ne peux pas répondre directement à la question

    CONTEXT : {context}
TEMPLATE;
```

Listing 12 : Le message système

Une fois que des documents pertinents ont été trouvés, ils permettent de créer un contexte riche et informatif. Ce contexte contient tous les éléments nécessaires pour répondre précisément à la question de l'utilisateur et joue un rôle crucial dans la direction que prendra la génération de la réponse par le LLM.

Pour guider le LLM de manière efficace, nous utiliserons un message système. Ce message contient des instructions spécifiques que le LLM devra suivre pour générer une réponse. il sert de cadre et de guide, assurant que le modèle comprend bien la nature de la question, les attentes en termes de format et de contenu de la réponse, et les aspects spécifiques du contexte qui doivent être pris en compte.

2.4 CONCEPTION DE L'APPLICATION WEB

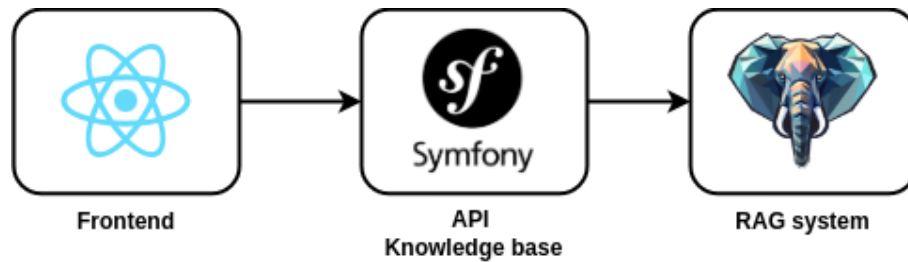


FIGURE 2.10 : Architecture du chatbot

Maintenant que nous avons les données et l'architecture [RAG](#) en place, nous allons aborder la conception et le développement du chatbot. Ce chatbot est constitué de trois principaux composants : un frontend, une [API](#), et le système [RAG](#). Dans cette section, nous détaillerons le frontend et l'[API](#), ainsi que leur interaction avec le système [RAG](#), nous choisissons le nom "Juro" pour notre chatbot, inspiré de la contraction "Juridique" et "Robot".

2.4.1 *Le frontend*

Le frontend de notre application est développé en utilisant React ¹⁵, une bibliothèque JavaScript populaire pour la construction d'interfaces utilisateur dynamiques et réactives. Le frontend sert de point de contact principal pour les utilisateurs, leur permettant de poser des questions et de recevoir des réponses de manière intuitive et conviviale

LES FONCTIONNALITÉS

Fonctionnalité	Importance
S'inscrire	Haute
S'authentifier	Haute
Créer un nouveau chat	Haute
Consulter les chats	Moyenne
Envoyer un message	Haute
Renommer un chat	Faible
Supprimer un chat	Moyenne
Choisir un chat proposé	Moyenne

TABLE 11 : Cas d'utilisation et leur importance

¹⁵ <https://fr.react.dev/>

Les fonctionnalités critiques pour le fonctionnement de base du système sont marquées comme "Haute", tandis que les fonctionnalités additionnelles ou de gestion ont des niveaux d'importance variables (moyenne ou faible).

- **S'inscrire** : Permet à un nouvel utilisateur de créer un compte. Cela inclut la saisie d'informations personnelles telles que l'adresse e-mail, le nom d'utilisateur et le mot de passe.

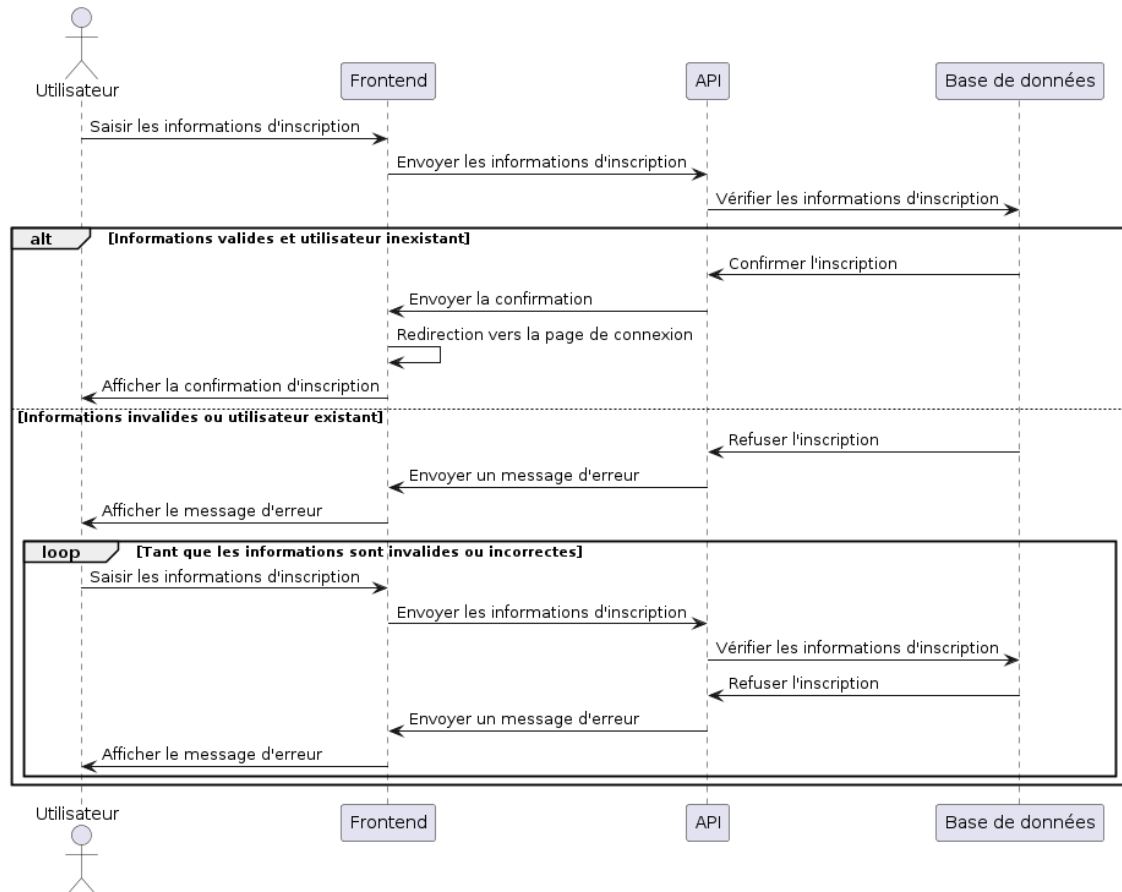


FIGURE 2.11 : Diagramme de séquence pour l'inscription

Dans ce diagramme de séquence, l'alt (alternative) montre les deux possibilités : si les identifiants sont corrects, l'utilisateur est authentifié ; sinon, un message d'erreur est affiché et l'utilisateur est invité à saisir à nouveau ses identifiants jusqu'à ce qu'ils soient corrects.

- **S'authentifier** : Permet à un utilisateur existant de se connecter à la plateforme en fournissant ses identifiants (nom d'utilisateur et mot de passe).

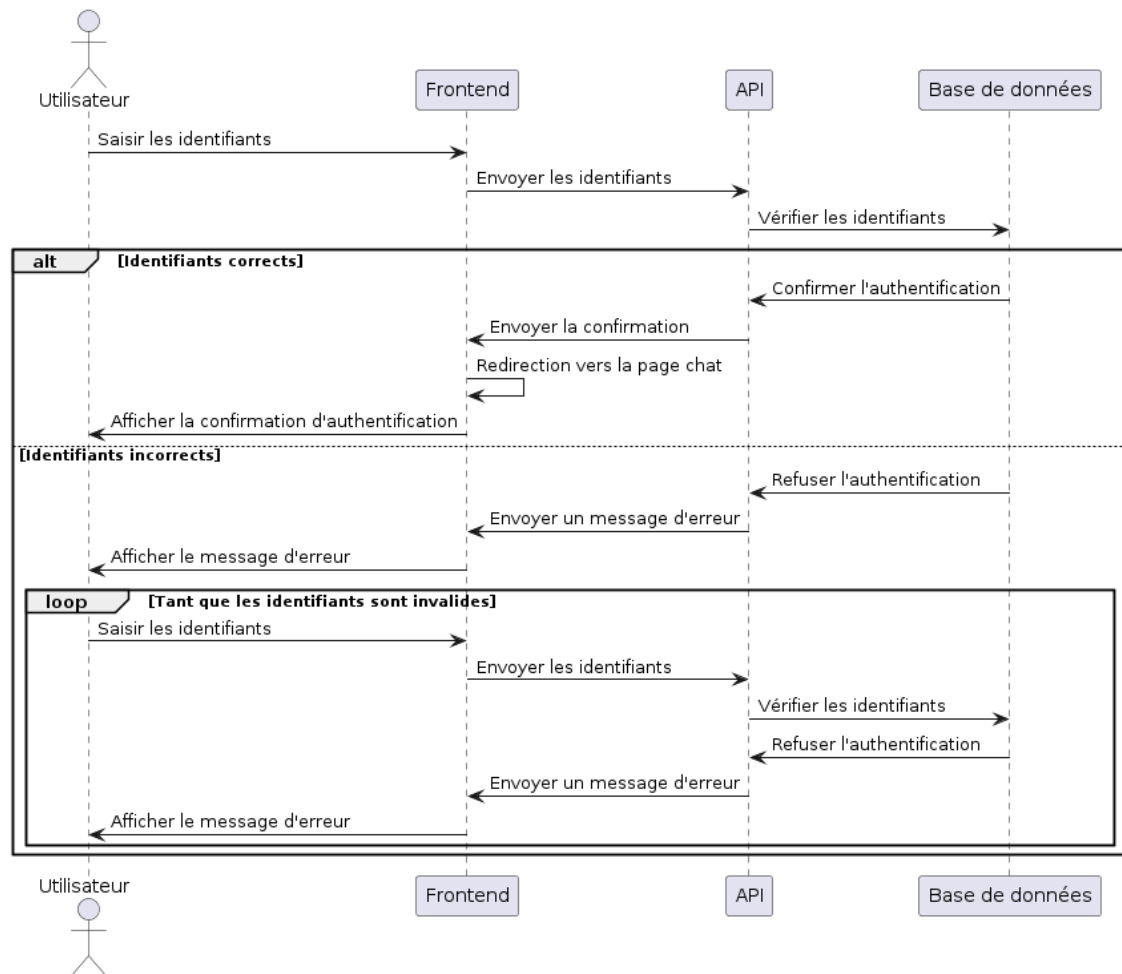


FIGURE 2.12 : Diagramme de séquence pour la connexion

Dans ce diagramme de séquence, l'alt (alternative) montre les deux possibilités : si les informations sont valides et que l'utilisateur n'existe pas, l'inscription est confirmée ; sinon, un message d'erreur est affiché et l'utilisateur est invité à saisir à nouveau ses informations jusqu'à ce qu'elles soient correctes.

- **Créer un nouveau chat** : Permet à l'utilisateur de créer une nouvelle session de chat. L'utilisateur peut nommer cette session et commencer à interagir avec le chatbot.

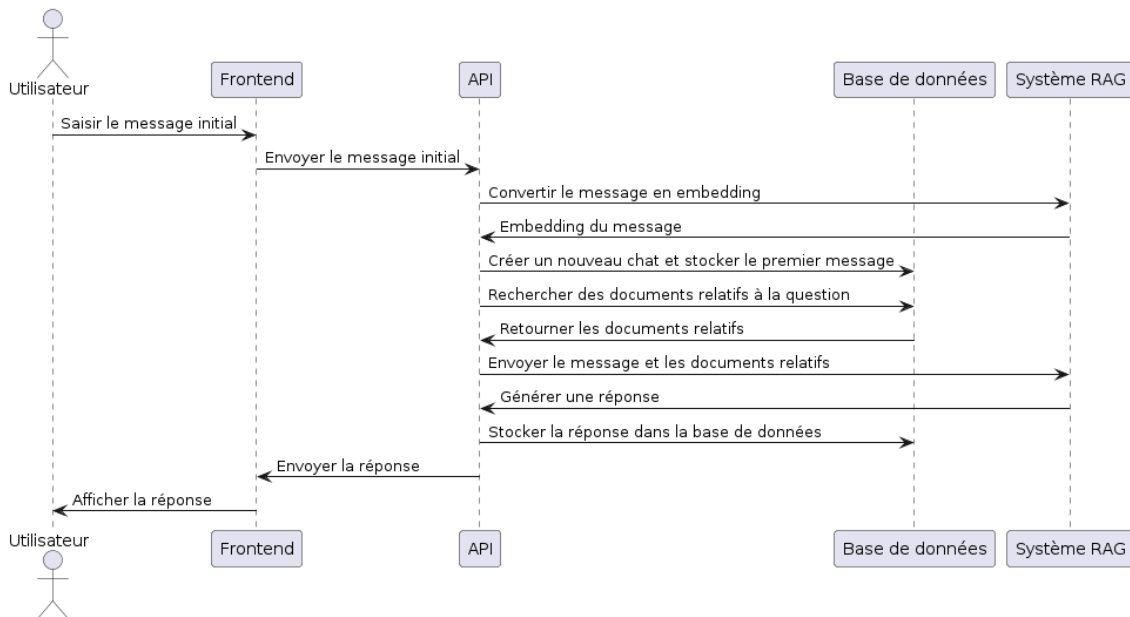


FIGURE 2.13 : Diagramme de séquence pour la création d'un chat

- **Envoyer un message** : Permet à l'utilisateur d'envoyer un message dans une session de chat active. Le message est traité par le système RAG pour générer une réponse appropriée.

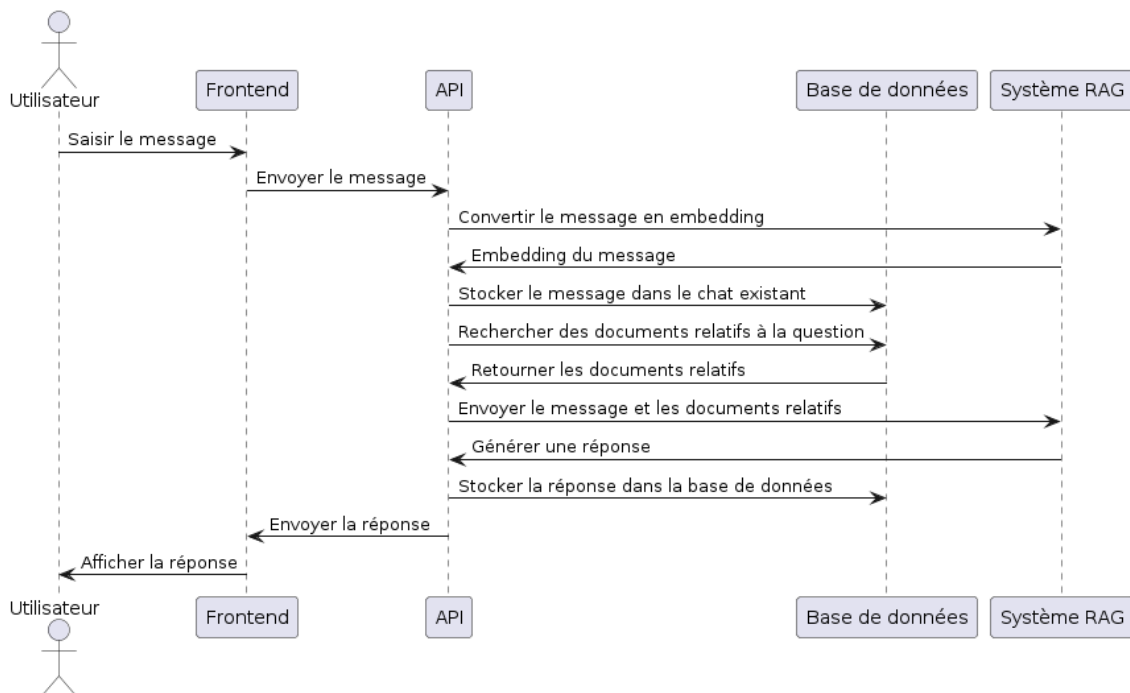


FIGURE 2.14 : Diagramme de séquence pour l'envoi d'un message

- **Renommer un chat** : Permet à l'utilisateur de modifier le nom d'une session de chat existante pour une meilleure organisation.

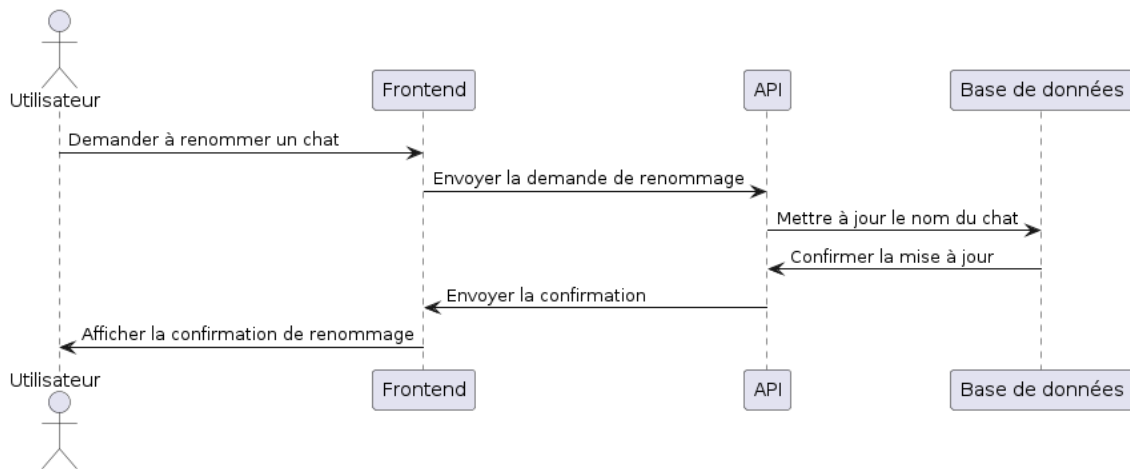


FIGURE 2.15 : Diagramme de séquence pour modifier du chat

- **Supprimer un chat** : Permet à l'utilisateur de supprimer une session de chat existante. Cette action est irréversible et supprimera toutes les données associées au chat.

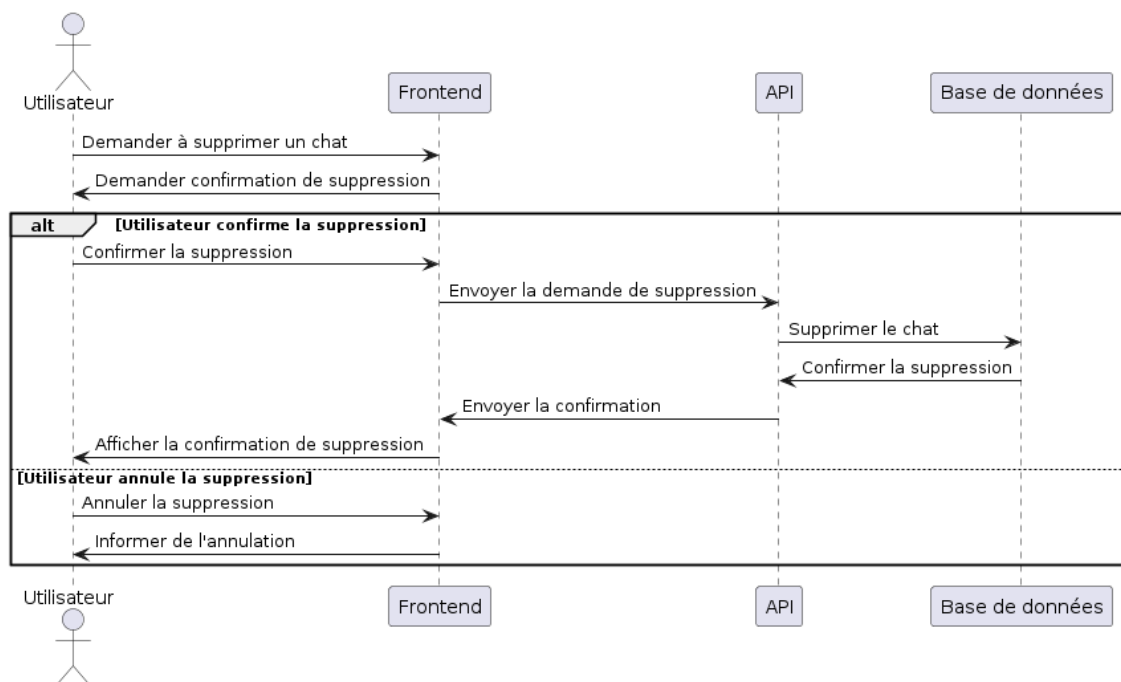


FIGURE 2.16 : Diagramme de séquence pour la suppression du chat

- **Consulter les chats** : Permet à l'utilisateur de voir la liste de toutes les sessions de chat créées. L'utilisateur peut sélectionner un chat pour le consulter ou pour effectuer d'autres actions comme renommer ou supprimer.

- **Choisir un chat proposé** : Permet à l'utilisateur de sélectionner un chat parmi une liste de chats proposés, basée sur des suggestions ou des chats précédemment sauvegardés.

2.4.2 Le backend - API

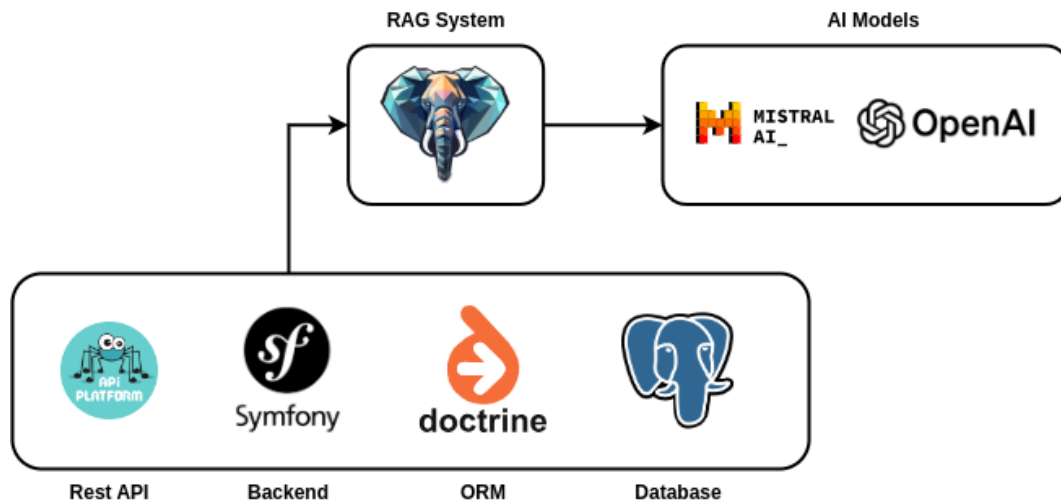


FIGURE 2.17 : Architecture du backend et services tiers

L'architecture backend de notre application repose sur plusieurs technologies clés qui travaillent ensemble pour fournir une expérience utilisateur fluide et performante

- **REST API** : API Platform ¹⁶ est un framework utilisé pour créer des [API](#) Representational State Transfer ([REST](#)) robustes et bien structurées. Elle facilite la communication entre le frontend et le backend en exposant des endpoints que le frontend peut appeler pour interagir avec le système.

Les requêtes Hypertext Transfert Protocol ([HTTP](#)) envoyées par le frontend sont reçues par API Platform, qui les traite et les dirige vers les contrôleurs appropriés dans le backend.

¹⁶ <https://api-platform.com/>

Chat		
GET	/api/chats	Retrieves the collection of Chat resources.
POST	/api/chats	Creates a Chat resource.
GET	/api/chats/{id}	Retrieves a Chat resource.
DELETE	/api/chats/{id}	Removes the Chat resource.
PATCH	/api/chats/{id}	Updates the Chat resource.

FIGURE 2.18 : Documentation de l'API générée par API Platform

- **Backend (Symfony)** : Symfony ¹⁷ est le framework [PHP](https://symfony.com/) utilisé pour développer le backend de l'application. Il fournit une structure solide pour construire des applications web robustes, en facilitant la gestion des requêtes, la logique métier, et les services.

Lorsque API Platform reçoit une requête, il appelle les contrôleurs Symfony qui contiennent la logique nécessaire pour traiter la requête. Symfony gère également les interactions avec le système RAG et la base de données.

- **ORM (Doctrine)** : Doctrine ¹⁸ est un Object-Relational Mapping ([ORM](https://www.doctrine-project.org/)) utilisé pour interagir avec la base de données de manière abstraite. Il permet de manipuler les données en utilisant des objets [PHP](#) plutôt que des requêtes [SQL](#) brutes.

Les contrôleurs Symfony utilisent Doctrine pour accéder à la base de données PostgreSQL. Doctrine traduit les opérations sur les objets [PHP](#) en requêtes [SQL](#) et les exécute sur la base de données.

- **Base de donnée (PostgreSQL)** : PostgreSQL est le système de gestion de base de données relationnelle utilisé pour stocker les données de l'application, y compris les utilisateurs, les messages, les chats, et les embeddings.

Doctrine interagit directement avec PostgreSQL pour effectuer des opérations de lecture et d'écriture. Les données sont stockées et récupérées selon les besoins de l'application.

¹⁷ <https://symfony.com/>

¹⁸ <https://www.doctrine-project.org/>

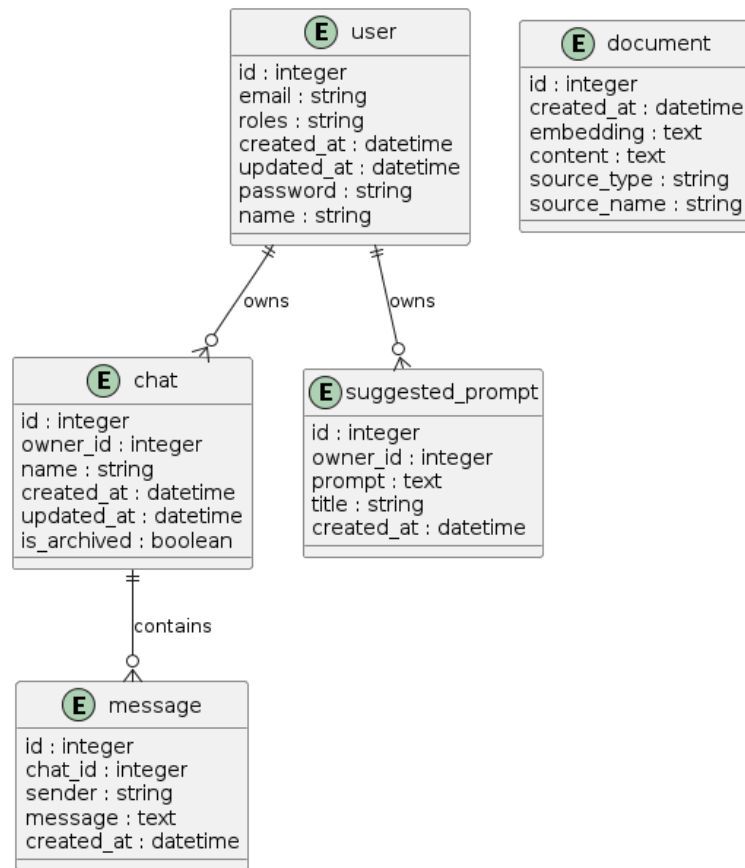


FIGURE 2.19 : Diagramme relationnel d'entité

- **RAG System (LLPhant)** : LLPhant ¹⁹ est un framework utilisé pour générer des réponses en utilisant des modèles d'IA. Il combine des techniques de récupération d'informations et de génération de texte pour fournir des réponses précises et contextuelles.

Lorsqu'une requête nécessite une réponse contextuelle, Symfony appelle le système **RAG**. Ce système utilise les embeddings stockés dans PostgreSQL pour trouver les documents pertinents, puis génère une réponse en s'appuyant sur des modèles d'IA (comme ceux de Mistral AI ²⁰ et OpenAI ²¹).

- **Les modèles d'IA** Ces modèles d'IA sont utilisés pour convertir les questions en embeddings et générer des réponses basées sur les documents récupérés.

Le système **RAG** envoie les questions aux modèles d'IA pour les convertir en embeddings. Il utilise également ces modèles pour générer des réponses contextuelles basées sur les informations récupérées de la base de données.

¹⁹ <https://llphant.io/>

²⁰ <https://console.mistral.ai/>

²¹ <https://openai.com/>

2.5 DÉPLOIEMENT ET MIS EN PRODUCTION

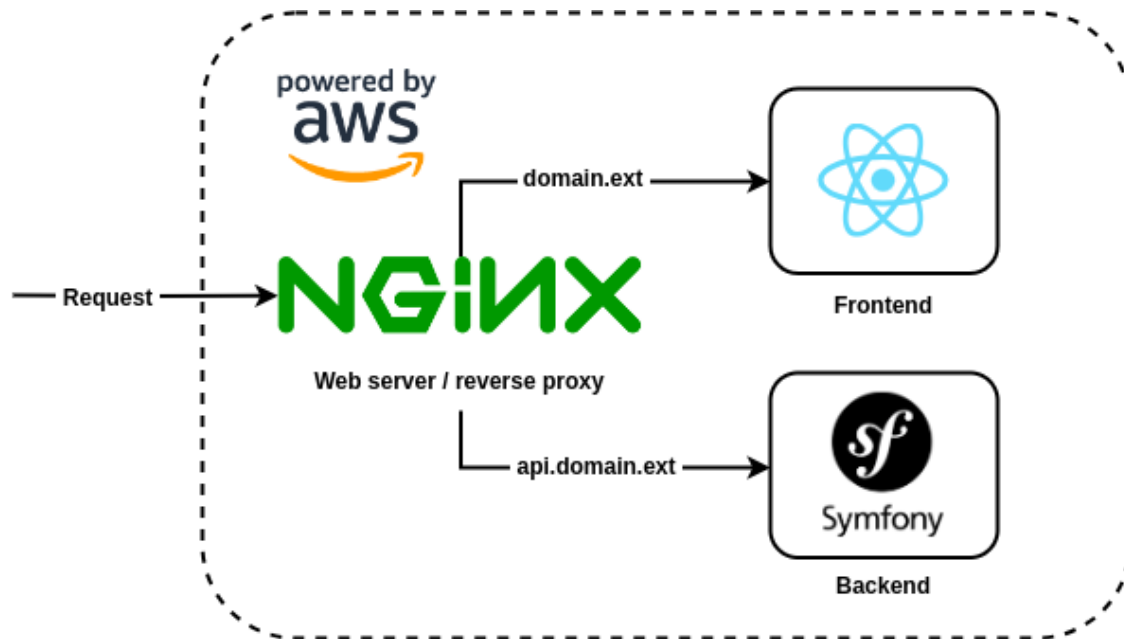


FIGURE 2.20 : Architecture de production

Toutes les composantes de l'application (NGINX ²², frontend, backend) sont déployées sur l'infrastructure AWS ²³, assurant scalabilité, fiabilité et disponibilité.

NGINX, sert de serveur web et de reverse proxy pour gérer les requêtes des utilisateurs. NGINX distribue les requêtes entre le frontend et le backend, selon le type de requête (client-side vs server-side).

2.5.1 Configuration du pare-feu

Une fois une instance (machine virtuelle) créée, nous pouvons y attacher une adresse Internet Protocol (IP) statique. Cette adresse IP statique garantit que l'adresse attribuée à l'instance ne changera pas après un redémarrage ou une ré-initialisation de celle-ci, assurant ainsi une accessibilité constante et fiable.

Pour sécuriser notre plateforme, nous mettons en place des mesures strictes de contrôle des accès. Nous restreignons l'ouverture des ports réseau uniquement à ceux nécessaires au fonctionnement de nos services. Les ports autorisés sont :

- **SSH (port 22)** : Utilisé pour les connexions sécurisées à distance, permettant aux administrateurs de gérer l'instance de manière sécurisée.

²² <https://nginx.org/>

²³ <https://aws.amazon.com/>

- **HTTP (port 80)** : Utilisé pour le trafic web non sécurisé, permettant aux utilisateurs d'accéder à l'application web.
- **HTTPS (port 443)** : Utilisé pour le trafic web sécurisé, garantissant que les données transmises entre l'utilisateur et le serveur sont cryptées.
- **PostgreSQL (port 5432)** : Utilisé pour les connexions à la base de données PostgreSQL, permettant aux services backend de lire et d'écrire des données de manière sécurisée.

En limitant l'accès à ces ports spécifiques, nous minimisons les risques d'attaques potentielles en réduisant la surface d'attaque disponible. Cela contribue à protéger notre infrastructure et à garantir que seuls les services nécessaires sont accessibles, tout en empêchant les accès non autorisés aux autres ports. Cette stratégie de sécurité est cruciale pour maintenir l'intégrité et la confidentialité de notre plateforme.

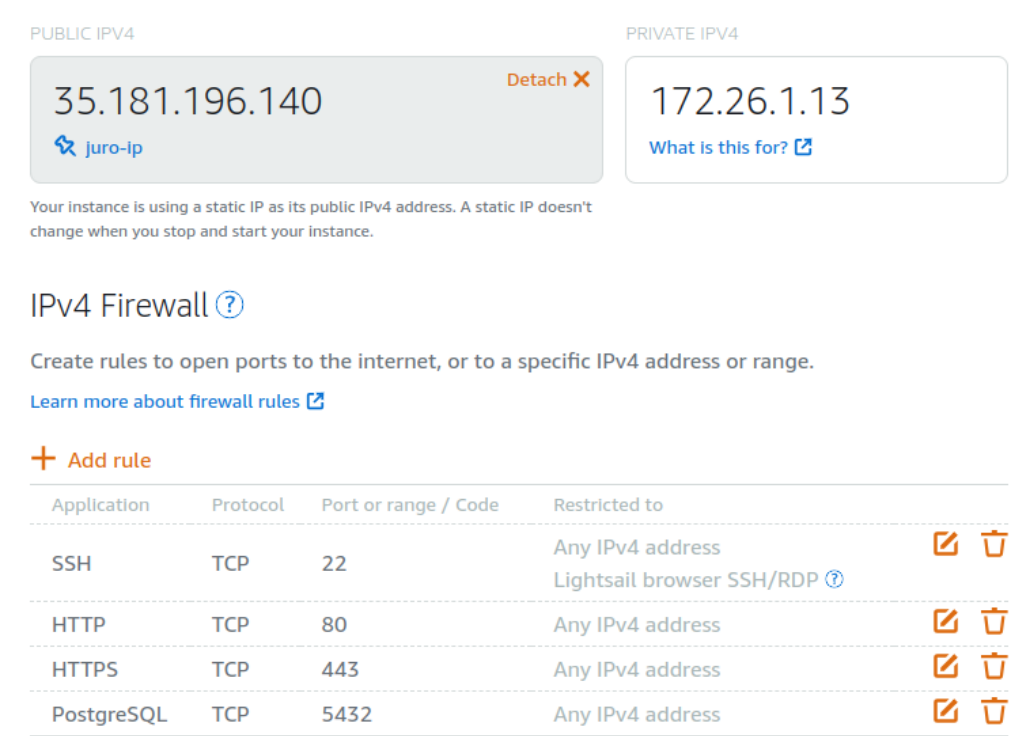


FIGURE 2.21 : Configuration Firewall

2.5.2 Configuration du nom de domaine, DNS

Pour la configuration [DNS](#), nous ajoutons un enregistrement de type A pour lier le nom de domaine "juro.life" à l'adresse [IP](#) statique de notre instance. Cet enregistrement de type A assure que toute requête dirigée vers "juro.life" est résolue en direction de l'adresse [IP](#) associée, permettant aux utilisateurs d'accéder à notre application de manière fiable.

Afin de gérer les sous-domaines de manière efficace, nous utilisons la même adresse IP statique. Cela signifie que nous créons des enregistrements DNS supplémentaires pour chaque sous-domaine requis, tels que "api.juro.life" et "www.juro.life", et les faisons pointer vers la même adresse IP statique. Cette approche simplifie la gestion DNS et garantit que toutes les sous-domaines de notre domaine principal sont correctement résolus vers notre instance.

DNS records

Each record in a DNS zone defines how you want to route internet traffic for your domain. For example, you can add DNS records that route traffic to your Lightsail resources, another domain, or a mail server.

[Learn more about editing DNS records](#)

[+ Add record](#)

A RECORDS





Record name	Route traffic to	
juro.life	35.181.196.140	 
*.juro.life	35.181.196.140	 

FIGURE 2.22 : Configuration Domain Name System (DNS)

2.5.3 Configuration du reverse proxy

Maintenant que nous avons une adresse IP statique et un nom de domaine pointant vers cette adresse, nous pouvons configurer notre serveur pour rediriger les requêtes HTTP associées au domaine ou aux sous-domaines vers les applications appropriées qui tournent sur le serveur. De plus, nous mettons en place une redirection Hypertext Transfer Protocol Secure (HTTPS) pour assurer que toutes les communications entre les utilisateurs et notre application sont sécurisées.

```
server {
    server_name api.juro.life;
    root /var/www/html/juro-api/public;
    index index.php index.html index.htm index.nginx-debian.html;

    location / {
        root /var/www/html/juro-api;
        try_files /public/$uri /assets/$uri /index.php?$query_string;
    }
}
```

Listing 13 : Exemple de configuration du serveur web

Pour assurer la sécurité des communications, nous redirigeons automatiquement les requêtes HTTP vers HTTPS. Voici comment configurer cette redirection dans NGINX

```

server {
    if ($host = api.juro.life) {
        return 301 https://$host$request_uri;
    }

    listen 80;

    server_name api.juro.life;
    return 404;
}

```

Listing 14 : Exemple de redirection automatique vers HTTPS

En configurant NGINX de cette manière, nous assurons que toutes les requêtes [HTTP](#) vers notre domaine et ses sous-domaines sont correctement redirigées vers les applications respectives. De plus, en forçant les redirections vers [HTTPS](#), nous garantissons la sécurité des communications entre les utilisateurs et notre serveur. Cette configuration robuste permet de gérer efficacement les accès à notre plateforme tout en maintenant un haut niveau de sécurité.

2.6 RÉSUMÉ DU CHAPITRE

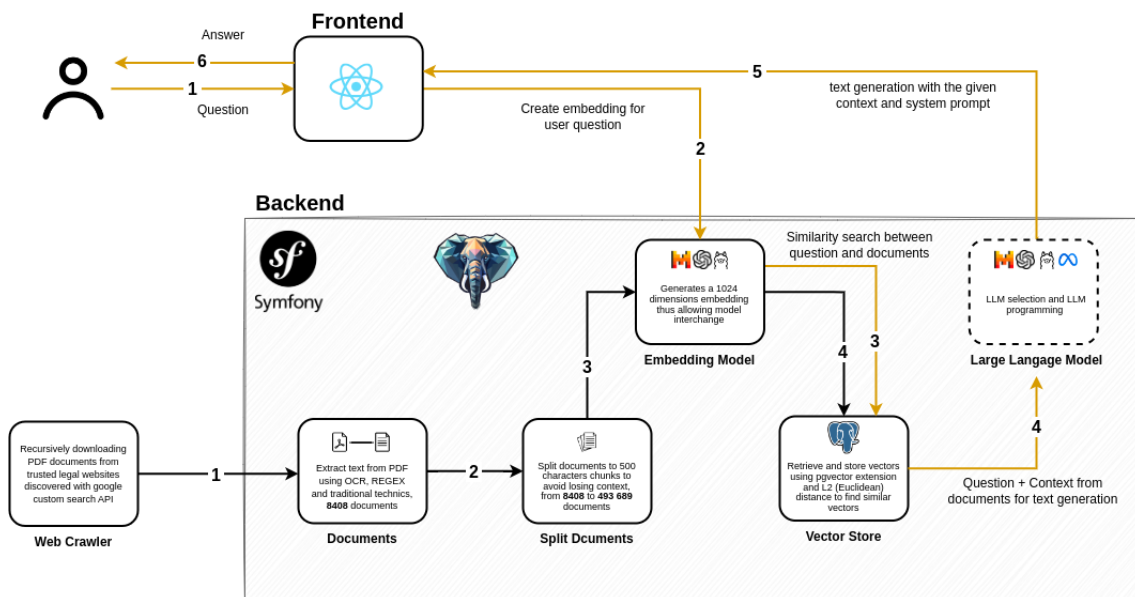


FIGURE 2.23 : Achitecture

Ce chapitre a fourni une vue d'ensemble complète et détaillée des différentes étapes et composantes impliquées dans la conception, le développement et le déploiement de notre chatbot juridique, mettant en évidence les choix technologiques et les stratégies d'implémentation adoptées pour créer un système performant et fiable.

ÉVALUATION ET FEEDBACK HUMAIN

Bien que les LLMs soient capables de restituer l'information avec une précision impressionnante, ils ne sont pas exempts d'erreurs et peuvent parfois produire des résultats erronés sous forme d'« hallucinations » [6, 74]. Ce terme désigne les instances où les modèles génèrent des contenus qui semblent plausibles mais qui ne sont pas fondés sur des faits réels ou vérifiables. Ces hallucinations peuvent inclure la fabrication de détails fictifs, la distorsion des informations existantes, ou l'émission de conclusions inexactes. Ce phénomène est particulièrement problématique dans le domaine juridique où la fiabilité et la précision des informations sont cruciales.

Étant donné notre objectif de vulgariser le Droit congolais de manière accessible et précise, il est impératif d'intégrer une couche d'évaluation humaine. Cette évaluation sera assurée par des professionnels du droit congolais, qui apporteront leur expertise pour vérifier et corriger les outputs du modèle. Leur rôle sera d'assurer que les informations générées par le chatbot soient non seulement correctes mais aussi pertinentes et utiles pour les utilisateurs finaux. Cette démarche permettra de pallier les limitations des LLMs et de garantir que les informations diffusées respectent les normes juridiques et éthiques, renforçant ainsi la fiabilité et la crédibilité de notre projet.

L'intégration de feedback humain dans le processus d'évaluation contribue également à un cycle d'amélioration continue du modèle, où les corrections et les insights fournis par les experts peuvent être utilisés pour affiner et ajuster les algorithmes sous-jacents. Cette collaboration entre IA et expertise humaine est essentielle pour créer un outil robuste et fiable, capable de servir efficacement les besoins d'information juridique de la communauté.

3.1 CRITÈRES ET MÉTHODES D'ÉVALUATIONS

Dans une première phase d'évaluation, nous mettrons à l'épreuve les modèles existants en utilisant le test de magistrature congolais de 2022. Pour ce faire, nous appliquerons des critères d'évaluation rigoureux pour juger de la qualité des réponses générées par ces modèles. Ces critères sont les suivants :

1. Pertinence : Nous évaluerons si les réponses fournies sont directement liées à la question posée et si elles traitent les aspects pertinents du cas juridique présenté. Il est crucial que chaque réponse aborde les éléments clés de la question pour être jugée pertinente.

2. **Clarté et Structure** : La formulation des réponses doit être claire et compréhensible. La structure des réponses devra suivre une logique cohérente, facilitant ainsi la compréhension et la suivabilité des arguments.
3. **Analyse Juridique** : Il est essentiel que les réponses démontrent une compréhension approfondie des principes juridiques impliqués. Nous attendons des analyses juridiques précises et adaptées au contexte du cas traité.
4. **Raisonnement et Logique** : Les réponses doivent reposer sur un raisonnement solide et logique. Les arguments doivent être non seulement convaincants mais également bien développés pour soutenir les conclusions.
5. **Originalité et Créativité** : Nous apprécierons les réponses qui offrent des perspectives originales ou des solutions créatives aux problèmes juridiques posés, ajoutant ainsi de la valeur à la simple restitution des faits ou des lois.

Pour mener cette évaluation de manière objective, les réponses générées par les modèles seront collectées via un formulaire Google Form ¹. De manière cruciale, les juristes chargés d'évaluer ces réponses ne sauront pas quel modèle a généré quelles réponses. Cette anonymisation est conçue pour prévenir tout biais dans l'évaluation, garantissant que les modèles sont jugés strictement sur la base de leur performance et non en fonction de leur notoriété présumée.

LE TEST DE MAGISTRATURE 2022

Question	Catégorie
Peut-on constituer un prévenu gardien d'un objet saisi ? Dans la négative, donnez nous trois raisons.	procédure pénale
Les expressions, auteur présumé de l'infraction, l'inculpé, prévenu et condamné, traduisent quelles étapes des instances judiciaires.	procédure pénale
Quelle est la différence entre l'amnistie et la grâce du point de l'organe de décision ?	procédure pénale
Le ministère public peut-il requérir devant le juge le classement sans suite d'un dossier fixé devant le tribunal ? Dans la négative, donnez deux raisons.	procédure pénale
Le ministère public peut-il aussi introduire la procédure de suspicion légitime du tribunal dont il est membre de composition, si oui à quel titre.	procédure pénale
La nationalité congolaise peut-elle être détenue concurremment avec une autre nationalité par un sujet congolais vivant en RDC ? Justifiez votre réponse.	droit civil des personnes
La dissolution du mariage par les autorités coutumières ou familiales peut-elle produire d'effets ? Justifiez votre réponse.	droit civil des personnes

¹ <https://workspace.google.com/intl/en/products/forms>

Quelles sont les trois formes de testament consacré par la législation congolaise ? Explicitez-les.	droit civil des personnes
Monsieur FULANI marié à Madame SONGOLO décède en laissant derrière lui deux enfants nés avant le mariage, quatre enfants pendant le mariage, trois enfants hors mariage, un enfant adoptif, deux frères et trois sœurs. Le de cujus n'a laissé qu'un seul bien de valeur en l'occurrence un immeuble acheté auprès de l'ex ONL. L'aîné des enfants estime que le bien doit lui revenir. Les enfants nés pendant le mariage soutiennent qu'ils sont des enfants légitimes et peuvent seuls prétendre à l'héritage du de cujus. De leur côté, les frères et sœurs du défunt pensent que l'immeuble laissé par leur cadet ne doit revenir en priorité qu'à la famille. L'épouse du de cujus formule aussi les mêmes prétentions. S'agissant d'un seul bien de valeur, quelles solutions préconisez-vous à ces différentes prétentions ?	droit civil des personnes
Comment qualifie-t-on dans leur ordre progressif de degré de criminalité, les deux extrêmes de la tentative punissable ?	droit pénal général
Un condamné incarcéré à 12 heures du matin, pour subir un jour d'emprisonnement, constate dans sa fiche de libération qu'on l'a fait sortir le lendemain du jour d'incarcération à 14 heures, est-il en droit de se plaindre pour détention illégale ?	droit pénal général
En République Démocratique du Congo, la peine de fouet qui ne pouvait être infligée que par les juridictions indigènes a été supprimée par a- L'accession du Congo à l'indépendance le 30 juin 1960 b- L'accord global et inclusif du Sun City, après l'A.F.D.L. c- Le décret du 18 décembre 1951 avant l'indépendance	droit pénal général
Le calcul de jour de détention d'une personne incarcérée n commence-t-il : a- Le jour de la condamnation par le tribunal ? b-Le jour où la condamnation est coulée en force de chose jugée ? c- Le jour où sa détention est confirmée par la chambre de conseil ? d-Le jour où elle est placée sous mandat d'arrêt provisoire ? e-Le jour où elle a été privée de sa liberté ?	droit pénal général
En quoi l'amende judiciaire est différente de l'amende transactionnelle ? donnez au moins 4 points de différence.	droit pénal général

TABLE 12 : Questions du test de magistrature Congolais 2022

3.2 ÉVALUATION DES MODÈLES EXISTANTS

Nous débuterons notre évaluation en attribuant à chaque modèle un identifiant anonyme afin de garantir l'impartialité de l'évaluation. Les modèles seront désignés comme suit :

```
models = {
    'Modèle A': 'gpt3.5-turbo',
    'Modèle B': 'gemini-pro',
    'Modèle C': 'llama2',
    'Modèle D': 'mistral',
    'Modèle E': 'vicuna'
}
```

Après cette étape d'association, nous présenterons une série de questions issues du test de magistrature congolais de 2022 à chacun de ces modèles. Les réponses fournies par chaque modèle seront ensuite collectées et enregistrées au format [CSV](#) pour faciliter l'analyse et la comparaison ultérieures. voici un exemple pour le modèle gemini-pro [55] de Google :

```
import os
import google.generativeai as genai
import pandas as pd

genai.configure(api_key=os.getenv('GOOGLE_KEY'))
model = genai.GenerativeModel('gemini-pro')
questions = pd.read_csv('_magistrature.csv')

def generate_content(x):
    print(f"Answering question: {x['question']}")
    prompt = f"""
        Dans le contexte du Droit Congolais (RDC) précisément {x['category']},
        répondez à la question suivante : {x['question']}
    """

    response = model.generate_content(prompt)
    return response.text

questions['model'] = 'gemini-pro'
questions['answer'] = questions.apply(lambda x: generate_content(x), axis=1)
questions.to_csv('./data/answers-gemini-pro.csv', index=False)
```

Listing 15 : Évaluation du modèle Gemini Pro sur le test de magistrature 2022.

Chaque ensemble de réponses correspondant à un modèle sera également associé à une lettre, afin de préserver l'anonymat tout au long du processus d'évaluation. Cette méthode nous permet de suivre précisément quelle réponse appartient à quel modèle sans révéler cette information aux évaluateurs, assurant ainsi que les évaluations restent

centrées uniquement sur la qualité des réponses par rapport aux critères établis, sans influence extérieure.

3.2.1 Évaluation qualitative

21/02/2024 08:49:46	4 : Bon	4 : Bon	3 : Moyen	1 : Très faible	2 : Faible	1 : Très faible	3 : Moyen	2 : Faible
21/02/2024 16:35:49	4 : Bon	4 : Bon	3 : Moyen	4 : Bon	4 : Bon	3 : Moyen	4 : Bon	3 : Moyen
21/02/2024 19:09:23	3 : Moyen	5 : Excellent	4 : Bon	4 : Bon	2 : Faible	4 : Bon	3 : Moyen	2 : Faible
22/02/2024 14:21:20	3 : Moyen	3 : Moyen	4 : Bon	2 : Faible	2 : Faible	1 : Très faible	3 : Moyen	1 : Très faible
27/02/2024 08:17:56	5 : Excellent	5 : Excellent	5 : Excellent	4 : Bon	5 : Excellent	4 : Bon	5 : Excellent	5 : Excellent
16/03/2024 21:36:18	3 : Moyen	4 : Bon	5 : Excellent	2 : Faible	3 : Moyen	2 : Faible	2 : Faible	2 : Faible
17/03/2024 07:55:09	4 : Bon	5 : Excellent	5 : Excellent	4 : Bon	4 : Bon	1 : Très faible	3 : Moyen	4 : Bon
20/03/2024 19:18:20	5 : Excellent	5 : Excellent	5 : Excellent	5 : Excellent	5 : Excellent	3 : Moyen	5 : Excellent	5 : Excellent
17/04/2024 17:19:24	4 : Bon	5 : Excellent	4 : Bon	3 : Moyen	4 : Bon	1 : Très faible	4 : Bon	2 : Faible
17/04/2024 17:21:08	5 : Excellent	5 : Excellent	5 : Excellent	4 : Bon	5 : Excellent	1 : Très faible	5 : Excellent	1 : Très faible
17/04/2024 17:51:03	4 : Bon	5 : Excellent	4 : Bon	1 : Très faible	4 : Bon	2 : Faible	4 : Bon	3 : Moyen
17/04/2024 19:45:14	5 : Excellent	5 : Excellent	5 : Excellent	5 : Excellent	5 : Excellent	3 : Moyen	5 : Excellent	4 : Bon
18/04/2024 18:58:31	4 : Bon	5 : Excellent	5 : Excellent	4 : Bon	5 : Excellent	1 : Très faible	4 : Bon	4 : Bon
21/04/2024 21:05:08	1 : Très faible	5 : Excellent	3 : Moyen	5 : Excellent	5 : Excellent	1 : Très faible	1 : Très faible	1 : Très faible
22/04/2024 22:10:44	3 : Moyen	4 : Bon	2 : Faible	4 : Bon	3 : Moyen	1 : Très faible	4 : Bon	3 : Moyen
22/06/2024 14:03:46	1 : Très faible	4 : Bon	4 : Bon	1 : Très faible	1 : Très faible	1 : Très faible	2 : Faible	2 : Faible

FIGURE 3.1 : Extrait des évaluations reçues via Google Form

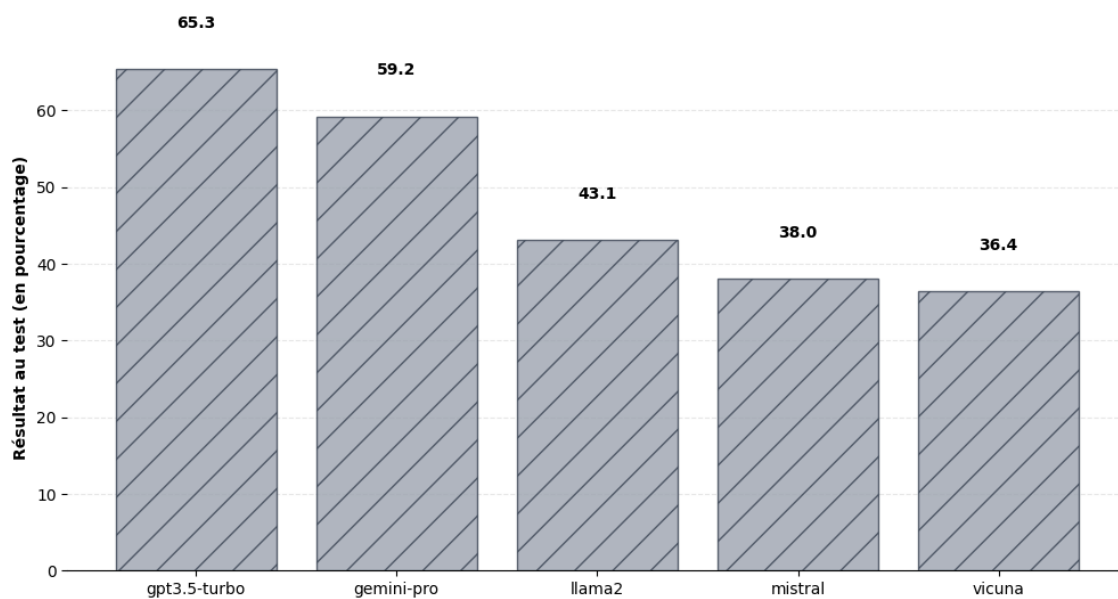


FIGURE 3.2 : Résultats après évaluation des différents modèles ,en pourcentage. (voir Code [21](#))

3.2.2 Évaluation quantitative

Données de l'évaluation obtenues avec l'outil artificialanalysis.ai

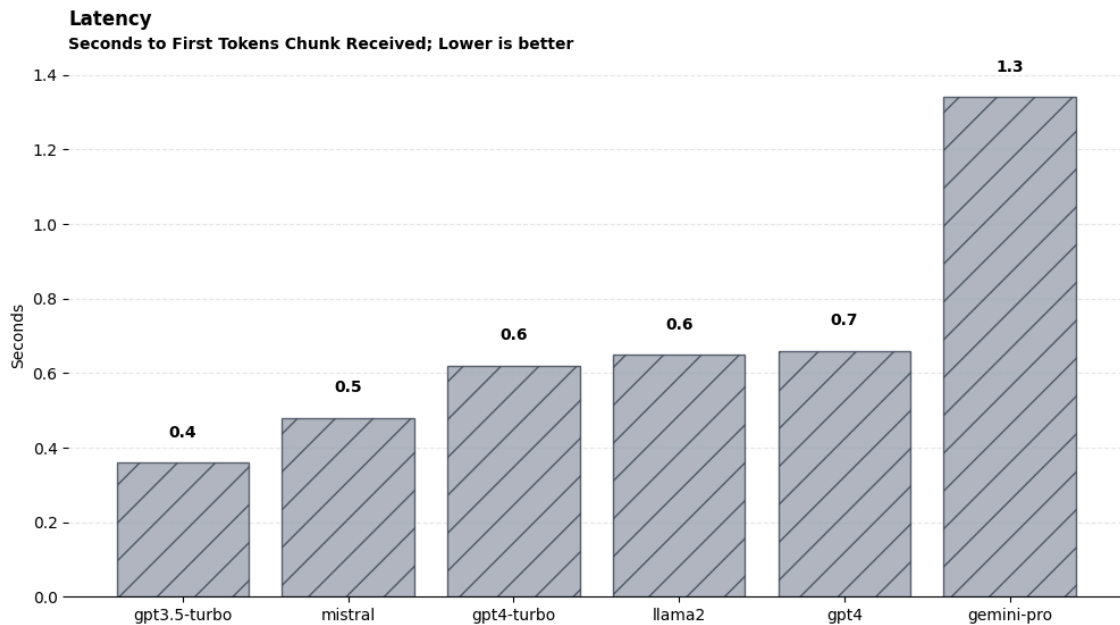


FIGURE 3.3 : Délai de réception du premier token, en secondes, après l'envoi de la demande d'API.

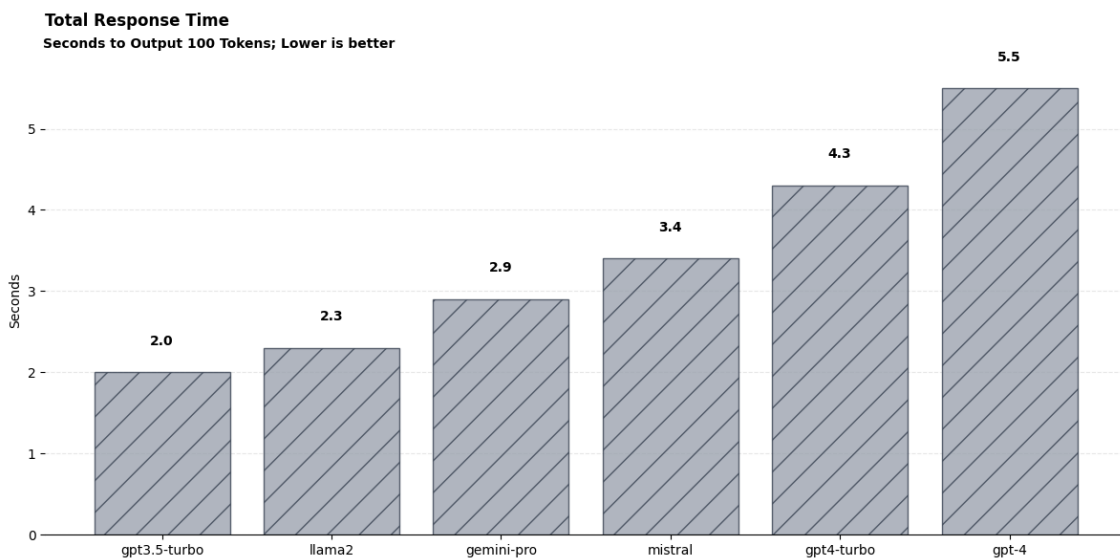


FIGURE 3.4 : Temps nécessaire pour recevoir une réponse de 100 tokens. Estimation basée sur la latence (temps de réception du premier morceau) et la vitesse de sortie (nombre de tokens par seconde).

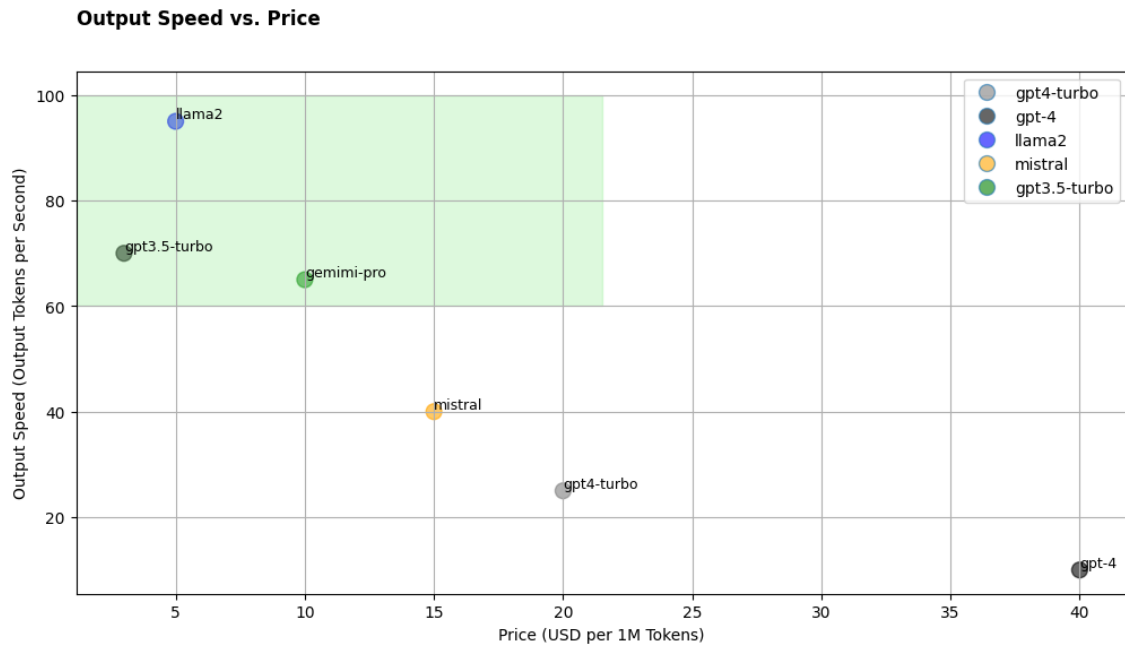


FIGURE 3.5 : Prix par token, représenté en USD par million de jetons. Le prix est un mélange des prix des tokens d'entrée et de sortie (ratio 3 1).

L'évaluation quantitative révèle que les modèles d'entreprise, avec leurs ressources dédiées, garantissent une haute disponibilité. Cependant, au-delà des capacités intrinsèques de chaque modèle, l'infrastructure sous-jacente joue un rôle crucial dans les performances globales.

Une infrastructure optimisée permet non seulement de maximiser l'efficacité des modèles, mais aussi d'assurer une réactivité et une fiabilité supérieures. La performance n'est donc pas uniquement liée à la sophistication du modèle, mais aussi à la robustesse et à l'efficacité de l'environnement dans lequel il opère. L'optimisation des serveurs, la gestion efficace des ressources et la minimisation des latences réseau sont autant de facteurs déterminants pour atteindre des performances de pointe.

3.3 ÉVALUATION DE NOTRE MODÈLE

Avant de procéder à une évaluation humaine, il est crucial de souligner la capacité de notre modèle à citer les sources utilisées pour générer ses réponses.

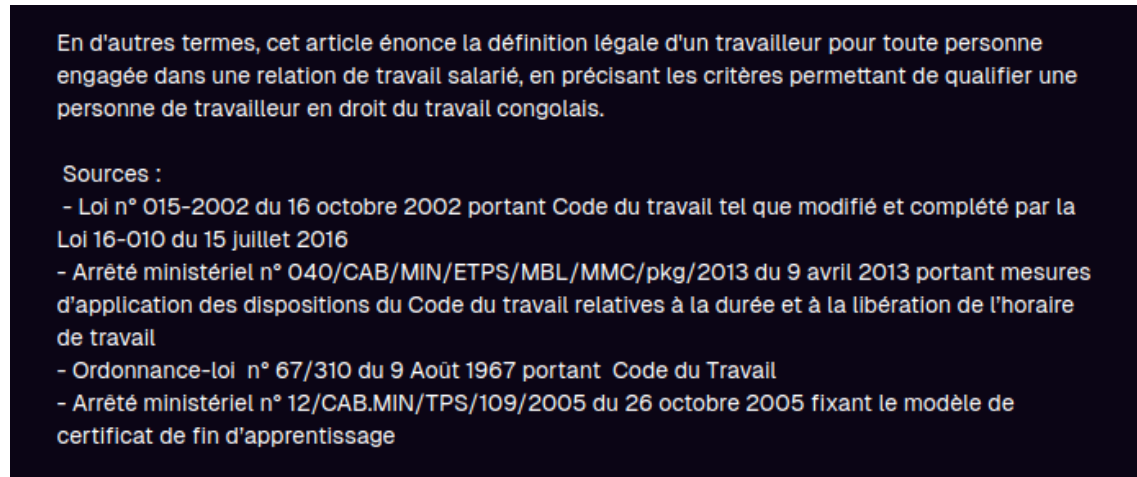


FIGURE 3.6 : Juro : Réponse avec citation

En citant les sources, notre modèle permet aux utilisateurs de vérifier l'origine des informations fournies, augmentant ainsi la confiance dans la validité et l'authenticité des réponses. Aussi la capacité à référencer les sources permet à notre modèle de mieux contextualiser les informations. Les utilisateurs peuvent comprendre non seulement le contenu de la réponse, mais aussi le contexte dans lequel cette information a été obtenue, enrichissant ainsi leur compréhension globale.

En fournissant des références, notre modèle aide les utilisateurs à approfondir leur recherche sur des sujets spécifiques. Ils peuvent consulter les sources originales pour obtenir des informations supplémentaires ou une perspective plus détaillée.

3.3.1 Évaluation qualitative

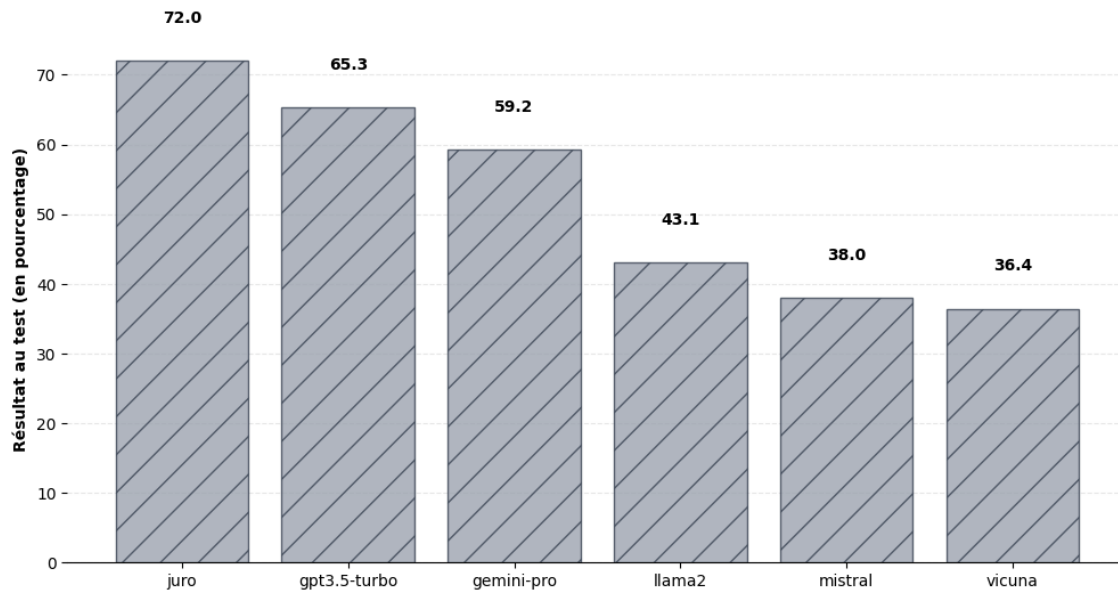


FIGURE 3.7 : Résultats après évaluation de Juro par rapport aux modèles existants, en pourcentage.

Notre modèle Juro a également été évalué par des juristes, en utilisant le même procédé et la même méthodologie que pour les autres modèles. Cette évaluation a permis de mesurer la performance de Juro dans un contexte spécifique et pertinent, celui du droit congolais.

Les résultats finaux montrent que Juro surpasse significativement les autres modèles testés. Avec un score de 72%, Juro se distingue par sa capacité à comprendre et à répondre aux questions juridiques congolaises de manière plus précise et plus pertinente. En comparaison, les autres modèles tels que GPT-3.5 Turbo (65.3%), Gemini Pro (59.2%), Llama2 (43.1%), Mistral (38.0%), et Vicuna (36.4%) ont affiché des performances inférieures, soulignant ainsi l'avantage compétitif de Juro dans ce domaine.

3.3.2 Évaluation quantitative

Pour évaluer quantitativement les performances de notre modèle, nous l'avons déployé sur une infrastructure cloud AWS (voir Section 2.5). Plus précisément, notre modèle a été hébergé sur une machine virtuelle avec les caractéristiques suivantes **2 GB RAM, 2 vCPUs, 60 GB SSD**. Cette configuration de base permet de mesurer la performance brute de notre modèle dans un environnement réaliste et accessible.

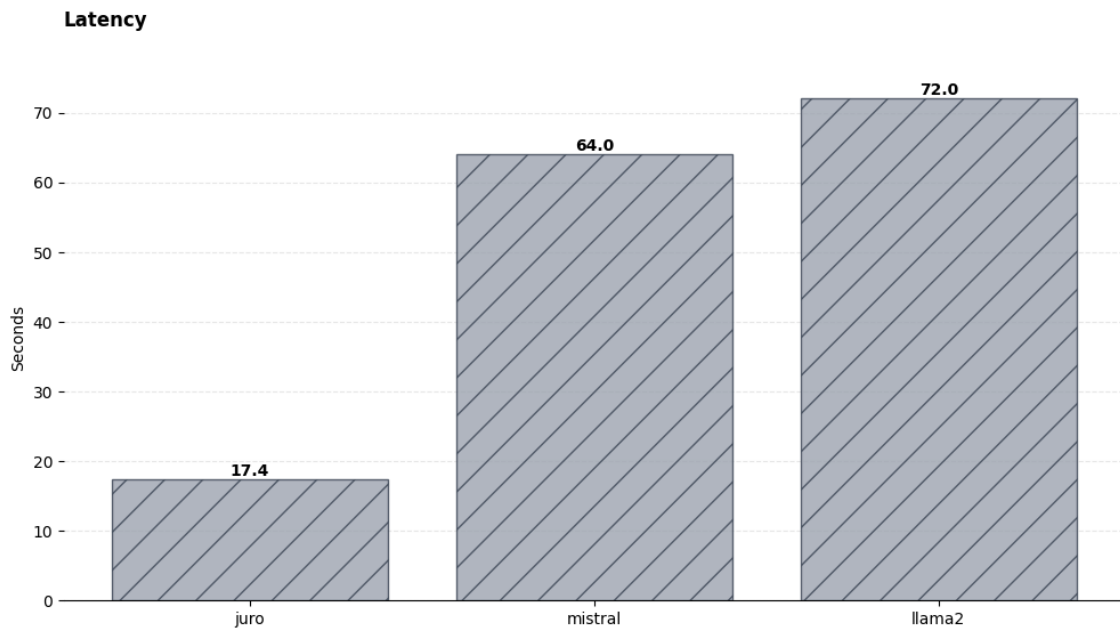


FIGURE 3.8 : Temps moyen nécessaire pour recevoir une réponse.

Pour notre modèle, nous nous appuyons sur les API de Mistral et OpenAI, ainsi que sur Ollama pour les modèles open source, notamment Mistral-7b et Llama-7b. L'inférence de ces modèles open source se fait sur notre propre serveur, qui gère également la récupération des documents nécessaires. Étant donné les spécifications techniques de notre serveur, le temps de réponse est naturellement plus lent, que ce soit pour les questions simples de droit, comme la recherche de lois, ou pour les questions doctrinales qui nécessitent un raisonnement approfondi. Cette lenteur inhérente à notre infrastructure impacte la rapidité de traitement et peut représenter un défi lors de l'interrogation de notre modèle pour des analyses juridiques complexes.

3.4 RÉSULTATS ET PERSPECTIVES

Les évaluations qualitatives et quantitatives menées sur notre modèle montrent des résultats prometteurs mais également des pistes d'amélioration. Les principaux résultats obtenus sont :

1. Les réponses générées par notre modèle se sont révélées globalement pertinentes et précises dans le contexte juridique. Cependant, certaines réponses nécessitent une compréhension plus approfondie des nuances juridiques et contextuelles.
2. Les évaluations qualitatives via Google Form ont montré que les utilisateurs ont trouvé les réponses utiles mais ont parfois relevé des incohérences mineures.
3. Le délai de réception du premier token après l'envoi de la demande d'API a été mesuré en secondes, révélant une latence plus élevée par rapport aux modèles d'entreprise. Cette différence est en grande partie due à notre infrastructure serveur moins optimisée.
4. Le temps nécessaire pour recevoir une réponse de 100 tokens a également été mesuré, avec des résultats indiquant que la vitesse de sortie (nombre de tokens par seconde) pourrait être améliorée pour une meilleure réactivité.

Les résultats obtenus ouvrent plusieurs voies d'amélioration et de développement futur :

1. L'optimisation de l'infrastructure serveur pourrait réduire la latence et améliorer la réactivité globale du modèle. L'intégration de serveurs plus puissants et de technologies de mise à l'échelle automatique pourrait être envisagée.
2. L'intégration de données spécifiques au domaine juridique et l'enrichissement continu du dataset utilisé pourraient améliorer la précision et la pertinence des réponses. Il est crucial de maintenir un dataset à jour et diversifié pour couvrir une large gamme de cas juridiques.
3. La mise en place de fonctionnalités avancées telles que la compréhension contextuelle améliorée, la gestion des biais dans les données et la capacité à fournir des explications détaillées des réponses pourrait renforcer la confiance des utilisateurs et améliorer l'utilité du modèle.
4. Encourager les collaborations entre experts en droit, ingénieurs en intelligence artificielle et autres parties prenantes est essentiel pour développer des solutions véritablement innovantes et adaptées aux besoins réels du secteur juridique. Les ateliers, les conférences et les projets communs pourraient favoriser l'échange de connaissances et de bonnes pratiques.

En définitive, bien que notre modèle ait montré des résultats prometteurs, il reste encore beaucoup de travail à accomplir pour améliorer ses performances et répondre pleinement aux besoins du secteur juridique. Les perspectives d'avenir sont nombreuses et offrent de nombreuses opportunités pour exploiter le potentiel des technologies d'intelligence artificielle dans le domaine du droit.

CONCLUSION

En rétrospective, ce travail a permis d'explorer en profondeur l'application des modèles d'intelligence artificielle dans le domaine du droit, avec un accent particulier sur l'utilisation de systèmes Retrieval-Augmented Generation (RAG). À travers une méthodologie rigoureuse incluant la collecte, le pré-traitement et la modélisation des données, nous avons démontré l'efficacité et les limites des modèles existants tout en proposant des améliorations significatives via notre modèle développé.

En réfléchissant à l'avenir, plusieurs pistes de développement et de recherche apparaissent prometteuses. L'amélioration continue des modèles de langage, notamment par l'intégration de données spécifiques au domaine juridique, pourrait considérablement enrichir le contexte et la précision des réponses générées. En outre, la combinaison des approches de machine learning traditionnelles avec les techniques de deep learning pourrait donner naissance à des systèmes hybrides encore plus performants et capables de mieux comprendre les nuances juridiques.

Il est également crucial de développer des mécanismes pour identifier et atténuer les biais présents dans les données et les algorithmes, assurant ainsi des systèmes d'IA plus équitables et transparents. L'extension des cas d'utilisation des modèles RAG dans d'autres domaines du droit, comme le droit international, pourrait offrir de nouvelles perspectives et applications innovantes.

L'amélioration de l'interface utilisateur des applications basées sur l'IA, pour les rendre plus intuitives et accessibles aux non-spécialistes, facilitera une adoption plus large dans les pratiques juridiques quotidiennes. Enfin, encourager les collaborations entre experts en droit, ingénieurs en intelligence artificielle et autres parties prenantes est essentiel pour développer des solutions véritablement innovantes et adaptées aux besoins réels du secteur juridique.

En conclusion, ce travail ouvre la voie à de nombreuses possibilités pour l'application de l'intelligence artificielle dans le domaine du droit. Il est essentiel de poursuivre les recherches et les développements dans cette direction pour exploiter pleinement le potentiel de ces technologies émergentes et transformer la pratique juridique moderne.

ANNEXES

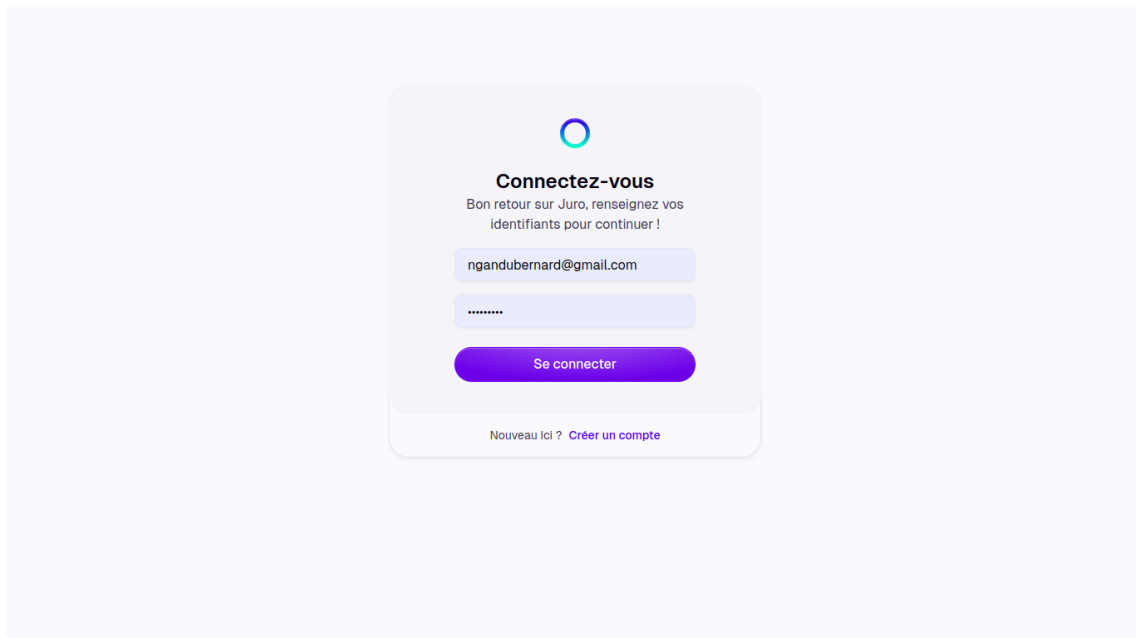


FIGURE .1 : Capture d  cran page de connexion

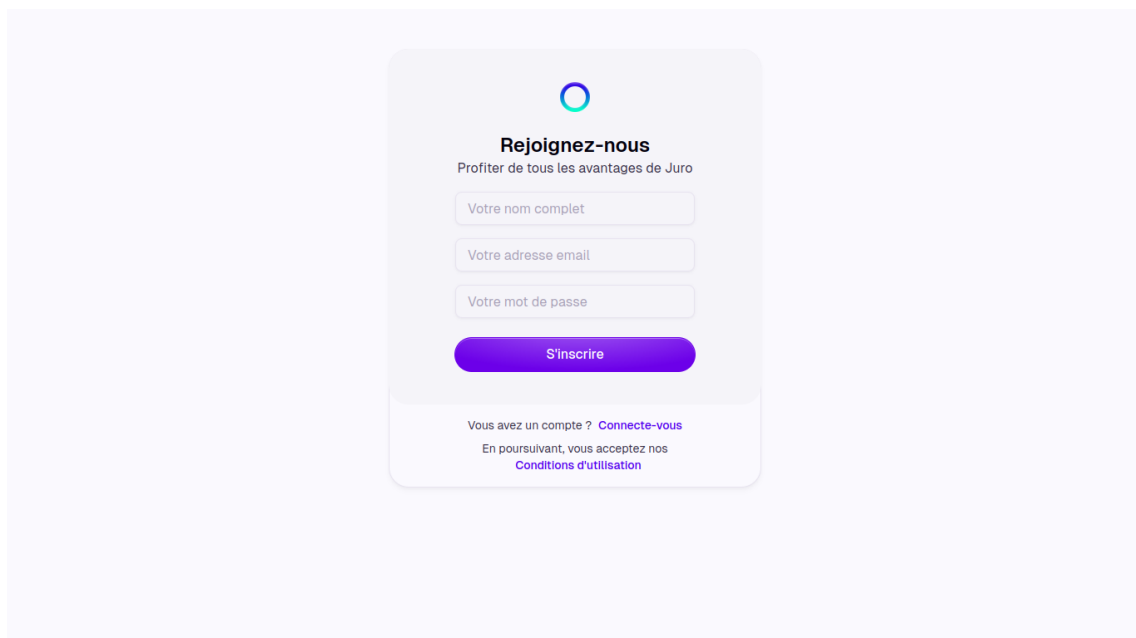


FIGURE .2 : Capture d  cran page d'inscription



FIGURE .3 : Capture d'écran page de chat

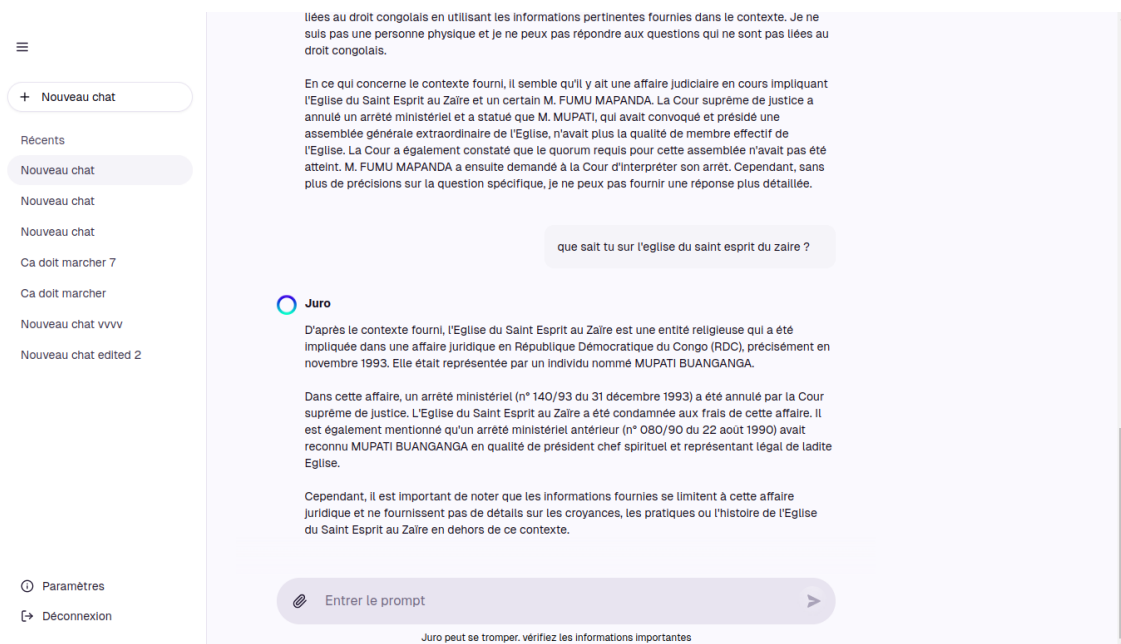


FIGURE .4 : Capture d'écran page lecture de message

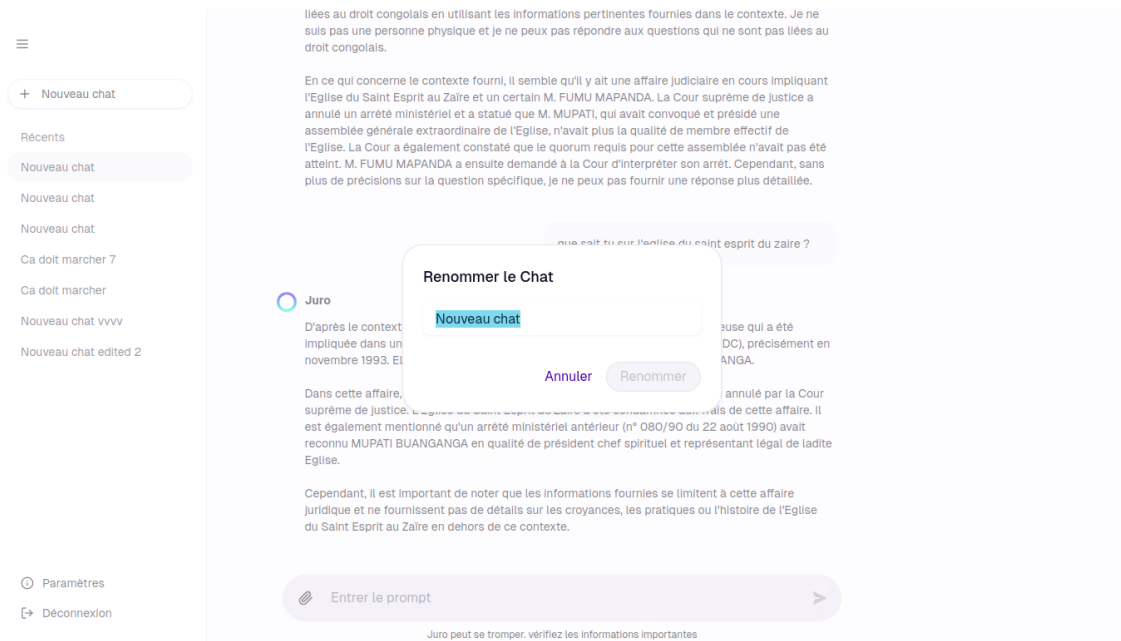


FIGURE .5 : Capture d'écran modifier le chat

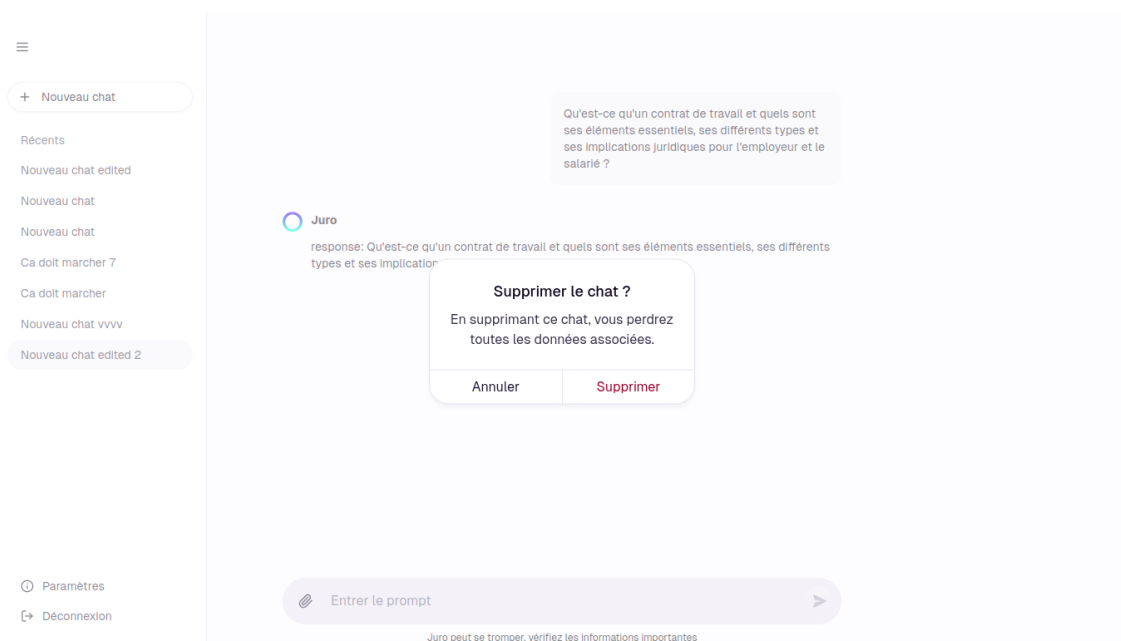


FIGURE .6 : Capture d'écran supprimer le chat


```

# Extract images and use OCR if necessary
image_list = page.get_images(full=True)
for image_index, img in enumerate(page.get_images(full=True)):
    xref = img[0]
    # Extract the image bytes
    base_image = doc.extract_image(xref)
    image_bytes = base_image["image"]
    image = Image.open(io.BytesIO(image_bytes))
    ocr_text = pytesseract.image_to_string(
        image,
        lang='eng',
        output_type=Output.STRING
    )

    extracted_content.append({
        'page': page_num + 1,
        'type': 'image',
        'content': ocr_text,
        'image_index': image_index + 1
    })

```

Listing 16 : Extraction avec OCR sur les images

```

import matplotlib.pyplot as plt

years = [
    '2010', '2011', '2012', '2013', '2014', '2015', '2016', '2017',
    '2018*', '2019*', '2020*', '2021*', '2022*', '2023*', '2024*', '2025*'
]
values = [2, 5, 6.5, 9, 12.5, 15.5, 18, 26, 33, 41, 64.2, 79, 97, 120, 147, 181]

fig, ax = plt.subplots(figsize=(12, 6), dpi=100)
ax.spines['right'].set_visible(False)
ax.spines['top'].set_visible(False)
ax.spines['left'].set_visible(False)

plt.grid(alpha=0.3, zorder=0, linestyle='dashed', axis='y')
bars = plt.bar(
    years, values, hatch="/", color="#9ca3af",
    edgecolor="#374151", alpha=0.8, width=0.8, zorder=3
)

for bar in bars:
    yval = bar.get_height()
    plt.text(
        bar.get_x() + bar.get_width()/2, yval + 5,
        round(yval, 1), ha='center', va='bottom', fontweight="bold"
    )

plt.ylabel('Data Volume in Zettabytes', fontweight="bold")
plt.show()

```

Listing 17 : Code python utilisé pour générer la figure 1.11

```

import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# Fonction à minimiser
def f(x, y):
    return x**2 + y**2

# Gradient de la fonction
def grad_f(x, y):
    return np.array([2*x, 2*y])

# Paramètres de descente de gradient
alpha = 0.1 # taux d'apprentissage
n_iterations = 10
x_start = np.array([0.8, 0.8]) # point de départ

# Descente de gradient
points = [x_start]
for _ in range(n_iterations):
    grad = grad_f(*points[-1])
    new_point = points[-1] - alpha * grad
    points.append(new_point)

points = np.array(points)

# Création de la grille pour le graphique
x = np.linspace(-1, 1, 400)
y = np.linspace(-1, 1, 400)
x, y = np.meshgrid(x, y)
z = f(x, y)

# Plot 3D
fig = plt.figure(figsize=(14, 9))
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(x, y, z, alpha=0.5, cmap='winter', edgecolor='none')
ax.scatter(points[:, 0], points[:, 1], f(points[:, 0], points[:, 1]), color='red', s=50)
ax.view_init(45, 280)
ax.set_xlabel('X', fontweight="bold")
ax.set_ylabel('Y', fontweight="bold")
ax.set_zlabel('f(X, Y)', fontweight="bold")

plt.show()

```

Listing 18 : Code python utilisé pour générer la figure ??

```

import os
import hashlib
from tqdm import tqdm

def calculate_md5(file_path):
    """Calculate the MD5 hash of a file."""
    hash_md5 = hashlib.md5()
    with open(file_path, 'rb') as f:
        for chunk in iter(lambda: f.read(4096), b''):
            hash_md5.update(chunk)
    return hash_md5.hexdigest()

def rename_files_in_directory(directory_path):
    """Rename each file in the directory to its MD5 hash."""
    files = [
        f for f in os.listdir(directory_path)
        if not os.path.isdir(os.path.join(directory_path, f))
    ]

    for filename in tqdm(files, desc="Renaming files"):
        file_path = os.path.join(directory_path, filename)
        md5_hash = calculate_md5(file_path)
        new_file_name = md5_hash
        new_file_path = os.path.join(directory_path, new_file_name)
        os.rename(file_path, new_file_path)

directory_path = '/content/drive/MyDrive/DATA/LawLLM/PDF/'
rename_files_in_directory(directory_path)

```

Listing 19 : Script python permettant de renommer un fichier par son hash MD5

```

import os
from openai import OpenAI
import pandas as pd

questions = pd.read_csv('_magistrature.csv')
client = OpenAI(api_key=os.getenv('OPENAI_KEY'))

def generate_content(x):
    print(f"Answering question: {x['question']}")
    prompt = f"""
        Dans le contexte du Droit Congolais (RDC) précisément {x['category']},
        répondez à la question suivante : {x['question']}
    """

    completion = client.chat.completions.create(
        model="gpt-3.5-turbo",
        messages=[
            {
                "role": "system",
                "content": """
                    Tu es un assistant virtuel qui peut répondre
                    à des questions sur le droit congolais (RDC).
                """
            },
            {"role": "user", "content": prompt}
        ]
    )
    return completion.choices[0].message.content

questions['model'] = 'gpt-3.5-turbo'
questions['answer'] = questions.apply(lambda x: generate_content(x), axis=1)
questions.to_csv('./data/answers-gpt-3.5-turbo.csv', index=False)

```

Listing 20 : Évaluation des modèles OpenAI sur le test de magistrature 2022.

```

import matplotlib.pyplot as plt

years = ['gpt3.5-turbo', 'gemini-pro', 'llama2', 'mistral', 'vicuna']
values = [65.33, 59.20, 43.11, 38.00, 36.44]

fig, ax = plt.subplots(figsize=(12, 6), dpi=100)
ax.spines['right'].set_visible(False)
ax.spines['top'].set_visible(False)
ax.spines['left'].set_visible(False)

plt.grid(alpha=0.3, zorder=0, linestyle='dashed', axis='y')
bars = plt.bar(
    years,
    values,
    hatch="/",
    color="#9ca3af",
    edgecolor="#374151",
    alpha=0.8,
    width=0.8,
    zorder=3
)

for bar in bars:
    yval = bar.get_height()
    plt.text(
        bar.get_x() + bar.get_width()/2,
        yval + 5,
        round(yval, 1),
        ha='center',
        va='bottom',
        fontweight="bold"
    )

plt.ylabel('Résultat au test (en pourcentage)', fontweight="bold")
plt.show()

```

Listing 21 : Code python utilisé pour générer la figure 3.2

BIBLIOGRAPHIE

- [1] Neeraj AGARWAL. *N-gram Language Modeling in Natural Language Processing - KD-nuggets*. URL : <https://www.kdnuggets.com/2022/06/ngram-language-modeling-natural-language-processing.html>.
- [2] Md. AL-AMIN, Mohammad Shazed ALI, Abdus SALAM, Arif KHAN, Ashraf ALI, Ahsan ULLAH, Md Nur ALAM et Shamsul Kabir CHOWDHURY. *History of generative Artificial Intelligence (AI) chatbots : past, present, and future development*. 2024. arXiv : [2402.05122](https://arxiv.org/abs/2402.05122) [cs.GL].
- [3] Shraddha ANALA. *A Guide to Word Embedding*. 2020. URL : <https://towardsdatascience.com/a-guide-to-word-embeddings-8a23817ab60f>.
- [4] Vincent Barra ANTOINE CORNUÉJOLS Laurent Michet. *Apprentissage artificiel : Deep learning, concepts et algorithmes*. 3rd. Eyrolles, 2018, p. 239-263.
- [5] Jean-Luc AUBERT et Éric SAVAUX. *Introduction au droit et thèmes fondamentaux du droit civil*. 2010. URL : http://books.google.ie/books?id=gnQf0AEACAAJ&dq=Introduction+au+droit+et+th%C3%A8mes+fondamentaux+du+droit+civil&hl=&cd=1&source=gbs_api.
- [6] Zechen BAI, Pichao WANG, Tianjun XIAO, Tong HE, Zongbo HAN, Zheng ZHANG et Mike Zheng SHOU. *Hallucination of Multimodal Large Language Models : A Survey*. 2024. arXiv : [2404.18930](https://arxiv.org/abs/2404.18930) [cs.CV].
- [7] Piotr BOJANOWSKI, Edouard GRAVE, Armand JOULIN et Tomas MIKOLOV. *Enriching Word Vectors with Subword Information*. 2016. arXiv : [1607.04606](https://arxiv.org/abs/1607.04606) [cs.CL].
- [8] Giuseppe BONACCORSO. *Mastering Machine Learning Algorithms*. Packt Publishing Ltd, 2018. URL : http://books.google.ie/books?id=2HteDwAAQBAJ&printsec=frontcover&dq=Mastering+Machine+Learning+Algorithms&hl=&cd=1&source=gbs_api.
- [9] Soha BOROOJERDI et George RUDOLPH. "Handwritten Multi-Digit Recognition With Machine Learning". In : mai 2022, p. 1-6. DOI : [10.1109/IETC54973.2022.9796722](https://doi.org/10.1109/IETC54973.2022.9796722).
- [10] Sergey BRIN et Lawrence PAGE. "The anatomy of a large-scale hypertextual Web search engine". In : *Computer Networks and ISDN Systems* 30.1 (1998). Proceedings of the Seventh International World Wide Web Conference, p. 107-117. ISSN : 0169-7552. DOI : [https://doi.org/10.1016/S0169-7552\(98\)00110-X](https://doi.org/10.1016/S0169-7552(98)00110-X). URL : <https://www.sciencedirect.com/science/article/pii/S016975529800110X>.
- [11] Tom B. BROWN et al. *Language Models are Few-Shot Learners*. 2020. arXiv : [2005.14165](https://arxiv.org/abs/2005.14165) [cs.CL]. URL : <https://arxiv.org/abs/2005.14165>.
- [12] Ambroise Katambu BULAMBO. *La RD Congo. Les TICS dans l'apprentissage des Droits de l'Homme*. Editions Universitaires Europeennes, 2012. URL : http://books.google.ie/books?id=PVNeLwEACAAJ&dq=INTRODUCTION+GENERAL+A+L%E2%80%9999ETUDE+DU+DR0IT+congolais&hl=&cd=5&source=gbs_api.

- [13] Olivier CAELEN et Marie-Alice BLETE. *Developing Apps with GPT-4 and ChatGPT*. "O'Reilly Media, Inc.", 2023. URL : http://books.google.ie/books?id=XaXTEAAQBAJ&printsec=frontcover&dq=Developing+Apps+with+GPT-4+and+ChatGPT&hl=&cd=1&source=gbs_api.
- [14] Soumen CHAKRABARTI. *Mining the Web*. Morgan Kaufmann, 2002. URL : http://books.google.ie/books?id=5Zxw1h6yc_UC&printsec=frontcover&dq=Mining+the+Web.+Discovering+Knowledge+from+Hypertext+Data&hl=&cd=1&source=gbs_api.
- [15] Wei-Lin CHIANG et al. *Vicuna : An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality*. 2023. URL : <https://lmsys.org/blog/2023-03-30-vicuna/>.
- [16] Jiayi CUI, Zongjian LI, Yang YAN, Bohua CHEN et Li YUAN. *ChatLaw : Open-Source Legal Large Language Model with Integrated External Knowledge Bases*. 2023. arXiv : 2306.16092 [cs.CL].
- [17] Marc Peter DEISENROTH, A. Aldo FAISAL et Cheng Soon ONG. *Mathematics for Machine Learning*. Cambridge University Press, 2020. URL : http://books.google.ie/books?id=t4XQDwAAQBAJ&printsec=frontcover&dq=Mathematics+for+Machine+Learning&hl=&cd=1&source=gbs_api.
- [18] Pranav Nataraj DEVARAJ, Rakesh Teja P V AU2, Aaryav GANGRADE et Manoj Kumar R. *Development of a Legal Document AI-Chatbot*. 2023. arXiv : 2311.12719 [cs.AI].
- [19] Google DEVELOPERS. *Machine Learning : Embeddings*. 2022. URL : <https://developers.google.com/machine-learning/crash-course/embeddings/video-lecture>.
- [20] Jacob DEVLIN, Ming-Wei CHANG, Kenton LEE et Kristina Toutanova. *BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2018. arXiv : 1810.04805 [cs.CL].
- [21] ILUNGA KABULULU ETIENNE. *INTRODUCTION GENERALE A L'ETUDE DU DROIT*. 1^{re} éd. DRC : leganet.cd, jan. 2012. URL : <https://leganet.cd/Doctrine.textes/g%C3%A9n%C3%A9ralit%C3%A9/Intro.ilunga.2012.pdf>.
- [22] Peter FLACH. *Machine Learning*. Cambridge University Press, 2012. URL : http://books.google.ie/books?id=0fp4h_oXsZ4C&printsec=frontcover&dq=machine+learnig+peter+flach&hl=&cd=1&source=gbs_api.
- [23] David FOSTER. *Generative Deep Learning*. "O'Reilly Media, Inc.", 2019. URL : http://books.google.ie/books?id=RqegDwAAQBAJ&printsec=frontcover&dq=Generative+Deep+Learning&hl=&cd=1&source=gbs_api.
- [24] Ian GOODFELLOW, Yoshua BENGIO et Aaron COURVILLE. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016.
- [25] Peter J. GREEN. "Markov chain Monte Carlo in Practice". In : 1996. URL : <https://api.semanticscholar.org/CorpusID:125093681>.
- [26] MAKWA KANDUNGI JOËL. *Cours d'introduction générale à l'étude du droit - part 1*. 1^{re} éd. T. 1. RDC : Université Nouveaux Horizons, oct. 2021.
- [27] MAKWA KANDUNGI JOËL. *Cours d'introduction générale à l'étude du droit - part 2*. 1^{re} éd. T. 2. RDC : Université Nouveaux Horizons, oct. 2021.
- [28] MAKWA KANDUNGI JOËL. *Cours d'introduction générale à l'étude du droit - part 3*. 1^{re} éd. T. 3. RDC : Université Nouveaux Horizons, oct. 2021.

- [29] JASPREET. *A Concise History of Neural Networks - Towards Data Science*. 2022. URL : <https://towardsdatascience.com/a-concise-history-of-neural-networks-2070655d3fec>.
- [30] Albert Q. JIANG et al. *Mistral 7B*. 2023. arXiv : [2310.06825](https://arxiv.org/abs/2310.06825) [cs.CL].
- [31] Dan JURAFSKY et James H. MARTIN. *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*. Upper Saddle River, N.J. : Pearson Prentice Hall, 2009. ISBN : 9780131873216 0131873210. URL : http://www.amazon.com/Speech-Language-Processing-2nd-Edition/dp/0131873210/ref=pd_bxgy_b_img_y.
- [32] JURIXIO. *La hiérarchie des normes (ou pyramide des normes de Kelsen)*. 2023. URL : <https://www.jurixio.fr/la-hierarchie-des-normes-la-pyramide-de-kelsen/>.
- [33] Raphael KASSEL. *Epoch : Définition, mode de fonctionnement et utilisation*. 2023. URL : <https://datascientest.com/qu-est-ce-qu-un-epoch-en-machine-learning>.
- [34] Patrick LEWIS et al. *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. 2021. arXiv : [2005.11401](https://arxiv.org/abs/2005.11401) [cs.CL].
- [35] LUISQUINTANILLA. *Machine learning tasks - ML.NET*. 2022. URL : <https://learn.microsoft.com/en-us/dotnet/machine-learning/resources/tasks>.
- [36] A. A. MARKOV. "Essai d'une recherche statistique sur le texte du roman "Eugene Onegin" illustrant la liaison des epreuve en chain ('Example of a statistical investigation of the text of "Eugene Onegin" illustrating the dependence between samples in chain')". In : *Izvestia Imperatorskoi Akademii Nauk (Bulletin de l'Académie Impériale des Sciences de St.-Pétersbourg)*. 6^e sér. 7 (1913). English translation by Morris Halle, 1956., p. 153-162.
- [37] Tomas MIKOLOV, Kai CHEN, Greg CORRADO et Jeffrey DEAN. *Efficient Estimation of Word Representations in Vector Space*. 2013. arXiv : [1301.3781](https://arxiv.org/abs/1301.3781) [cs.CL].
- [38] M. MINSKY et S. PAPERT. *Perceptrons*. Cambridge, MA : MIT Press, 1969.
- [39] Lucain Kasongo MWADIAVITA. *Précis d'introduction générale au droit positif congolais*. Editions L'Harmattan, 2023. URL : http://books.google.ie/books?id=6r7MEAAQBAJ&printsec=frontcover&dq=INTRODUCTION+GENERAL+A+L%E2%80%9999ETUDE+DU+DROIT+congolais&hl=&cd=1&source=gbs_api.
- [40] OPENAI. *GPT-4 Technical Report*. 2023. arXiv : [2303.08774](https://arxiv.org/abs/2303.08774) [cs.CL].
- [41] Jeffrey PENNINGTON, Richard SOCHER et Christopher D. MANNING. "GloVe : Global Vectors for Word Representation". In : *Empirical Methods in Natural Language Processing (EMNLP)*. 2014, p. 1532-1543. URL : <http://www.aclweb.org/anthology/D14-1162>.
- [42] Cherie M POLAND. *Generative AI and US Intellectual Property Law*. 2023. arXiv : [2311.16023](https://arxiv.org/abs/2311.16023) [cs.CY].
- [43] Daniel PRITCHETT. *Build Chatbot Interactions*. Pragmatic Bookshelf, 2019. URL : http://books.google.ie/books?id=9xWjDwAAQBAJ&printsec=frontcover&dq=9781680506327&hl=&cd=3&source=gbs_api.
- [44] *Qu'est-ce que RAG ? - Explication de la génération augmentée par extraction - AWS*. URL : <https://aws.amazon.com/fr/what-is/retrieval-augmented-generation/>.
- [45] *Qu'est-ce que le surapprentissage*. URL : <https://aws.amazon.com/fr/what-is/overfitting>.

- [46] RAPHAËL. *Pyramide de Kelsen et hiérarchie des normes*. 2024. URL : <https://aideauxtd.com/pyramide-de-kelsen/>.
- [47] Bruce RATNER. *Statistical and Machine-Learning Data Mining*. CRC Press, 2012. URL : http://books.google.ie/books?id=I3KZCfmhWJwC&printsec=frontcover&dq=Statistical+and+Machine-Learning+Data+Mining:+Techniques+for+Better+Predictive+Modeling+and+Analysis+of+Big+Data,+Second+Edition&hl=&cd=2&source=gbs_api.
- [48] Ronald L. RIVEST. *The MD5 Message-Digest Algorithm*. Internet RFC 1321. 1992. URL : <http://tools.ietf.org/html/rfc1321>.
- [49] D. ROTHMAN. *Transformers for Natural Language Processing*. Packt Publishing Ltd, 2022.
- [50] Jamell SAMUELS. *One-Hot Encoding and Two-Hot Encoding : An Introduction*. Jan. 2024. DOI : [10.13140/RG.2.2.21459.76327](https://doi.org/10.13140/RG.2.2.21459.76327).
- [51] Ray SMITH. *Tesseract OCR Engine*. 2007. URL : <https://web.archive.org/web/20160819190257/tesseract-ocr.googlecode.com/files/TesseractOSCON.pdf>.
- [52] Aegis SOFTTECH. *How to Learn Machine Learning in Three months and advance your IT Career ?* 2019. URL : <https://www.aegissofttech.com/articles/learn-machine-learning.html>.
- [53] Heydar SOUDANI, Evangelos KANOULAS et Faegheh HASIBI. *Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge*. 2024. arXiv : [2403.01432](https://arxiv.org/abs/2403.01432) [cs.CL].
- [54] A. M. TURING. "I.—COMPUTING MACHINERY AND INTELLIGENCE". In : *Mind* LIX.236 (oct. 1950), p. 433-460. ISSN : 0026-4423. DOI : [10.1093/mind/LIX.236.433](https://doi.org/10.1093/mind/LIX.236.433). eprint : <https://academic.oup.com/mind/article-pdf/LIX/236/433/30123314/lix-236-433.pdf>. URL : <https://doi.org/10.1093/mind/LIX.236.433>.
- [55] Gemini TEAM. *Gemini : A Family of Highly Capable Multimodal Models*. 2023. arXiv : [2312.11805](https://arxiv.org/abs/2312.11805) [cs.CL].
- [56] Gemma TEAM. "Gemma". In : (2024). DOI : [10.34740/KAGGLE/M/3301](https://doi.org/10.34740/KAGGLE/M/3301). URL : <https://www.kaggle.com/m/3301>.
- [57] Siri TEAM. *Deep Learning for Siri's Voice : On-device Deep Mixture Density Networks for Hybrid Unit Selection Synthesis*. URL : <https://machinelearning.apple.com/research/siri-voices>.
- [58] Hugo TOUVRON et al. *LLaMA : Open and Efficient Foundation Language Models*. 2023. arXiv : [2302.13971](https://arxiv.org/abs/2302.13971) [cs.CL].
- [59] Ashish VASWANI, Noam SHAZEER, Niki PARMAR, Jakob USZKOREIT, Llion JONES, Aidan N. GOMEZ, Lukasz KAISER et Illia POLOSUKHIN. *Attention Is All You Need*. 2023. arXiv : [1706.03762](https://arxiv.org/abs/1706.03762) [cs.CL].
- [60] *Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2020, with forecasts from 2021 to 2025*. 2023. URL : <https://www.statista.com/statistics/871513/worldwide-data-created/>.
- [61] Haohan WANG et Bhiksha RAJ. *On the Origin of Deep Learning*. 2017. arXiv : [1702.07800](https://arxiv.org/abs/1702.07800) [cs.LG].

- [62] Joseph WEIZENBAUM. “ELIZA—a computer program for the study of natural language communication between man and machine”. In : *Commun. ACM* 9.1 (1966), 36–45. ISSN : 0001-0782. DOI : [10.1145/365153.365168](https://doi.org/10.1145/365153.365168). URL : <https://doi.org/10.1145/365153.365168>.
- [63] Joseph WEIZENBAUM. “ELIZA—a computer program for the study of natural language communication between man and machine”. In : *Communications of the ACM* 9 (1966), p. 36 -45. URL : <https://api.semanticscholar.org/CorpusID:1896290>.
- [64] WIKIPÉDIA. *Beautiful Soup* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 1-mai-2023]. 2023. URL : http://fr.wikipedia.org/w/index.php?title=Beautiful_Soup&oldid=203869267.
- [65] WIKIPÉDIA. *Corpus* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 21-juin-2023]. 2023. URL : <http://fr.wikipedia.org/w/index.php?title=Corpus&oldid=205358830>.
- [66] WIKIPÉDIA. *MD5* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 21-avril-2023]. 2023. URL : <http://fr.wikipedia.org/w/index.php?title=MD5&oldid=203545836>.
- [67] WIKIPÉDIA. *Robot d’indexation* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 17-juin-2023]. 2023. URL : http://fr.wikipedia.org/w/index.php?title=Robot_d%27indexation&oldid=205243717.
- [68] WIKIPÉDIA. *Algorithme* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 9-mars-2024]. 2024. URL : <http://fr.wikipedia.org/w/index.php?title=Algorithme&oldid=213207404>.
- [69] WIKIPÉDIA. *Frank Rosenblatt* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 4-mars-2024]. 2024. URL : http://fr.wikipedia.org/w/index.php?title=Frank_Rosenblatt&oldid=213039317.
- [70] WIKIPÉDIA. *Hans Kelsen* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 17-mars-2024]. 2024. URL : http://fr.wikipedia.org/w/index.php?title=Hans_Kelsen&oldid=213425509.
- [71] WIKIPÉDIA. *Hiérarchie des normes* — *Wikipédia, l’encyclopédie libre*. [En ligne ; Page disponible le 5-février-2024]. 2024. URL : http://fr.wikipedia.org/w/index.php?title=Hi%C3%A9rarchie_des_normes&oldid=212163472.
- [72] Tomáš ZEMČÍK. “A Brief History of Chatbots”. In : *DEStech Transactions on Computer Science and Engineering* (oct. 2019). DOI : [10.12783/dtcse/aicae2019/31439](https://doi.org/10.12783/dtcse/aicae2019/31439).
- [73] Banghua ZHU, Evan FRICK, Tianhao WU, Hanlin ZHU et Jiantao JIAO. *Starling-7B : Improving LLM Helpfulness & Harmlessness with RLAI*. 2023.
- [74] Zhihao ZHU, Ninglu SHAO, Defu LIAN, Chenwang WU, Zheng LIU, Yi YANG et Enhong CHEN. *Understanding Privacy Risks of Embeddings Induced by Large Language Models*. 2024. arXiv : [2404.16587](https://arxiv.org/abs/2404.16587) [cs.CL].

TSHABU NGANDU Bernard : *Conception et développement d'un chatbot basé sur un LLM
comme support de vulgarisation au système juridique Congolais* , TFC, © Juillet 2024

DIRECTEURS :

Prof. BAGULA Antoine PhD.

Ass. MBALE Landry

LIEUX :

République démocratique du Congo, Haut-Katanga, Lubumbashi

CADRE TEMPOREL :

Juillet 2024