جـامـعـة ابـن طـفـيـل
ⵜⴰⵙⴷⴰⵡⵉⵜ ⵙⵉ ⵟⴼⵉⵍ
Ibn Tofaïl University

Faculté des Sciences

Université Ibn Tofail

Faculté des Sciences, Kénitra

Mémoire de Projet de Fin d'Etudes

Master Intelligence Artificielle et Réalité Virtuelle

Sujet :
Création et Intégration d'un chatbot intelligent sur la plateforme Analytix

Établissement d'accueil : Inorisk, Casablanca

Elaboré par : Mr.Mohamed Oussama el Mousaouy :

3**Encadré par :* Mr. Tarik Boujiha (etablissement -UIT)
Mr. Nordine Ferhaoui (Entreprise)
Mr. Youssef Kinany Alaoui (Entreprise)

Soutenu le 2024, devant le jury composé de :

- Mr ou Mme ..... (Etablissement)
- Mr ou Mme ..... (Etablissement)
- Mr ou Mme ..... (Etablissement)
- Mr ou Mme ..... (Etablissement)

Année universitaire 2023/2024

Dédicace

Je dédie ce travail à mes parents, **Fatima et Abdlader**, pour leur amour, leur soutien inconditionnel, et leurs encouragements constants tout au long de mon parcours. Votre patience et vos sacrifices m'ont permis d'avancer sereinement dans cette aventure.

À mes sœurs, **Yousra et Doha**, qui m'ont toujours inspiré par leur douceur et leur bienveillance. Vous avez été des soutiens précieux, et je vous remercie pour votre présence et vos encouragements.

À mon frère **Amine**, pour ta complicité et ta joie de vivre, qui m'ont souvent redonné de la force.

Enfin, merci à chacun d'entre vous pour votre amour, votre patience et votre confiance. C'est grâce à vous que je suis arrivé jusqu'ici.

Remerciements

Je tiens à exprimer ma gratitude envers toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce projet de fin d'études.

Je remercie tout particulièrement **Monsieur FERHAOUI Nordine**, **Monsieur ATTOUF**, et **Monsieur Youssef Kinany Alaoui Abdellatif** pour leur accompagnement et leur expertise tout au long de mon stage. Votre soutien et vos conseils ont été essentiels pour la réussite de ce projet.

Je souhaite également exprimer mes sincères remerciements à **Madame AFKHKHAR Zaineb** pour son appui constant et son implication.

Je remercie ma coordinatrice, **Madame Touahni Raja**, pour sa disponibilité et son soutien durant tout mon parcours.

Enfin, un grand merci à mon encadrant académique, **Monsieur Traik Boujiha**, pour ses précieux conseils et pour son suivi tout au long de ce projet. Votre encadrement et vos encouragements ont été d'une grande aide pour atteindre mes objectifs.

Table des matières

Table des figures

Liste des tableaux

Introduction

Ce rapport de stage présente le projet d'intégration d'un chatbot intelligent au sein de l'entreprise Inforisk, spécialisée dans le renseignement commercial au Maroc. Ce projet s'inscrit dans une démarche de transformation digitale visant à optimiser l'expérience utilisateur et la gestion des leads sur les plateformes web d'Inforisk, notamment sur sa plateforme Analytix dédiée à la gestion des risques.

Le rapport se structure en cinq chapitres. Le premier chapitre expose une présentation détaillée d'Inforisk, son positionnement sur le marché, ses services, ses plateformes, et son organisation interne, en mettant l'accent sur le rôle crucial du département IT. Le deuxième chapitre dresse un état de l'art des technologies de chatbot, en abordant leur évolution historique, les différents types de chatbots, l'importance des grands modèles de langage (LLM) comme GPT et BERT, et les techniques d'optimisation comme le fine-tuning et le prompt engineering.

Le troisième chapitre se focalise sur la méthodologie de collecte et de préparation des données qui ont servi à alimenter et à entraîner le chatbot. Le quatrième chapitre explique en détail le choix du modèle de langage Mistral NeMo pour le développement du chatbot, en le comparant à d'autres modèles comme Llama 3 (7B) et GPT-4. Enfin, le cinquième chapitre décrit l'intégration technique du chatbot sur la plateforme Analytix, en mettant l'accent sur l'environnement de développement, la conception centrée sur l'utilisateur (UX/UI), et les tests effectués pour valider le bon fonctionnement et la robustesse du système.

Ce rapport met en lumière les différentes étapes du projet, depuis la phase de conception jusqu'à l'intégration finale, en passant par le choix technologique et la préparation des données. Il témoigne de l'importance de l'IA dans l'amélioration des processus d'entreprise et de l'expérience client.

Chapitre 1

Présentation d'Inforisk

Introduction

Ce rapport présente le projet d'intégration d'un chatbot basé sur l'intelligence artificielle (IA) chez Inforisk. L'objectif principal est d'améliorer l'engagement utilisateur, l'efficacité du support client et la gestion des leads sur les plateformes Inforisk et Analytix.

Dans ce chapitre, nous présenterons Inforisk, ses activités et son positionnement sur le marché. Nous détaillerons ensuite les objectifs spécifiques du projet, la méthodologie adoptée et le plan de travail. Enfin, nous analyserons l'état de l'art des chatbots IA et des solutions existantes dans des contextes similaires.

1.1 Inforisk en bref

Qui sommes-nous ?

INFORISK, créée en 2007 et basée à Casablanca, est une société spécialisée dans le renseignement commercial sur les sociétés marocaines. Nous collectons des informations légales, financières, judiciaires et commerciales pour fournir une vision complète de l'historique d'une entreprise (voir Figure 1.1).

Figure 1.1 – Logo d'Inforisk

Notre expertise

Inforisk collecte, encode, structure et rapproche des données primaires telles que les états financiers, les éléments d'identification et les données légales et judiciaires. Notre base de données, mise à jour quotidiennement, se distingue par :

- Son exhaustivité : couverture de toutes les entreprises marocaines.
- Sa mise à jour : actualisation quotidienne.
- Sa fiabilité : informations croisées avec de multiples sources officielles.

La Figure 1.2 illustre les activités principales d'Inforisk.

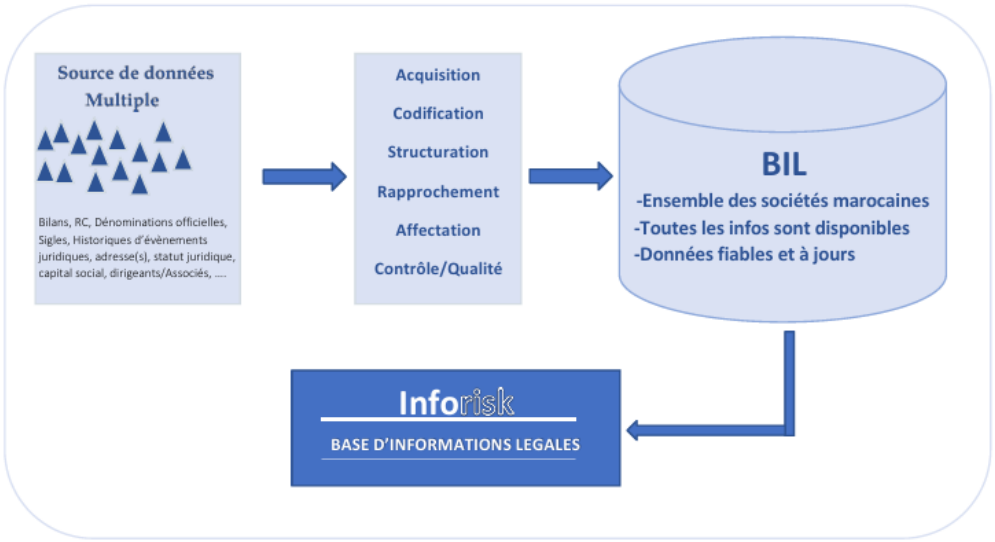

Figure 1.2 – Les activités d'Inforisk

Chiffres clés

Le tableau 1.1 présente quelques chiffres clés d'Inforisk.

Nos services

- **Risk Management :** Solutions pour réduire les risques de contreparties (clients ou fournisseurs), avec des outils d'aide à la décision pour l'octroi de crédit et la gestion des délais de paiement.
- **Marketing et Prospection :** Informations commerciales et marketing qualifiées pour une prospection ciblée et efficace.
- **Expertise et Conseil :** Réalisation d'études sectorielles ou micro-économiques sur mesure pour l'analyse des risques.

Chiffre	Description
750K	Entreprises enregistrées
150K	Événements collectés par an
2M	Dirigeants et Associés
2M	États financiers de 2005 à 2021

Table 1.1 – Chiffres clés d'Inforisk

Nos plateformes

- **Charika.ma :** Première plateforme B2B au Maroc offrant un accès à une base de données complète, fiable et mise à jour sur les entreprises marocaines (voir Figure 1.3).

Figure 1.3 – Logo de Charika.ma

- **AnalytiX :** Plateforme dédiée à la gestion et à la surveillance des risques de contreparties, couvrant le Maroc et l'international (voir Figure 1.4).

Figure 1.4 – Logo d'AnalytiX

Nos clients

Notre clientèle se compose d'entreprises issues de divers secteurs (privé, semi-public et public), notamment des banques, des organismes de crédit et des compagnies d'assurance.

Le département IT d'Inforisk

Le département informatique d'Inforisk est structuré en différentes équipes, incluant l'équipe de Data Science et l'équipe de développement. Ces équipes travaillent en étroite collaboration pour garantir le fonctionnement optimal des services d'Inforisk. La Figure 1.5 présente l'organisation de ce département.

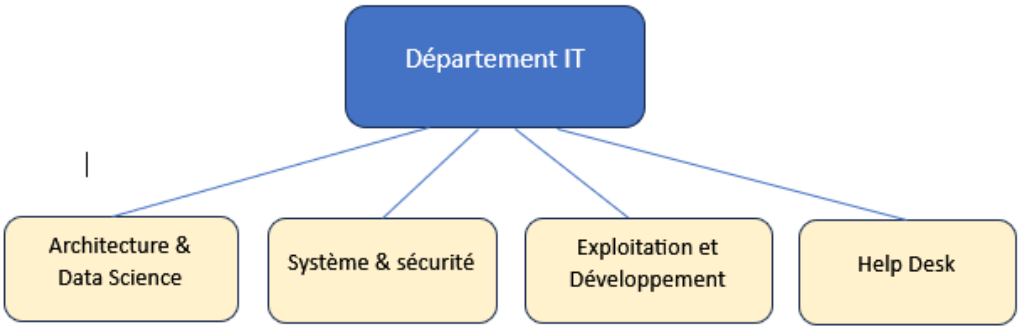

Figure 1.5 – Le département informatique d'Inforisk

1.2 Objectifs et mise en œuvre du projet

Objectif principal

Le projet vise à concevoir, développer et intégrer un chatbot IA pour améliorer l'engagement utilisateur, l'efficacité du support client et la gestion des leads sur les plateformes Inforisk et Analytix.

Etapes clés du projet

1. **Benchmark des solutions de chatbot IA**
   - Identifier les technologies IA, NLP, Machine Learning et les frameworks/API disponibles sur le marché.
   - Analyser les fonctionnalités : capacités de traitement des requêtes, gestion des conversations, adaptabilité aux besoins métiers et compatibilité avec les infrastructures existantes.
   - Évaluer la flexibilité et l'évolutivité des solutions en fonction des exigences futures du projet.

- Sélectionner la solution optimale en tenant compte du coût, de la facilité d'intégration, des capacités d'automatisation, de l'intelligence contextuelle et de l'efficacité des algorithmes d'apprentissage continu.

2. **Intégration du chatbot**
   - **Sur www.inforisk.ma**
     - Fournir des réponses aux questions fréquentes (FAQ) sur les produits, services et offres.
     - Expliquer les caractéristiques des produits de manière interactive et accessible.
     - Collecter des données utilisateur (informations démographiques, préférences) pour alimenter la génération de leads.
   - **Sur la plateforme Analytix**
     - Offrir un support client automatisé : prise en charge des requêtes, orientation vers les ressources appropriées et résolution des problèmes courants.
     - Optimiser la navigation : proposer un guidage personnalisé à travers les fonctionnalités de la plateforme.
     - Mettre en place un reporting en temps réel : suivre les performances du chatbot et son impact sur les interactions client.
3. **Amélioration de l'expérience utilisateur**
   - Intégrer une interaction conversationnelle naturelle (NLP) pour permettre aux utilisateurs d'interagir avec le chatbot dans un langage courant.
   - Personnaliser les réponses en fonction du comportement, des préférences et des interactions passées des utilisateurs.
   - Assurer une disponibilité omnicanal (web, mobile, messagerie instantanée) pour une expérience utilisateur fluide et continue.
4. **Collecte et gestion des leads**
   - Recueillir des données clés sur les utilisateurs : coordonnées, préférences, produits d'intérêt.
   - Qualifier les leads automatiquement : filtrer et prioriser les leads en fonction de leur probabilité de conversion.
   - Intégrer le chatbot au CRM existant : transmettre les données col-

lectées à l'équipe commerciale pour une exploitation efficace.
- Automatiser les processus commerciaux : prise de rendez-vous, envoi d'informations complémentaires.

5. **Tests et optimisations**
   - Mettre en place des tests unitaires et une intégration continue pour garantir la robustesse du chatbot dans différents scénarios.
   - Réaliser des tests fonctionnels pour valider les réponses aux sollicitations des utilisateurs et la gestion des erreurs.
   - Assurer une optimisation continue du chatbot en se basant sur les retours utilisateurs et les données de performance.
   - Produire un rapport de tests avec des recommandations d'amélioration pour garantir la performance du chatbot à long terme.

1.3 Conclusion

Le projet d'intégration d'un chatbot IA chez Inforisk s'inscrit dans une démarche de transformation digitale visant à améliorer l'expérience utilisateur, la gestion des leads et le support client sur nos plateformes.

En exploitant le potentiel de l'IA, nous visons à créer un chatbot offrant une expérience personnalisée et efficiente, capable de collecter des données précieuses pour nos équipes commerciales et de contribuer au renforcement de la relation client.

Chapitre 2

État de l'Art

2.1 Introduction

Ce chapitre a pour objectif de présenter les concepts fondamentaux liés aux chatbots et aux grands modèles de langage (LLM). Nous aborderons leur évolution historique, les différents types de chatbots, l'importance des LLM comme GPT et BERT, les techniques de Retrieval-Augmented Generation (RAG), ainsi que le fine-tuning et le prompt engineering pour optimiser leurs performances.

2.2 Présentation des Chatbots

Historique des Chatbots

Les chatbots sont des systèmes logiciels conçus pour interagir avec les utilisateurs via le langage, qu'il soit oral ou textuel. Le premier chatbot connu, ELIZA, développé dans les années 1960 par Joseph Weizenbaum, simulait une psychothérapeute en utilisant des règles simples pour générer des réponses. Depuis, les chatbots ont connu une évolution remarquable grâce aux progrès technologiques. Ils sont passés de systèmes basés sur des règles strictes à des modèles sophistiqués d'intelligence artificielle (IA), capables de comprendre et de générer du langage naturel. Les avancées récentes dans le domaine des réseaux neuronaux et de l'apprentissage automatique ont permis une révolution dans la capacité des chatbots à comprendre et à répondre avec une précision accrue.

1. **Chatbots basés sur des règles :** Ces chatbots fonctionnent selon un ensemble de règles prédéfinies et programmées manuellement. Ils analysent les entrées des utilisateurs à la recherche de mots-clés

spécifiques et fournissent des réponses pré-déterminées en fonction des règles correspondantes.

- **Avantages :** Faciles à mettre en œuvre, prévisibles, contrôle total sur les réponses.
- **Inconvénients :** Interactions limitées, incapable de gérer des conversations complexes ou des requêtes imprévues, nécessite une maintenance importante pour ajouter de nouvelles règles.

2. **Chatbots basés sur l'apprentissage supervisé :** Ces chatbots utilisent des algorithmes d'apprentissage automatique pour apprendre à partir d'un ensemble de données d'entraînement composé de paires questions-réponses. Ils identifient des patterns et des corrélations dans les données pour générer des réponses aux nouvelles requêtes.

   - **Avantages :** Plus flexibles que les chatbots basés sur des règles, peuvent apprendre des données et s'améliorer avec le temps, capables de gérer des conversations plus complexes.
   - **Inconvénients :** Nécessite un grand ensemble de données étiquetées pour l'entraînement, peut avoir du mal avec des requêtes hors du domaine d'entraînement.

3. **Chatbots basés sur les modèles de langage (LLM) :** Les LLM sont des modèles d'apprentissage profond entraînés sur d'énormes ensembles de données textuelles, ce qui leur permet de comprendre et de générer un langage naturel de manière fluide. Ils peuvent être utilisés pour créer des chatbots capables d'interactions plus naturelles et contextuelles.

   - **Avantages :** Interactions plus humaines et conversationnelles, capables de gérer des sujets variés et de s'adapter à différents styles de langage, peuvent générer des réponses originales et créatives.
   - **Inconvénients :** Peuvent parfois générer des réponses incorrectes ou biaisées, peuvent nécessiter des techniques de fine-tuning et de prompt engineering pour des performances optimales.

2.3 Les Grands Modèles de Langage (LLM)

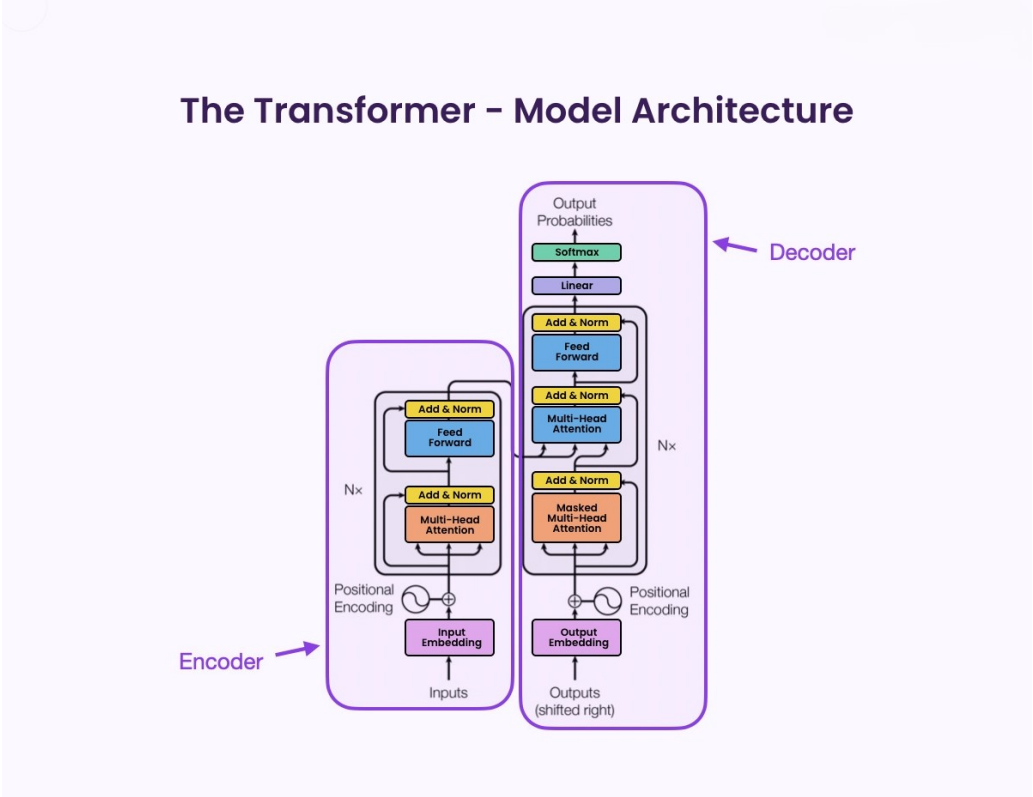

Figure 2.1 – Illustration d'un LLM en action.

Définition des LLM

Les Grands Modèles de Langage (LLM) sont des réseaux neuronaux profonds capables de traiter et de générer du langage naturel. Ils reposent généralement sur l'architecture des transformers, un modèle d'apprentissage profond introduit en 2017, qui utilise des mécanismes d'attention pour mieux comprendre le contexte des phrases. Grâce à une formation sur de gigantesques ensembles de données textuelles, les LLM peuvent répondre à des questions, écrire du contenu et réaliser d'autres tâches en langage naturel avec une précision impressionnante.

Principaux Modèles LLM Utilisés

1) **GPT (Generative Pre-trained Transformer) :** Développé par OpenAI, GPT est un modèle de génération de texte qui a gagné en popularité grâce à sa capacité à produire du texte fluide et cohérent. GPT-3, avec ses 175 milliards de paramètres, est l'un des modèles les plus connus. Il est utilisé dans divers cas d'utilisation allant des assistants virtuels à la rédaction automatisée.
2) **BERT (Bidirectional Encoder Representations from Transformers) :** Développé par Google, BERT est principalement utilisé pour les tâches de compréhension du langage. Contrairement à GPT, qui est un modèle unidirectionnel (génère du texte de gauche à droite), BERT utilise un apprentissage bidirectionnel, ce qui lui permet de mieux comprendre le contexte des mots dans une phrase.
3) **Autres modèles (T5, RoBERTa, etc.) :** Outre GPT et BERT, d'autres modèles tels que T5 (Text-To-Text Transfer Transformer) et RoBERTa (Robustly Optimized BERT Approach) ont également été largement adoptés dans le traitement du langage naturel, chacun ayant ses propres avantages en fonction des besoins de l'application.

2.4 Applications RAG (Retrieval-Augmented Generation)

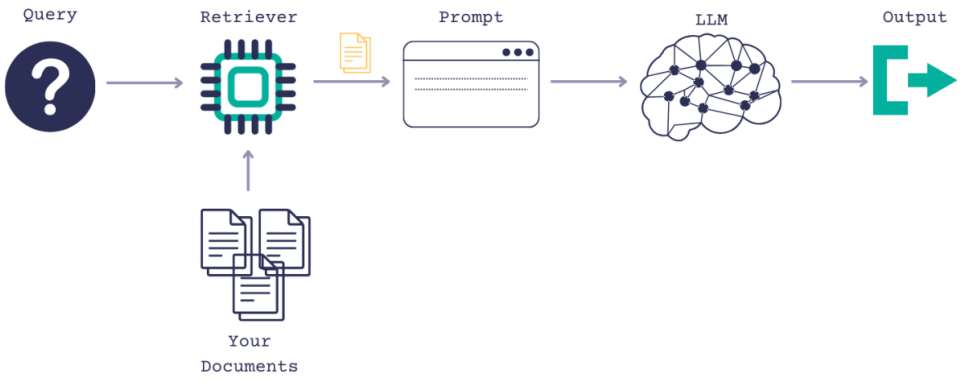

Figure 2.2 – Schéma du fonctionnement d'un système RAG.

Définition des RAG

Les systèmes RAG (Retrieval-Augmented Generation) combinent la génération de texte avec des mécanismes de récupération d'informations. Plutôt que de se limiter aux données intégrées dans le modèle lors de l'entraînement, les RAG peuvent rechercher des informations dans des bases de données ou sur le web en temps réel avant de générer une réponse. Cela permet d'améliorer la précision des réponses, en particulier lorsque le modèle doit traiter des informations spécialisées ou actualisées.

Fonctionnement des RAG

Le fonctionnement des RAG repose sur deux étapes principales :

1) **Récupération des informations :** Lorsque l'utilisateur envoie une requête, un module de recherche d'informations identifie et récupère les documents ou passages de texte les plus pertinents à partir d'une base de données externe ou du web. Ce processus utilise généralement des techniques de similarité sémantique pour trouver les informations les plus pertinentes par rapport à la requête de l'utilisateur.
2) **Génération de réponses augmentée par les informations récupérées :** Les informations récupérées lors de l'étape précédente sont ensuite fournies en contexte au modèle de langage. Le modèle utilise ces informations pour générer une réponse plus complète, précise et contextuellement pertinente. L'intégration des informations récupérées peut se faire de différentes manières, par exemple, en les ajoutant directement au prompt d'entrée du modèle ou en les utilisant pour pondérer les sorties possibles du modèle.

Avantages des RAG

Les systèmes RAG offrent plusieurs avantages clés :

- **Précision améliorée :** Accès à des informations à jour ou spécialisées non disponibles dans les données d'entraînement du modèle.
- **Adaptabilité :** Fourniture de réponses adaptées à des contextes spécifiques en récupérant les données les plus pertinentes.

- **Réduction des erreurs :** Réduction des risques d'erreurs liées à des informations obsolètes ou incorrectes dans le modèle.

Exemples d'Applications des RAG

- **Support client :** Amélioration des chatbots de support technique en récupérant des informations spécifiques à partir de bases de données d'assistance.
- **Recherche médicale :** Recherche dans des bases de données scientifiques pour fournir des réponses basées sur les dernières recherches ou études cliniques.
- **Systèmes d'aide à la rédaction :** Recherche et citation automatique de sources dans des articles ou documents, améliorant la précision et la fiabilité du contenu généré.

2.5 Fine-Tuning et Prompt Engineering

Fine-Tuning des LLM

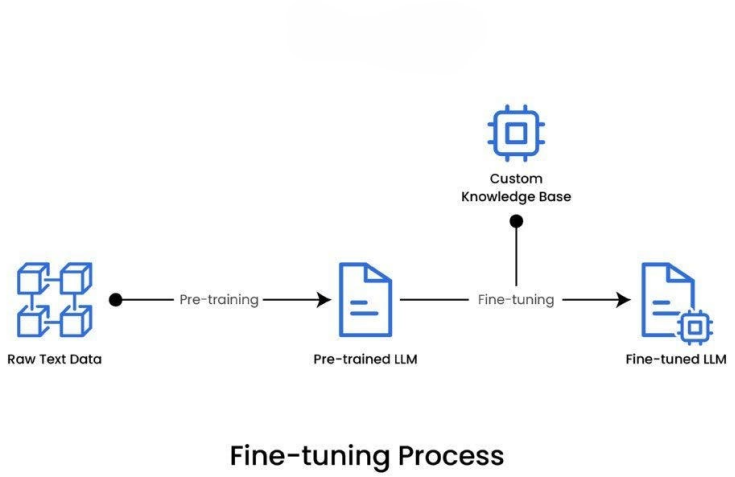

Figure 2.3 – Représentation schématique du fine-tuning d'un LLM.

Le fine-tuning consiste à ajuster un modèle LLM pré-entraîné sur un ensemble de données spécifique pour améliorer ses performances sur une tâche particulière. Par exemple, dans le cadre d'un chatbot, un modèle peut être affiné avec des dialogues spécifiques à un domaine (comme la médecine ou le service client) pour améliorer la pertinence et la précision des réponses. Cette approche permet d'adapter un modèle généraliste à des contextes ou besoins spécifiques.

Définition du Prompt Engineering

Figure 2.4 – Fonctionnement de prompt pour un chatbot.

Le prompt engineering est une technique utilisée pour optimiser les interactions avec les modèles de langage. Un "prompt" est la requête ou l'instruction donnée au modèle pour générer une réponse. En concevant des prompts plus efficaces, il est possible d'améliorer les performances du modèle sans nécessiter de modifications ou de fine-tuning complexes. Le prompt engineering est

particulièrement important pour les modèles comme GPT, où la formulation précise de la question peut grandement influencer la qualité de la réponse.

Optimisation des Interactions via le Prompt Engineering

L'optimisation des interactions avec les chatbots se fait souvent par des ajustements dans la formulation des prompts. Voici quelques stratégies :

- **Simplification et clarification des instructions :** Les prompts doivent être clairs et concis pour éviter toute ambiguïté dans la réponse générée.
- **Contexte spécifique :** Les prompts peuvent inclure des informations contextuelles pour guider le modèle vers des réponses plus pertinentes.
- **Réglage itératif :** Les développeurs ajustent souvent les prompts en fonction des performances observées pour affiner les réponses du chatbot à des requêtes spécifiques.

2.6 Conclusion

Ce chapitre a permis de dresser un panorama des concepts fondamentaux liés aux chatbots et aux grands modèles de langage (LLM), ainsi que des évolutions majeures dans ce domaine. Des chatbots basés sur des règles simples aux modèles sophistiqués de langage comme GPT ou BERT, l'évolution technologique a considérablement amélioré la capacité des systèmes à comprendre et générer du texte de manière fluide et pertinente.

L'intégration des techniques de Retrieval-Augmented Generation (RAG) marque une avancée significative, permettant aux chatbots d'accéder à des informations actualisées et spécialisées, rendant les interactions plus précises et contextuellement adaptées. Les techniques de fine-tuning et de prompt engineering jouent également un rôle crucial dans l'amélioration des performances des modèles, en ajustant les réponses pour des cas d'utilisation spécifiques.

Ce chapitre pose ainsi les bases théoriques et technologiques qui seront approfondies dans les chapitres suivants, où nous examinerons la mise en œuvre concrète de ces concepts pour le développement d'un chatbot intelligent et performant.

Chapitre 3

Collecte et Préparation des Données

3.1 Introduction

Cette section décrit le processus de collecte des données utilisées pour la création du chatbot. Ces données proviennent principalement du site web d'Inforisk et de documents internes à l'entreprise, notamment des fichiers PDF et Word. Elles constituent la base de connaissances exploitée pour l'entraînement du modèle d'intelligence artificielle.

Sources des Données

Les principales sources de données sont les suivantes :

- **Site web d'Inforisk** : `https://www.inforisk.ma/`, qui présente les services et produits offerts par l'entreprise. Ce site a été une source essentielle pour collecter des informations sur les produits et services d'Inforisk.
- **Documents internes** : Des fichiers PDF et Word fournis par l'entreprise, contenant des informations détaillées sur les services, produits, et procédures internes de l'entreprise.
- **Texte brut** : Les données collectées du site web et des documents internes ont été regroupées sous forme de texte brut. Ce texte brut a ensuite servi de contexte pour générer les questions et réponses via un modèle de génération augmentée par récupération (RAG).

Ces données ont été fondamentales pour créer une base de questions pertinentes et des réponses adaptées au domaine d'expertise d'Inforisk.

Méthodologie de Collecte

La collecte des données a été effectuée à l'aide de plusieurs outils :

- **BeautifulSoup (Python)** : Cette bibliothèque a été utilisée pour extraire les données du site web d'Inforisk. Grâce à un script d'extraction basé sur BeautifulSoup, il a été possible de parcourir les différentes pages du site, récupérer des descriptions de services, FAQ, et autres éléments textuels.

Figure 3.1 – Logo de BeautifulSoup

- **Pandas** : Utilisée pour structurer et nettoyer les données, la bibliothèque Pandas a permis de traiter les données efficacement sous forme de tableaux.

Figure 3.2 – Logo de Pandas

- **Bibliothèque "re" (expressions régulières)** : L'utilisation d'expressions régulières a été cruciale pour extraire les informations des fichiers PDF et Word, permettant d'isoler les éléments textuels essentiels tout en éliminant les métadonnées et autres informations non pertinentes.

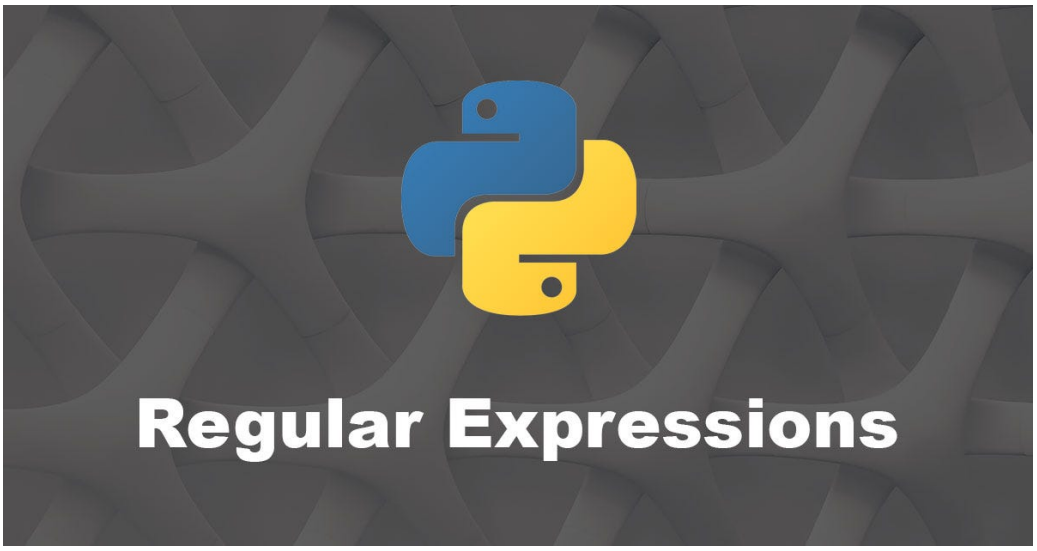

Figure 3.3 – Illustration d'expressions régulières

Ces outils ont assuré une collecte de données optimisée et prête pour le prétraitement.

3.2 Prétraitement des Données

Une fois les données collectées, un prétraitement rigoureux a été effectué pour garantir leur qualité et leur cohérence, condition indispensable pour leur exploitation dans l'entraînement du modèle.

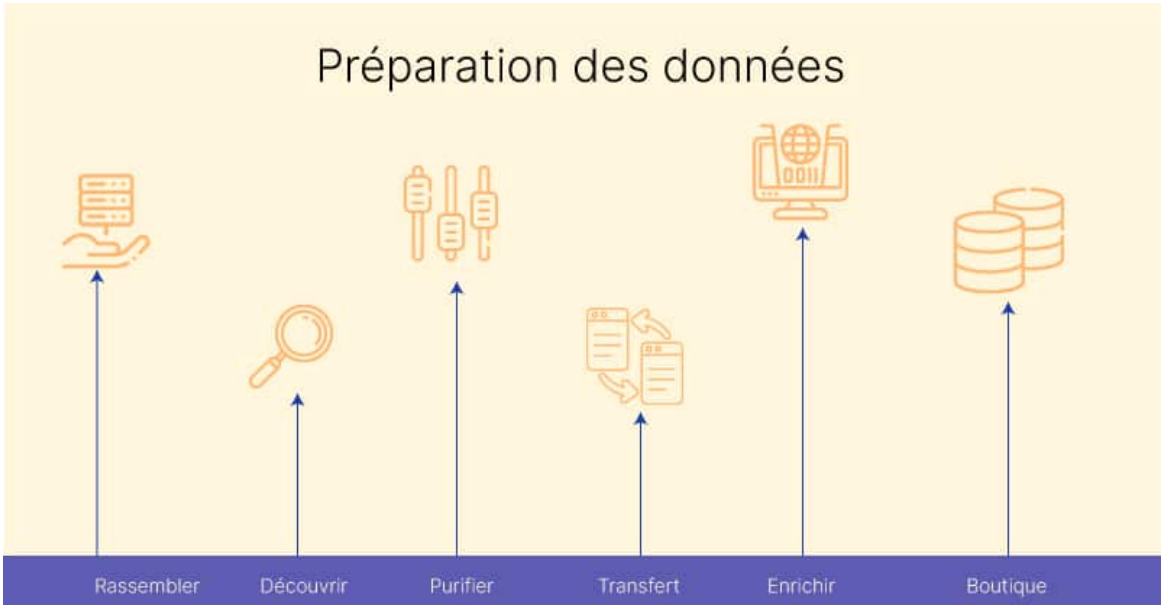

Figure 3.4 – Schéma du processus de préparation des données

Nettoyage des Données

Les étapes suivantes ont été mises en œuvre lors du nettoyage des données :

- **Suppression des doublons** : Afin d'éviter les redondances, les doublons ont été identifiés et supprimés.
- **Traitement des données manquantes** : Certaines informations étaient incomplètes et ont nécessité un traitement manuel ou leur exclusion.
- **Filtrage du bruit** : Les informations non pertinentes, telles que les métadonnées et éléments de mise en page, ont été éliminées.

Normalisation des Données

Pour assurer l'uniformité, une normalisation des données a été réalisée :

- **Uniformisation des formats** : Les textes récupérés ont été convertis dans un format uniforme, notamment les données issues des PDF et Word.
- **Harmonisation des structures** : Les données textuelles ont été restructurées pour garantir leur intégration homogène dans le processus de génération des questions et réponses.

3.3 Prompt Engineering pour la Génération des Questions et Réponses

Une étape clé du projet a été la génération de questions et réponses à partir des données collectées, en utilisant des techniques de Prompt Engineering avec le modèle GPT-4 et la méthode RAG (Retrieval-Augmented Generation). Cette approche a permis d'utiliser le texte brut extrait des différentes sources comme contexte pour générer des questions et réponses pertinentes.

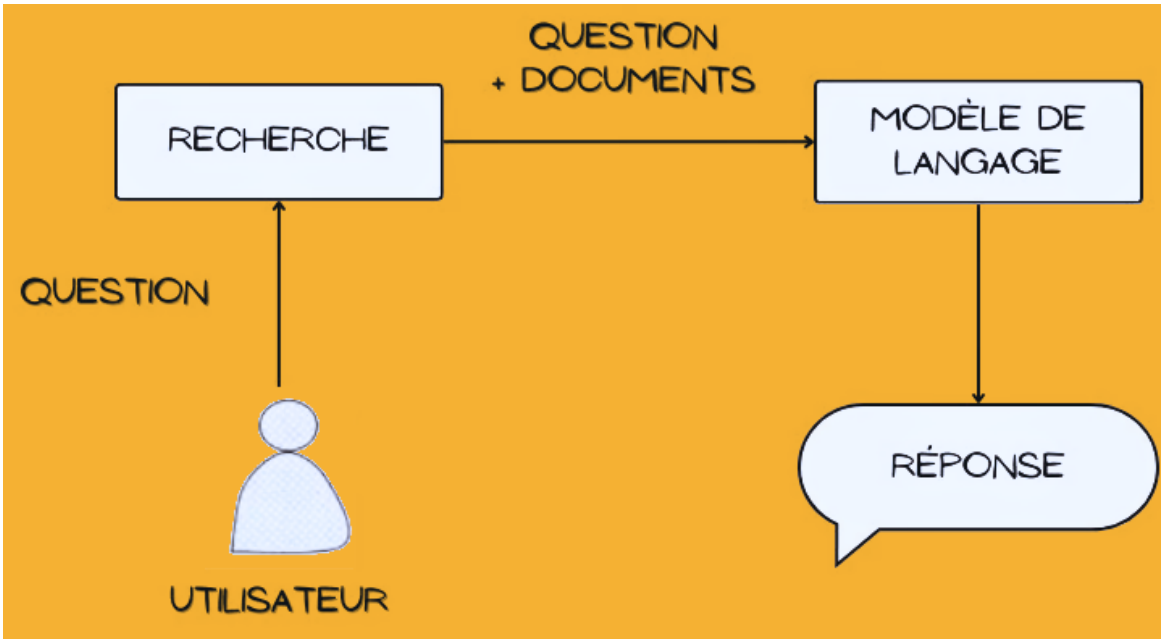

Figure 3.5 – Schéma du fonctionnement d'un système RAG

Utilisation du RAG pour la Génération de Questions

La première phase a consisté à générer automatiquement des questions à partir des données textuelles brutes. Grâce à l'approche RAG, le modèle a récupéré des informations pertinentes dans les documents collectés et les a utilisées pour formuler des questions.

Pour une meilleure structuration des questions, elles ont été générées par catégories à l'aide de prompts spécifiques. Voici un exemple de prompt utilisé :

> **Objectif** : Générer 60 questions pertinentes, spécifiques et représentatives que des clients potentiels pourraient poser à une entreprise de services, en se basant sur les informations fournies dans une description détaillée de l'entreprise et de ses offres.
>
> **Instructions** :
>
> - **Structuration des questions** : Répartir les questions en catégories principales, par exemple :
>   - Services offerts
>   - Produits
>   - Politique de confidentialité
>   - Support client
> - **Format et qualité des questions** :
>   - Rédiger avec un style clair, professionnel et comme si un vrai client s'adressait à l'entreprise.
>   - Éviter les questions trop générales, vagues ou impossibles à répondre convenablement.
>   - Chaque question doit être :
>   - Unique et originale
>   - Pertinente par rapport au contexte décrit
>   - Réaliste et représentative d'une réelle préoccupation client.
> - **Format de sortie** : Le résultat final doit être un fichier JSON nommé "questions.json" contenant un tableau de 60 objets questions, chaque objet ayant les clés suivantes :
>   - `"categorie"` : la catégorie de la question
>   - `"question"` : le texte de la question.

L'utilisation de ces prompts a permis de générer une série de questions couvrant divers aspects des services d'Inforisk, telles que les descriptions de produits, les options de service, et les préoccupations courantes des clients.

```
[17]: df_gpt.sample(15)
```

[17]:

	categorie	question
154	Service Ciblage	Comment garantissez-vous l'anonymat et la prot...
2219	Information sur les entreprises	Votre base de données permet-elle de connaître...
1946	Accompagnement et service client	L'efficacité de votre support client est-elle ...
2167	Expansion internationale	Vos outils de gestion du risque client sont-il...
1347	Informations internationales	Comment puis-je personnaliser mes rapports de ...
3259	Qualité de service	Quel est votre délai moyen de réponse aux dema...
973	Accompagnement et service client	Pouvez-vous expliquer comment INFORISK aide à ...
3186	Services proposés	Comment procédez-vous pour fiabiliser et mettr...
1035	Aspects juridiques et confidentialité	Quels types d'accords de confidentialité avez-...
1164	Études sectorielles et indicateurs économiques	Pourriez-vous fournir des rapports détaillés s...
3645	Évaluation de la satisfaction client	Quel est votre degré de satisfaction globale p...
1707	Sources de données	Les informations sur les dossiers de redressem...
889	Impact économique et social	Comment INFORISK peut-il aider une entreprise ...
360	Analytix	Quelle est la différence entre les rapports Ex...
2652	Tarification et support	Exposez-vous clairement ce qui est inclus ou n...

Figure 3.6 – Exemple de questions générées par catégorie

Génération des Réponses via RAG

Dans la deuxième phase, la même approche RAG a été utilisée pour générer des réponses aux questions produites. Le modèle a de nouveau récupéré les informations pertinentes dans le texte brut pour fournir des réponses informatives et cohérentes avec les services d'Inforisk.

Exemple de prompt pour générer une réponse :

> **Fournir une réponse claire et détaillée à la question suivante concernant les services d'Inforisk :** [insérer la question ici].

Cette approche a permis d'obtenir des réponses précises et adaptées au domaine, prêtes à être intégrées dans le chatbot pour répondre aux utilisateurs.

[26]:
```
df.sample(15)
```

[26]:

	categorie	question	reponse
2966	Processus d'enrichissement	Comment se déroule concrètement le processus d...	Inforisk reçoit votre fichier d'entreprises da...
2440	Qualité et exhaustivité des données	Quelle est l'antériorité des données et inform...	Inforisk capte annuellement 150.000 événements...
1652	Services et solutions Inforisk	Quels sont les principaux services proposés pa...	Inforisk propose des services de prospection c...
3217	Services proposés	Pouvez-vous donner des exemples de problématiq...	Inforisk réalise des études sur les défaillanc...
3609	Information sur les entreprises	Quel volume d'événements juridiques concernant...	Inforisk capte annuellement 150.000 événements...
4157	Service Consultation	Comment la plateforme Analytix aide-t-elle les...	La plateforme Analytix fournit aux analystes f...
4733	Sécurité et confidentialité	Comment assurez-vous la conformité des pratiqu...	Inforisk propose des outils de Risk Management...
599	Support client et assistance technique	Quel est le processus pour signaler une erreur...	Inforisk capte annuellement 150.000 événements...
4784	Support client et assistance technique	Comment puis-je organiser une session de forma...	Inforisk propose des sessions de formation per...
3995	Stratégie et positionnement	Quelles sont les valeurs fondamentales qui gui...	Inforisk se concentre sur la complétude, la fi...
5786	Services et solutions Inforisk	Quelles informations sont nécessaires pour qua...	Inforisk qualifie les bases de données en élim...
5078	Évaluation de la satisfaction client	Les clients utilisent-ils souvent vos études p...	Les clients utilisent fréquemment les études l...
4584	Services proposés	Comment puis-je vérifier la fiabilité des donn...	Inforisk propose une rubrique "Informations Ju...
2408	Qualité et exhaustivité des données	Combien de bilans d'entreprises avez-vous coll...	Inforisk a collecté 620.000 bilans d'entrepris...
1181	Études sectorielles et indicateurs économiques	Comment vos études peuvent-elles aider à optim...	Inforisk permet d'accéder à des outils d'analy...

Figure 3.7 – Exemple de réponses générées par catégorie

Affinage des Questions et Réponses

Le processus d'affinage a consisté en plusieurs étapes :

- **Révision des questions et réponses générées** : Les questions et réponses générées ont été analysées pour garantir leur cohérence et leur pertinence vis-à-vis des attentes des utilisateurs d'Inforisk.
- **Correction et reformulation si nécessaire** : Certaines réponses ont été reformulées pour mieux correspondre aux spécificités des services proposés par Inforisk.
- **Sélection des questions et réponses les plus pertinentes** : Seules les questions et réponses les plus pertinentes ont été retenues pour assurer la qualité et la pertinence du chatbot.

Utilisation de l'API GPT-4 pour la Génération Automatisée

Pour mettre en œuvre le modèle GPT-4 dans le cadre de la méthode RAG (Retrieval-Augmented Generation), nous avons utilisé la bibliothèque OpenAI, installée via la commande `pip install openai`. Cette bibliothèque permet d'interagir facilement avec l'API GPT-4 pour envoyer des prompts et recevoir des réponses.

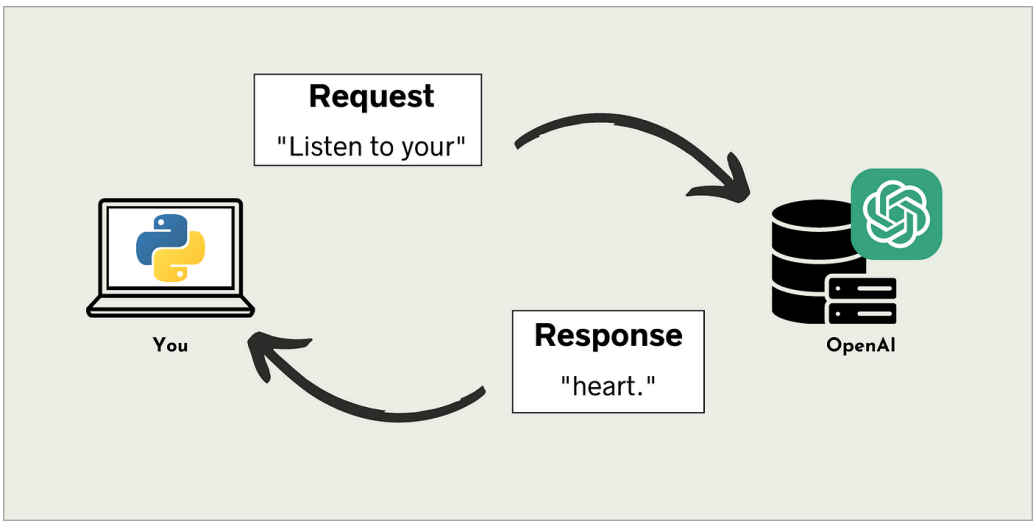

Figure 3.8 – Logo de l'API GPT-4

L'intégration de l'API s'est faite en plusieurs phases :

- **Configuration de l'API** : Après avoir obtenu une clé d'API d'OpenAI, nous avons utilisé la bibliothèque `open IA` en Python pour envoyer des requêtes à l'API, contenant le prompt ainsi que les paramètres d'ajustement tels que la température et le nombre maximum de tokens.
- **RAG pour la génération de questions** : Le modèle GPT-4 a été utilisé dans une approche RAG pour récupérer des informations contextuelles pertinentes à partir des données textuelles brutes collectées. Un prompt, conçu spécifiquement pour guider la génération de questions, a été envoyé à l'API. Par exemple, nous avons utilisé des prompts comme :

  > "Sur la base des informations disponibles sur les services et produits d'Inforisk, quelles questions fréquentes les clients pourraient-ils poser ?" "Quels aspects des services proposés par Inforisk doivent être clarifiés pour les clients ?"

  L'API renvoyait alors une série de questions générées automatiquement, que nous avons filtrées et affinées pour garantir leur pertinence.
- **RAG pour la génération de réponses** : Une fois les questions générées, un processus similaire a été mis en place pour générer des réponses. Un prompt contenant la question et le contexte associé a été envoyé à GPT-4 via l'API, avec une requête comme :

  > "Fournir une réponse claire et détaillée à la question suivante concernant les services d'Inforisk : [insérer la question ici]."

Le modèle GPT-4 a ensuite récupéré les informations pertinentes à partir des données textuelles pour produire une réponse cohérente et détaillée.

- **Affinage des résultats** : Après réception des réponses de l'API, nous avons effectué une révision manuelle pour assurer leur précision et pertinence vis-à-vis des besoins des utilisateurs d'Inforisk. Cela inclut des ajustements mineurs dans le wording et la structure pour s'adapter au ton et au style souhaités.

L'utilisation de l'API GPT-4 a ainsi permis de s'appuyer sur les capacités avancées du modèle pour générer des contenus question-réponse automatisés et contextuellement appropriés, contribuant à l'efficacité du chatbot.

3.4 Conclusion

La collecte et la préparation des données constituent une étape cruciale dans le développement d'un chatbot performant et pertinent. Grâce à une méthodologie rigoureuse, nous avons pu rassembler des informations issues de sources variées, les nettoyer, les structurer et les normaliser pour garantir leur cohérence. L'utilisation du modèle GPT-4, en particulier via une approche de génération augmentée par récupération (RAG), a permis de transformer ces données brutes en un ensemble de questions et réponses pertinentes, directement exploitables pour répondre aux utilisateurs d'Inforisk.

L'intégration de l'API OpenAI a considérablement facilité l'automatisation de ce processus, tout en offrant une flexibilité et une précision accrues dans la génération de contenu. Ces efforts de collecte et de traitement des données sont ainsi essentiels pour poser des bases solides à l'étape suivante : l'entraînement du modèle et son implémentation dans un environnement de production.

Chapitre 4

Développement et Fine-Tuning de Chatbot avec des Modèles de Langage

4.1 Introduction aux Modèles de Langage (LLM)

Les modèles de langage (LLM) sont des systèmes d'intelligence artificielle conçus pour traiter et générer du langage naturel. Ils sont couramment utilisés dans la génération de texte, les chatbots, la traduction automatique et d'autres tâches complexes liées au traitement du langage naturel (NLP). Dans cette étude, nous examinons trois modèles de pointe : Mistral NeMo, Llama 3 (7B), et GPT-4. Chaque modèle a des capacités spécifiques en termes de performance, de coût et de flexibilité pour le fine-tuning.

4.2 Comparaison entre Mistral NeMo, Llama 3 (7B) et GPT-4

Critères	Mistral NeMo	Llama 3 (7B)	GPT-4
Paramètres	12 milliards	7 milliards	Estimé à plus de 175 milliards
Performance	Surpasse de nombreux benchmarks	Compétitif mais inférieur à Mistral NeMo	Performance élevée
Fenêtre de Contexte	128 000 tokens	Fenêtre de contexte standard	Jusqu'à 8 192 tokens
Qualité	Sorties de haute qualité	Bonne qualité, moins robuste	Sorties de la plus haute qualité
Prix (entrée)	0,15 $ / 1M tokens	Nécessite une réservation de GPU	3,750 $ / 1M tokens
Prix (sortie)	0,15 $ / 1M tokens	Nécessite une réservation de GPU	15,000 $ / 1M tokens
Cas d'Utilisation	Adapté aux entreprises	Polyvalent mais limité	Applications variées
Fine-tuning	Oui, directement chez Mistral	Oui, open source	Oui, possibilité de fine-tuning
Open Source	Oui	Oui	Non

Table 4.1 – Comparaison entre Mistral NeMo, Llama 3 (7B) et GPT-4

4.3 Prix pour le Fine-Tuning et le Déploiement

Modèle	Coût d'Entraînement Unique	Stockage Mensuel	Entrée	Sortie
Mistral NeMo	1,000 $ / 1M tokens	2 $ par mois par modèle	0,15 $ / 1M tokens	0,15 $ / 1M tokens
Llama 3 (7B)	Nécessite réservation de GPU	Dépend de l'infrastructure	Nécessite réservation de GPU	Nécessite réservation de GPU
GPT-4	25,000 $ / 1M tokens	Gratuit	3,750 $ / 1M tokens	15,000 $ / 1M tokens

Table 4.2 – Prix pour le Fine-Tuning et le Déploiement des Modèles

4.4 Résumé des Conclusions

Mistral NeMo : Un modèle puissant avec 12 milliards de paramètres, excellent pour les tâches multilingues. Sa tarification compétitive et sa flexibilité le rendent idéal pour les entreprises.

Llama 3 (7B) : Open-source, mais moins performant pour des applications complexes, nécessitant une réservation de GPU.

GPT-4 : Modèle extrêmement performant, adapté aux tâches complexes mais coûteux.

4.5 Justification du Choix du Modèle de Mistral IA (mistral NeMo)

Figure 4.1 – Logo de Mistral

Performance et Capacité

Mistral NeMo : Excellent pour les tâches multilingues, adapté aux entreprises. Llama 3 (7B) : Moins performant pour des tâches exigeant une précision élevée. GPT-4 : Excelle dans les tâches complexes, mais coûteux.

Coût d'Utilisation

Mistral NeMo : Tarification compétitive. Llama 3 (7B) : Coûts imprévus liés à l'utilisation de GPU. GPT-4 : Coûts élevés pour le fine-tuning et l'utilisation.

Flexibilité et Fine-Tuning

Mistral NeMo : Permet un fine-tuning facile avec déploiement direct. Llama 3 (7B) : Pose des défis sans infrastructure de déploiement facile. GPT-4 : Options de fine-tuning disponibles, mais coûteuses.

4.6 Processus de Fine-Tuning sur Mistral NeMo

Introduction

Le fine-tuning permet d'adapter un modèle pré-entraîné à des tâches spécifiques. Nous décrirons les étapes pour effectuer le fine-tuning sur la plateforme Mistral NeMo.

Préparation des Données

Les données doivent être au format JSONL (JSON Lines) :

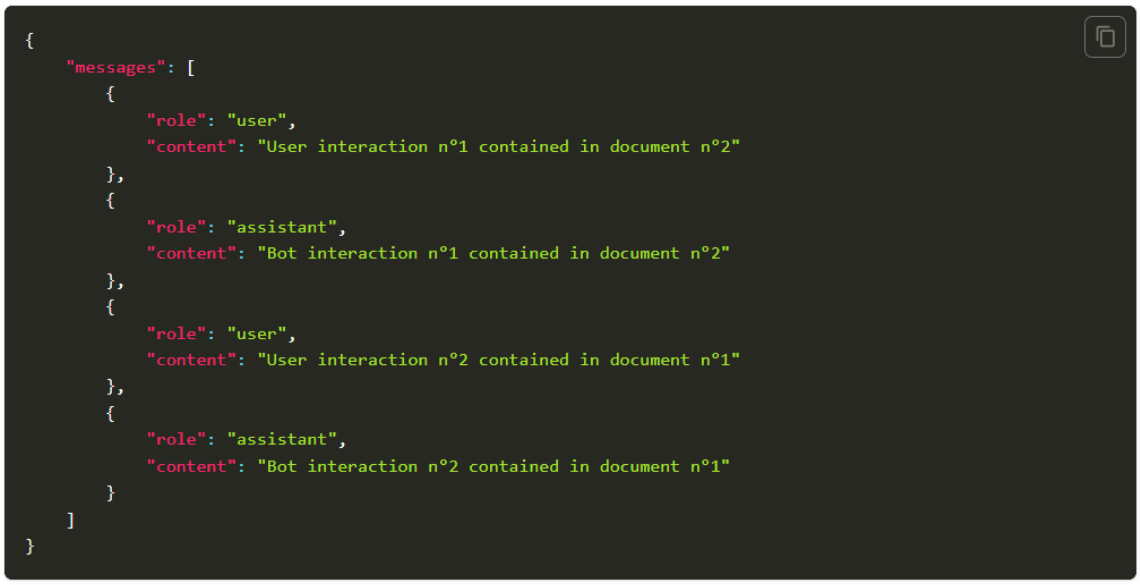

Figure 4.2 – Format des Données

Téléchargement des Données

Les fichiers JSONL sont téléchargés sur la plateforme Mistral NeMo.

```python

from mistralai import Mistral
import os

api_key = os.environ["MISTRAL_API_KEY"]

client = Mistral(api_key=api_key)

training_data = client.files.upload(
    file={
        "file_name": "ultrachat_chunk_train.jsonl",
        "content": open("ultrachat_chunk_train.jsonl", "rb"),
    }
)
```

Figure 4.3 – Script pour Téléchargement des Données

Création du Job de Fine-Tuning

Définir les paramètres tels que : Modèle à affiner Fichiers d'entraînement et de validation Hyperparamètres (par exemple, taux d'apprentissage : 0.0001, nombre d'étapes : 10)

```python


# create a fine-tuning job
created_jobs = client.fine_tuning.jobs.create(
    model="open-mistral-nemo-2407",
    training_files=[{"file_id": ultrachat_chunk_train.id, "weight": 1}],
    validation_files=[ultrachat_chunk_eval.id],
    hyperparameters={
        "training_steps": 10,
        "learning_rate":0.0001
    },
    auto_start=False
)

# start a fine-tuning job
client.fine_tuning.jobs.start(job_id = created_jobs.id)

created_jobs
```

Figure 4.4 – Code de Création du Job de Fine-Tuning

Suivi et Gestion des Jobs

Suivi de l'état, progression du job, ou annulation en cas d'erreurs.

```
# List jobs
jobs = client.fine_tuning.jobs.list()
print(jobs)

# Retrieve a jobs
retrieved_jobs = client.fine_tuning.jobs.get(job_id = created_jobs.id)
print(retrieved_jobs)

# Cancel a jobs
canceled_jobs = client.fine_tuning.jobs.cancel(job_id = created_jobs.id)
print(canceled_jobs)

```

Figure 4.5 – Code pour Afficher les Jobs

Utilisation du Modèle Affiné

Une fois qu'une tâche de fine-tuning est complétée, il est possible d'identifier le modèle affiné en consultant le champ retrieved_jobs.fine_tuned_model. Cela vous permettra de savoir quel modèle a été ajusté pour répondre à vos besoins spécifiques.

Pour interagir avec ce modèle affiné, vous pouvez utiliser notre point de terminaison de chat. Cette interface vous permet d'engager des discussions avec le modèle afin d'explorer ses capacités, de tester des réponses et de vérifier son efficacité dans différents scénarios d'utilisation.

```


chat_response = client.chat.complete(
    model=retrieved_job.fine_tuned_model,
    messages = [{"role":'user', "content":'quelle sont les services disponible chez inforisk'}]
)
```

Figure 4.6 – Code Utilisation du Modèle Affiné

4.7 Conclusion

Mistral NeMo se démarque par sa performance, sa capacité multilingue, et son coût compétitif, le rendant idéal pour les entreprises cherchant à développer des chatbots. Il offre une flexibilité et un processus de fine-tuning simple comparé à d'autres modèles comme Llama 3 (7B) et GPT-4. Bien que GPT-4 soit extrêmement performant pour des tâches complexes, ses coûts sont nettement plus élevés, tandis que Llama 3 (7B) présente des limites en termes de performance et d'infrastructure.

Chapitre 5

Intégration de Solution et Test

5.1 Introduction

Ce chapitre détaille l'intégration technique du chatbot IA développé pour la plateforme Analytix. L'objectif principal était de concevoir une interface utilisateur interactive et réactive, permettant une interaction fluide avec le modèle d'IA. Nous aborderons les technologies utilisées (HTML, CSS, JavaScript), l'intégration des API pour la communication utilisateur-modèle, la conception centrée sur l'utilisateur (UX/UI), et les méthodes de test pour un fonctionnement optimal.

5.2 Environnement de Développement

Le développement du chatbot sur Analytix s'est appuyé sur une combinaison de technologies clés :

- **HTML, CSS, JavaScript :** Trio fondamental pour l'interface utilisateur. HTML structure le contenu, CSS gère le style visuel, et JavaScript assure l'interactivité et la logique applicative, créant une interface dynamique et responsive.
- **API des Agents :** Intégrée pour une communication bidirectionnelle entre le chatbot et le modèle d'IA, gérant les requêtes utilisateurs et renvoyant les réponses générées en temps réel.

5.3 Intégration du Modèle IA

Méthode d'Intégration

L'intégration du modèle IA dans l'architecture du chatbot a nécessité les étapes suivantes :

- **Utilisation de l'API des agents :** L'API fournit des points de terminaison pour les requêtes de complétion, facilitant la communication fluide entre le chatbot et le modèle d'IA.

Gestion des Requêtes et des Réponses

L'interaction repose sur des requêtes JSON structurées, définissant les paramètres clés de la communication :

- **`max_tokens` :** Limite la longueur des réponses générées (en nombre de tokens) pour un contrôle précis.
- **`min_tokens` :** Définit la longueur minimale des réponses, garantissant des réponses suffisamment développées.
- **`stream` :** Active la réception des réponses en flux continu, offrant une expérience utilisateur plus interactive.
- **`stop` :** Permet de spécifier des tokens qui, lorsqu'ils sont générés, arrêtent la génération de la réponse.
- **`messages` :** Contient la requête utilisateur et le contexte de la conversation sous forme d'une liste d'objets JSON, chaque objet ayant un rôle ("user" ou "assistant") et un contenu textuel.
- **`response_format` :** Définit le format de sortie de la réponse, simplifiant le traitement par l'application.

5.4 Interface Utilisateur

Conception de l'Interface

L'interface utilisateur, cruciale pour l'expérience utilisateur, a été développée en tenant compte des bonnes pratiques UX/UI :

- **HTML :** Structure les éléments de la page : champs de saisie pour les questions, boutons d'envoi, zones d'affichage des réponses.

Figure 5.1 – HTML

- **CSS :** Style l'interface pour une esthétique professionnelle et une expérience utilisateur agréable. Utilisation de Flexbox et Grid pour une mise en page adaptative (responsive).

Figure 5.2 – CSS

- **JavaScript :** Gère toute l'interactivité : envoi des requêtes API, traitement des réponses, mise à jour dynamique du contenu sans rechargement de page.

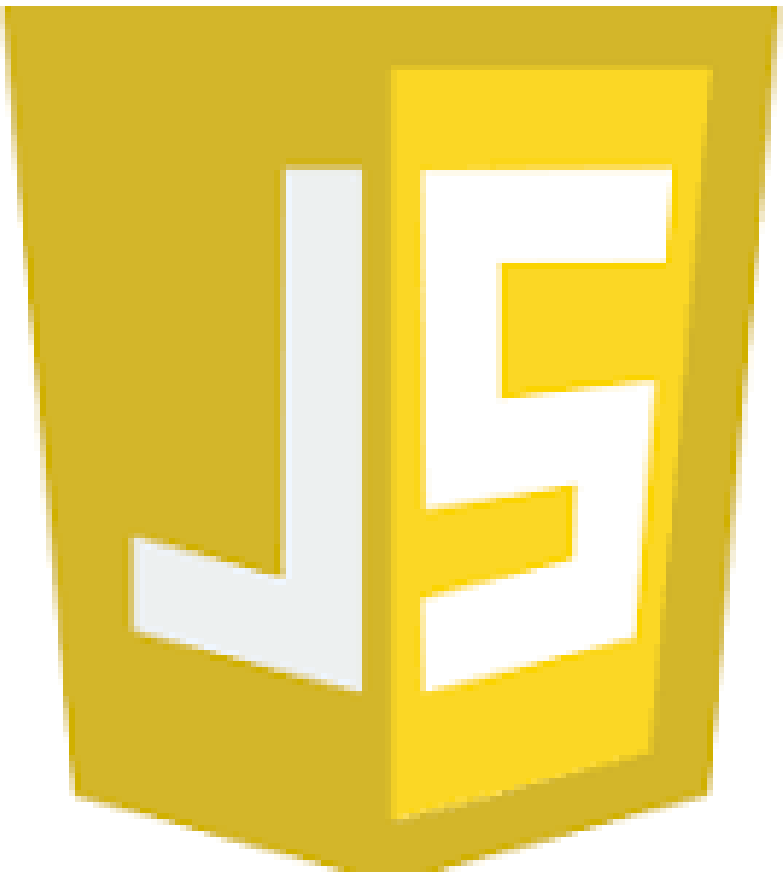

Figure 5.3 – JavaScript

Éléments Clés de l'UI

L'interface utilisateur s'articule autour d'éléments clés pour une interaction intuitive :

- **Liste d'options pour encadrer les questions des utilisateurs :** Permettre aux utilisateurs de choisir facilement leurs demandes après avoir saisi des questions au chatbot

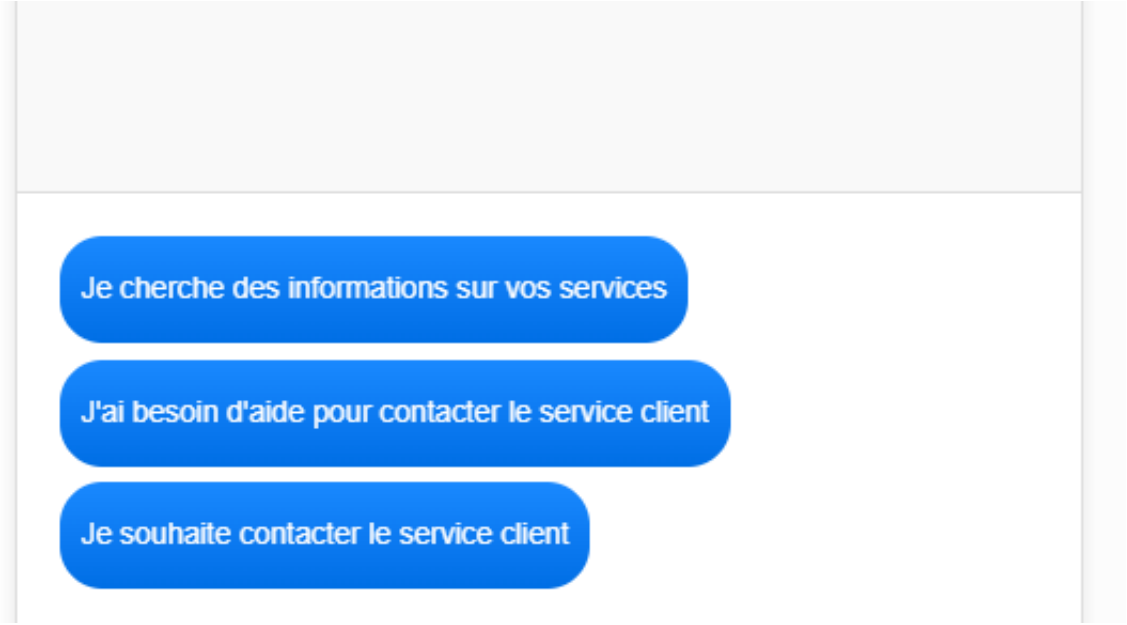

Figure 5.4 – Les Options Disponibile

- **Champs de saisie :** Permettent aux utilisateurs de saisir facilement leurs questions au chatbot.
- **Boutons interactifs :** Déclenchent l'envoi des requêtes au modèle d'IA pour traitement.
- **Affichage des réponses :** Zone dédiée à la présentation claire des réponses, utilisant un système de bulles de chat pour une simulation de conversation naturelle.

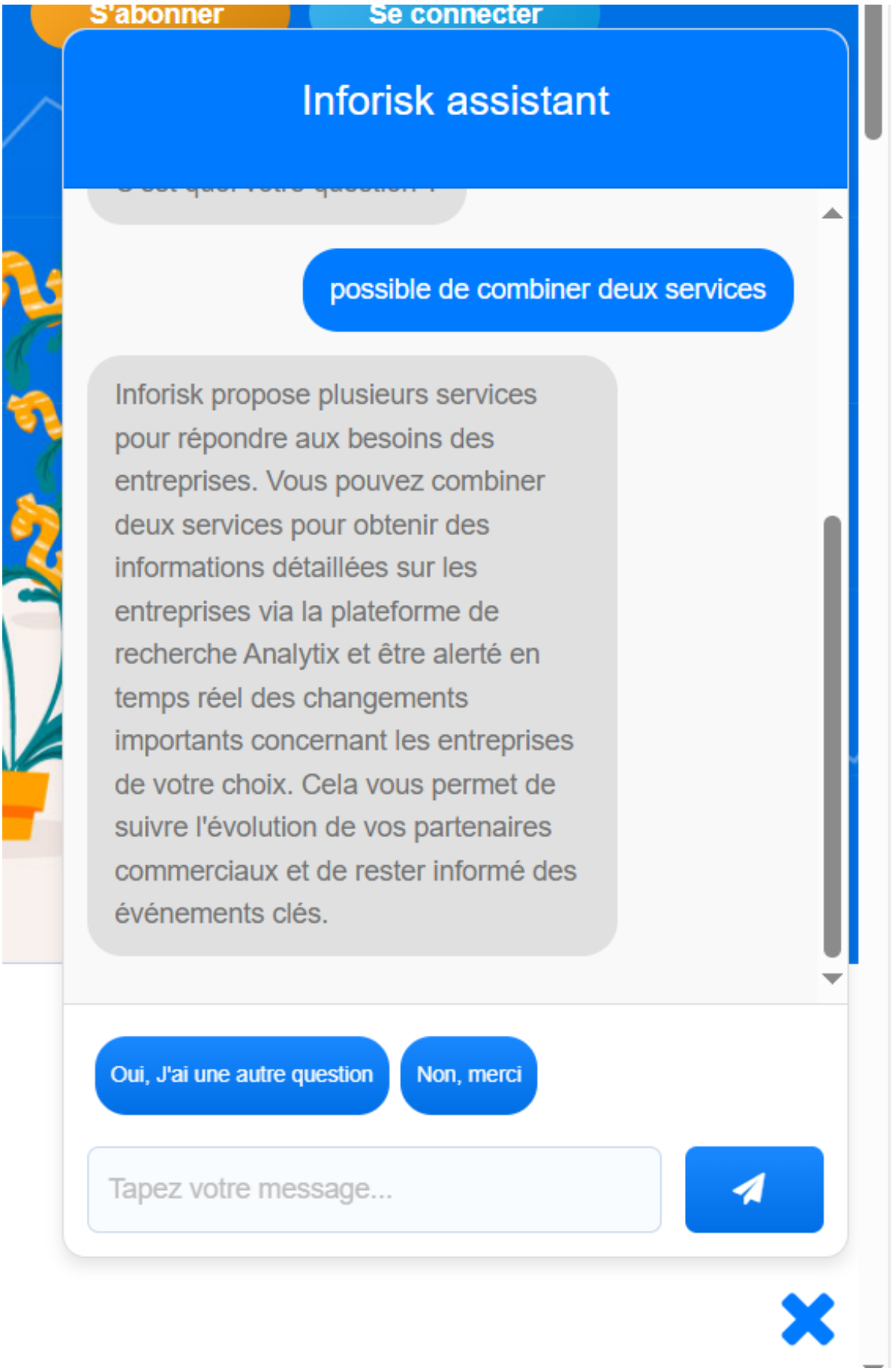

Figure 5.5 – La réponse

- **Chat bot dans la Plateforme :**

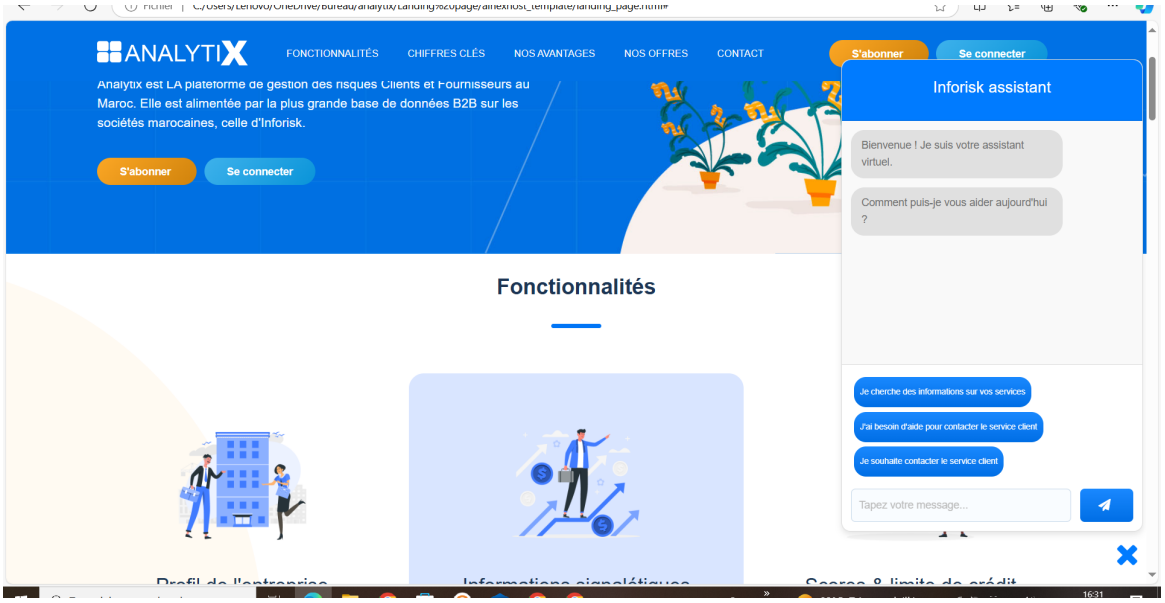

Figure 5.6 – Notre Composant Finale

5.5 API et Connectivité

Utilisation des API

L'API des agents est le cœur de la communication entre le chatbot et le modèle d'IA. Chaque requête envoyée via l'API est conçue pour obtenir des réponses précises et contextuelles en temps réel, en utilisant les paramètres décrits précédemment.

Gestion des Flux de Données

La gestion optimisée des flux de données est essentielle :

- **Requêtes API asynchrones :** Évitement du blocage de l'interface pendant le traitement des requêtes, garantissant une expérience utilisateur fluide, même en cas de temps de réponse variables.
- **Traitement des réponses :** Formatage et affichage des réponses dans l'interface de manière claire et lisible, en utilisant le format de sortie spécifié lors de la requête.

5.6 Tests et Validations

Méthodologie de Test

Le chatbot a été soumis à des tests intensifs via l'interface utilisateur d'Analytix. Des testeurs du support d'Inforisk ont interagi avec le chatbot en soumettant des requêtes variées, évaluant ainsi la pertinence et la précision des réponses générées.

Rôle des Testeurs du Support d'Inforisk

L'expertise technique des testeurs d'Inforisk a permis de valider la fluidité de l'interaction utilisateur et d'identifier les points à améliorer. Leur connaissance approfondie des processus métier et des cas d'utilisation spécifiques à Analytix a été inestimable.

Identification des Améliorations

Les tests ont permis d'identifier des axes d'amélioration, notamment la gestion des requêtes complexes nécessitant une compréhension contextuelle approfondie, et l'optimisation des performances globales du système pour des temps de réponse optimaux.

Validation du Fonctionnement

Grâce à cette phase de test rigoureuse menée en collaboration avec le support d'Inforisk, le chatbot a été validé pour sa robustesse fonctionnelle et sa capacité à offrir une expérience utilisateur optimale. Il a démontré sa capacité à répondre adéquatement à un large éventail de requêtes, confirmant ainsi la réussite de son intégration dans l'écosystème Analytix.

5.7 Conclusion

L'intégration du chatbot IA sur la plateforme Analytix a nécessité une approche soignée et structurée, combinant des technologies modernes et des méthodes de développement efficaces. L'utilisation de HTML, CSS, JavaScript et de l'API des agents a permis de créer une interface utilisateur réactive et intuitive, assurant une interaction fluide avec le modèle IA. Les tests effectués avec le support d'Inforisk ont permis de valider les performances et la robustesse du système, garantissant ainsi une expérience utilisateur optimale.

Conclusion Générale

Ce rapport de stage a permis de retracer l'intégralité du processus de développement et d'intégration d'un chatbot intelligent au sein de l'entreprise Inforisk. De la phase de benchmark des solutions existantes à la collecte et la préparation des données, en passant par le choix du modèle de langage Mistral NeMo et son fine-tuning, chaque étape a été soigneusement documentée. L'accent a été mis sur l'importance d'une conception centrée sur l'utilisateur (UX/UI) pour garantir une expérience utilisateur optimale. La phase de test, menée en collaboration avec le support d'Inforisk, a permis de valider le bon fonctionnement du chatbot et d'identifier des axes d'amélioration.

Ce projet illustre parfaitement comment l'intelligence artificielle, et plus précisément les chatbots, peuvent être mis au service des entreprises pour automatiser des tâches, améliorer l'expérience client et optimiser les processus internes. Le chatbot développé pour Inforisk représente une avancée significative dans sa transformation digitale et ouvre la voie à de futures innovations dans le domaine de l'IA conversationnelle.

Bibliographie

Ouvrages et Articles Scientifiques

* Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). *Attention is all you need*. Advances in neural information processing systems, 30.
Lien : `https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91pdf`

Ressources en Ligne

* OpenAI. (s.d.). *GPT-4*. Consulté le [date], sur `https://platform.openai.com/docs/models/gpt-4`
* Mistral AI. (s.d.). *Mistral NeMo*. Consulté le [date], sur `https://developer.nvidia.com/`
* Mistral AI. (s.d.). *Fine-tuning*. Consulté le [date], sur `https://docs.mistral.ai/guides/finetuning/`
* Mistral AI. (s.d.). *Technology*. Consulté le [date], sur `https://mistral.ai/fr/technology/`
* Meta AI. (s.d.). *Llama 3*. Consulté le [date], sur `https://ai.meta.com/llama/`
* BeautifulSoup. (s.d.). *Beautiful Soup Documentation*. Consulté le [date], sur `https://www.crummy.com/software/BeautifulSoup/bs4/doc/`
* The Pandas Development Team. (s.d.). *pandas documentation*. Consulté le [date], sur `https://pandas.pydata.org/pandas-docs/stable/`
* LangChain. (s.d.) *LangChain Documentation*. Consulté le [date], sur `https://www.langchain.com/`

Documents d'Entreprise

* Inforisk. (s.d.). *[Site web]*. sur `https://www.inforisk.ma/servlet/EspaceVisiteurServlet`.
* Inforisk. (s.d.). *[Presentation des services]*. Document interne non publié.