



Chatbot Juridique – Droit du Travail (RAG)

1. 🚀 Introduction et Contexte

Avec l'essor des **LLM**, les systèmes de question-réponse se développent rapidement, mais peuvent générer des informations incorrectes (hallucinations), surtout dans des domaines sensibles comme le droit.

Ce projet propose un **chatbot juridique en droit du travail** basé sur **RAG**, utilisant exclusivement le **Code du travail français** (Légifrance) pour fournir des réponses fiables et indiquer clairement quand une question est hors périmètre.

2. Objectifs du projet

Ce projet a pour objectif principal de concevoir et de mettre en œuvre un **chatbot juridique spécialisé en droit du travail**, capable de répondre à des questions simples en s'appuyant sur des sources légales fiables.

Les objectifs du projet peuvent être décomposés en deux catégories : **objectifs pédagogiques** et **objectifs techniques**.

2.1 Objectifs pédagogiques

Les objectifs pédagogiques de ce projet sont les suivants :

- Comprendre les principes fondamentaux du **Traitement Automatique des Langues (TAL / NLP)** appliqués à des documents juridiques.
- Mettre en œuvre un **pipeline RAG (Retrieval-Augmented Generation)** afin de réduire les hallucinations des modèles de langage.
- Manipuler des documents juridiques réels et volumineux, tels que le **Code du travail français**.
- Appréhender les limites des modèles de langage dans un contexte juridique sensible.

- Développer une approche rigoureuse et explicable pour la génération de réponses automatiques.

2.2 Objectifs techniques

Sur le plan technique, le projet vise à :

- Extraire et prétraiter le texte à partir de documents PDF juridiques.
- Segmenter les documents en passages (chunks) exploitables pour la recherche sémantique.
- Calculer des **embeddings** textuels à l'aide de modèles de type Sentence-BERT.
- Indexer les embeddings dans une base vectorielle (**FAISS**) afin de permettre une recherche efficace des passages pertinents.
- Intégrer un **modèle de langage (LLM)** pour générer des réponses à partir des passages récupérés.
- Développer une **interface utilisateur interactive** permettant à l'utilisateur de dialoguer avec le chatbot.
- Gérer explicitement les cas hors périmètre en indiquant clairement lorsque la réponse n'est pas disponible dans le corpus utilisé.

3. Description du corpus (données utilisées)

Le corpus utilisé dans ce projet est constitué exclusivement du **Code du travail français**, issu du site officiel **Légifrance**. Ce document regroupe l'ensemble des dispositions légales régissant les relations entre employeurs et salariés en France.

Le choix d'un corpus unique et officiel vise à garantir la **fiabilité juridique** des réponses fournies par le chatbot. Le document est fourni au format PDF et contient des articles structurés, mais également des sections non directement exploitables (sommaires, titres, pages blanches), nécessitant un prétraitement adapté.

L'utilisation du Code du travail comme unique source permet également de **délimiter clairement le périmètre fonctionnel** du chatbot et de gérer efficacement les questions hors domaine.

4. Architecture générale du système

Le système développé repose sur une architecture **RAG (Retrieval-Augmented Generation)**, combinant une phase de recherche d'information et une phase de génération de texte.

L'architecture globale du système se compose des modules suivants :

- un module d'extraction et de prétraitement des données,
- un module de calcul des embeddings et d'indexation vectorielle,
- un module de recherche sémantique (retrieval),
- un module de génération de réponses basé sur un modèle de langage,
- une interface utilisateur interactive.

Cette architecture permet de séparer clairement les responsabilités de chaque composant et d'améliorer la robustesse du système face aux hallucinations.

5. Prétraitement des données

Le prétraitement constitue une étape essentielle du pipeline. Le texte est d'abord extrait à partir du fichier PDF du Code du travail à l'aide d'une bibliothèque dédiée à la lecture de documents PDF.

Ensuite, un nettoyage est effectué afin d'éliminer les passages vides ou non pertinents. Le texte est ensuite segmenté en fragments appelés *chunks*, de taille fixe, avec un chevauchement partiel afin de préserver le contexte sémantique entre les segments.

Cette segmentation permet de rendre le corpus exploitable pour le calcul des embeddings et la recherche sémantique ultérieure.

6. Indexation et mécanisme de recherche (Retrieval)

Chaque chunk est transformé en un vecteur numérique (*embedding*) à l'aide d'un modèle de type **Sentence-BERT**. Ces vecteurs capturent la similarité sémantique entre les textes.

Les embeddings sont ensuite indexés dans une base vectorielle **FAISS**, permettant une recherche rapide et efficace des passages les plus pertinents par rapport à une question utilisateur.

Lorsqu'une question est posée, celle-ci est également convertie en embedding, puis comparée aux embeddings du corpus afin de récupérer les passages les plus proches sémantiquement.

7. Modèle de langage et génération des réponses

La génération des réponses repose sur un **modèle de langage de type LLM**, utilisé dans un cadre contrôlé. Les passages récupérés lors de la phase de retrieval sont injectés dans un prompt structuré.

Le modèle est explicitement contraint à :

- utiliser uniquement les informations fournies,
- ne pas inventer de contenu,
- répondre clairement « *Je ne sais pas* » lorsque la réponse n'est pas présente dans le corpus.

Cette approche permet de réduire significativement les hallucinations et d'améliorer la fiabilité des réponses produites.

8. Interface utilisateur

Une interface conversationnelle simple a été développée à l'aide de la bibliothèque **Gradio**. Cette interface permet à l'utilisateur de saisir une question juridique et d'obtenir une réponse générée par le chatbot.

L'interface est volontairement minimaliste afin de se concentrer sur la qualité des réponses et la démonstration du pipeline RAG. Elle est accessible via un lien public généré automatiquement lors de l'exécution du notebook.

9. Résultats et exemples de conversations

Plusieurs tests ont été réalisés afin d'évaluer le comportement du chatbot.

Chatbot juridique – Droit du travail (RAG)

Ce chatbot répond à des questions juridiques simples à partir du Code du travail.

question

Quels sont les droits du salarié en cas de licenciement ?

Clear Submit

Réponse du chatbot

En cas de licenciement, le salarié bénéficie de plusieurs droits prévus par le Code du travail. Tout d'abord, l'employeur doit respecter une procédure légale, comprenant la convocation à un entretien préalable, la tenue de cet entretien et la notification écrite du licenciement en précisant le motif.

Le salarié a également droit à un préavis, sauf en cas de faute grave ou lourde. Pendant cette période, il continue à percevoir sa rémunération.

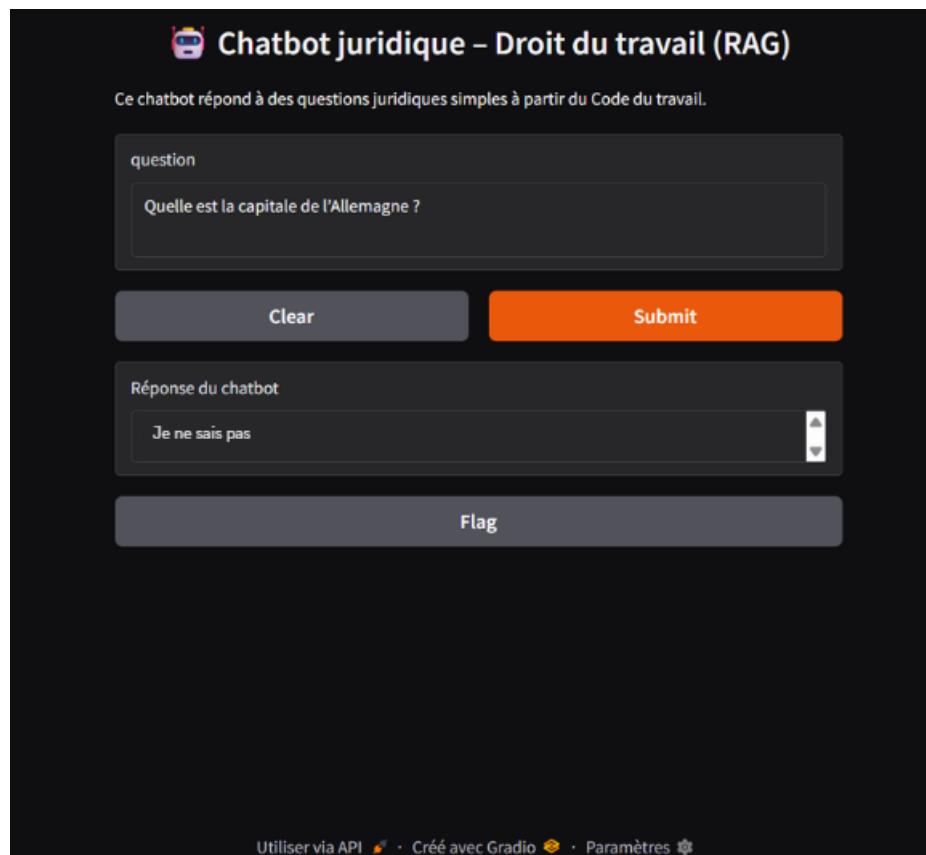
En cas de licenciement, le salarié peut percevoir une indemnité de licenciement, sous réserve de remplir les conditions d'ancienneté prévues par la loi.

Enfin, si le salarié estime que le licenciement est injustifié ou abusif, il peut saisir le Conseil de prud'hommes afin de contester la décision de l'employeur.

Flag

Utiliser via API · Créé avec Gradio · Paramètres

Pour des questions relevant du droit du travail, telles que « *Quels sont les droits du salarié en cas de licenciement ?* », le chatbot fournit des réponses cohérentes, basées sur les articles du Code du travail.



En revanche, pour des questions hors périmètre, par exemple « *Quelle est la capitale de l'Allemagne ?* », le système répond explicitement « *Je ne sais pas* », conformément aux contraintes imposées.

Ces résultats montrent que le pipeline RAG permet d'améliorer la pertinence et la fiabilité des réponses.

10. Limites du système

Malgré ses performances satisfaisantes, le système présente certaines limites :

- le chatbot dépend exclusivement du corpus utilisé et ne peut répondre à des questions hors du Code du travail,
- l'absence de métadonnées détaillées (numéro exact d'article, livre, titre) limite la précision des citations,
- les performances peuvent être affectées par la qualité de l'extraction du texte depuis le PDF.

11. Pistes d'amélioration

Plusieurs améliorations peuvent être envisagées :

- intégrer des sources complémentaires, telles que des guides CNIL ou de la jurisprudence,
- enrichir les chunks avec des métadonnées juridiques précises,
- mettre en place un mécanisme de re-ranking des passages,
- améliorer l'interface utilisateur en affichant explicitement les sources utilisées.

12. Conclusion

Ce projet a permis de concevoir un chatbot juridique fonctionnel basé sur une architecture RAG. L'utilisation exclusive du Code du travail comme source garantit la fiabilité des réponses et limite les risques d'hallucination.

Les résultats obtenus démontrent l'intérêt de l'approche RAG pour des applications sensibles telles que le droit. Ce travail constitue une base solide pour le développement de systèmes d'assistance juridique plus avancés.

13. Lien vers le dépôt GitHub

L'ensemble du code source du projet est disponible sur le dépôt GitHub suivant :

<https://github.com/adem0204/Chatbot-juridique>