

# MOHAMED JABER

## DEVELOPPEUR PYTHON



06 48 30 45 64



jaber.mohamed@outlook.fr



5 Boulevard Archimède 77420 Champs-sur-Marne

## EXPÉRIENCES PROFESSIONNELLES

### Enseignant Vacataire

Université Sorbonne Paris Nord : Octobre 2024 - Aujourd'hui

- Formation pratique et vulgarisée au Prompt Engineering pour des étudiants en économie et gestion
- Conception et animation d'un module d'initiation au prompt engineering appliqué à la gestion et à la bureautique.
- Transmission des bases du fonctionnement des modèles de langage (LLM) : tokens, contexte, raisonnement probabiliste.
- Présentation et mise en pratique des principaux types de prompts : zero-shot, one-shot, few-shot, role prompting, chain of thought et instruction explicite.
- Approche pédagogique centrée sur la vulgarisation, l'expérimentation et la compréhension critique de l'IA générative.
- Initiation à la rédaction de prompts efficaces pour automatiser des tâches courantes : création de documents, gestion de données sur Excel et résolution de problèmes professionnels
- Formation à la conception de prompts structurés pour automatiser la rédaction et la mise en page de documents professionnels (ex. rapport de stage).
- Utilisation du LLM comme assistant méthodologique pour planifier, reformuler et organiser le contenu d'un document.
- Application concrète à la création de tableaux Excel, graphiques et synthèses automatisées à partir de données textuelles ou chiffrées.
- Introduction au prompting multimodal : génération d'images et de musique à des fins de présentation et de créativité.
- Développement de la rigueur rédactionnelle à travers les piliers du prompt efficace : contexte, requête précise, format attendu, exemple, niveau de détails, ton.

### Developpeur Python Gen AI

Entreprise Teamnet - Paris : Octobre 2021 - Aujourd'hui

#### Intégration avancée de modèles de langage

- Développement d'une API multimodale intégrant ChatGPT, Gemini, Grok, DeepSeek et Llama via Groq.
- Utilisation du function calling pour la météo, la description d'images et vidéos, le pilotage audio (volume, stations radio), et la gestion de l'heure système.
- Implémentation d'un système de prompt chaining (prompts appelant d'autres prompts complexes) simulant un raisonnement structuré.
- Développement d'une API interne avec FastAPI, Pydantic et asyncio, pour la gestion dynamique des modèles et des prompts.
- Réalisation d'un benchmark complet (latence, cohérence, pertinence humaine) des LLM utilisés.

#### Système RAG (Retrieval-Augmented Generation)

- Conception d'un pipeline RAG complet orchestré avec LangChain.
- Vectorisation avec OpenAI et stockage des embeddings dans ChromaDB.
- Recherche hybride combinant embeddings et BM25 pour les contenus techniques et code.
- Reranking des résultats avec Cohere pour maximiser la pertinence.
- Pipeline complet : reformulation → recherche vectorielle/BM25 → reranking → reformulation finale via OpenAI.
- Évaluation humaine de la pertinence et mesure des performances globales du pipeline.

#### API conversationnelle multimodale

- Développement d'une API REST multipart gérant les flux audio (WAV) et texte (JSON).
- Intégration avec un ESP32 équipé d'un micro et d'un haut-parleur (interface vocale distante).
- Traitement serveur de la reconnaissance vocale (STT) et synthèse vocale (TTS) via Google Speech, OpenAI et ElevenLabs.
- Gestion complète du pipeline côté API : réception audio → transcription → génération → renvoi du WAV et du JSON.

### Gestion des données et stockage

- Redis utilisé comme session store temporaire (48h) avant transfert vers Google BigQuery.
- Développement d'une API d'administration permettant de modifier dynamiquement le modèle ou le prompt.
- Envoi automatique des données de session et logs vers BigQuery pour analyse et archivage.
- Exploitation de BigQuery pour le suivi des performances et la gestion des métriques.

### Architecture distribuée et infrastructure

- Déploiement sur serveur Linux (Debian) avec Nginx en reverse proxy.
- Gestion des logs SQL (BigQuery) et NoSQL (Redis) pour équilibrer performance et évolutivité.
- Mise en place partielle de CI/CD avec GitHub Actions et scripts Bash personnalisés.
- Conteneurisation Docker des micro-services et veille technique sur Kubernetes.
- Suivi des performances via logs structurés et scripts de monitoring interne.

### Reconnaissance vocale et traitement du langage

- Intégration de Vosk, Whisper et Google Speech-to-Text pour la reconnaissance vocale.
- Détection d'intentions et analyse de sentiments via GPT.
- Gestion de la mémoire conversationnelle et du contexte utilisateur avec Redis.
- Synthèse vocale multivoix et multilingue via Google TTS et ElevenLabs.

## Data scientist (Stage)

Angers Technopole - Angers : Avril 2021 - Septembre 2021

### • Traitement des non-conformités pharmaceutiques

#### • Identification des concepts pertinents, éléments de contexte et patterns :

- Analyse linguistique des descriptions de non-conformités pour détecter les entités clés, causes racines et schémas récurrents à l'aide d'outils de traitement automatique du langage (TAL).
- Construction manuelle de listes de synonymes et de termes équivalents du domaine pharmaceutique pour renforcer la précision de l'analyse.
- Utilisation de TermStat pour repérer automatiquement les ngrams significatifs et les cooccurrences fréquentes.

#### • Normalisation, prétraitement et étiquetage du texte :

- Conception d'un pipeline complet de prétraitement textuel : nettoyage, uniformisation, correction orthographique, suppression des ponctuations inutiles, gestion des unités de mesure et des formats de date.

#### • Personnalisation du tokenizer SpaCy pour préserver les mots composés et les identifiants techniques (protocoles, numéros de lots, unités).

- Étiquetage automatique des entités (unités, marques, mesures, équipements) via des règles et patterns linguistiques dédiés.

#### • Application de techniques de réduction de bruit : gestion des stopwords, suppression des majuscules, harmonisation terminologique.

#### • Classification et modélisation machine learning :

- Transformation du texte en données numériques via TF-IDF et création de la matrice document-terme pour représenter les non-conformités.
- Mise en place d'un jeu d'apprentissage/test (80/20) pour évaluer les modèles supervisés.
- Entraînement et comparaison de plusieurs modèles de classification (supervisés et non supervisés) afin de prédire la classe d'une non-conformité à partir de sa description.
- Évaluation des performances selon la pertinence des regroupements et la qualité du prétraitement textuel.

## COMPÉTENCES TECHNIQUES

- **Langages et frameworks**
- **Python** : Expert, développement API REST, scripting avancé, asyncio, FastAPI, Pydantic
- **SQL / NoSQL** : Notions avancées pour gestion de bases relationnelles et vectorielles (ChromaDB)
- **Développement Web** : API REST multimodale, intégration d'ESP32 pour flux audio
- **Prompt Engineering et LLM** : Conception de prompts complexes, prompt chaining, role prompting, chain of thought, zero/one/few-shot
- **GenAI et IA multimodale** : Intégration et benchmark de modèles ChatGPT, Gemini, Grok, DeepSeek, Llama via Groq
- **RAG (Retrieval-Augmented Generation)** : Pipeline complet avec LangChain, embeddings OpenAI, recherche hybride (embeddings + BM25), reranking Cohere
- **STT / TTS** : Google Speech, ElevenLabs, OpenAI, traitement audio serveur → transcription → génération → restitution
- **Data / Analyse** : Création automatisée de rapports Excel, synthèses, graphiques à partir de données textuelles ou chiffrées
- **Vectorisation et NLP** : OpenAI embeddings, ChromaDB, BM25, reranking
- **DevOps / Outils de productivité** : Versioning Git, pipelines asynchrones, gestion de flux multimodaux
- **Compétences linguistiques**
- **Anglais** : Très bon niveau parlé et technique (C1), capable de lire, comprendre et rédiger des documentations techniques, articles scientifiques et prompts complexes.

## FORMATION

### Master traitement automatique des langues (TAL)

Université Sorbonne Paris Nord : 2019-2021 (Mention : Très bien)