



3D Smart Factory

Rapport du projet de stage

Génération Intelligente de Questions pour les Tests de Recrutement

Sous la supervision de

Mr. Thierry Bertin Gardelle

MMe. KHLIF Ibtihal

Réalisé par

AIT JEDDI Assia	ENSET Mohammedia	Master Systèmes Distribués et Intelligence Artificielle
ELFAKIR Chaimae	ENSET Mohammedia	Master Systèmes Distribués et Intelligence Artificielle
Hbaly Hanane	FST Fés	Master de science de données et systèmes intelligents
Hlal Ikram	FST Fés	Master de science de données et systèmes intelligents

Année Académique : 2023/2024

Remerciements

Nous tenons à exprimer nos sincères remerciements à toute l'équipe de 3D Smart Factory pour nous avoir donné l'opportunité de travailler sur ce projet enrichissant. Leur soutien, leur confiance et leurs conseils ont été inestimables tout au long de cette expérience. Nous remercions particulièrement nos encadrants pour leur disponibilité et leurs recommandations précieuses, ainsi que nos collègues pour leur collaboration et leur partage de connaissances. Cette expérience nous a permis de développer nos compétences et d'acquérir une meilleure compréhension des défis concrets de l'industrie.

Merci à tous pour votre soutien et vos encouragements tout au long de ce projet.

Chapitre 1 : Introduction et Contexte du Projet

1- Introduction

Dans un monde où la technologie façonne significativement le paysage des affaires, l'optimisation des processus de recrutement devient non seulement une nécessité, mais aussi un avantage compétitif crucial pour des entreprises innovantes comme 3D Smart Factory. Ce chapitre introductif explore en profondeur le contexte et les motivations qui sous-tendent le développement d'une plateforme de recrutement intelligent utilisant des technologies de pointe, telles que le traitement automatique du langage naturel (NLP) et les modèles linguistiques de grande envergure (LLMs).

1-1- Contexte du projet

3D Smart Factory, entreprise à la pointe de la technologie dans le secteur de la fabrication intelligente, est confrontée à des défis significatifs dans son processus de recrutement. Traditionnellement, ce processus implique une gestion intensive des CV, une correspondance manuelle des compétences avec les besoins spécifiques des postes à pourvoir, et une évaluation laborieuse des candidatures. Ces défis se traduisent souvent par des retards dans le recrutement, une inefficacité opérationnelle, et une capacité limitée à identifier rapidement et précisément les talents les plus adaptés.

Face à cette réalité, 3D Smart Factory reconnaît la nécessité impérieuse d'intégrer des solutions innovantes pour optimiser ses pratiques de recrutement. L'automatisation des processus de tri et d'évaluation des candidatures apparaît comme une stratégie clé pour réduire les délais, améliorer l'efficacité opérationnelle, et garantir une meilleure adéquation entre les talents et les postes à pourvoir.

1-2- Objectif du projet

Les objectifs principaux de ce projet sont articulés autour de la transformation et de l'optimisation du processus de recrutement chez 3D Smart Factory :

- **Simplification et Automatisation** : Automatiser les tâches chronophages et répétitives liées à la gestion des candidatures et à l'évaluation des CVs, permettant ainsi de libérer du temps pour les recruteurs et d'accélérer le processus global de recrutement.
- **Amélioration de l'Efficacité du Matching** : Utiliser des technologies avancées de NLP pour analyser en profondeur les compétences, les expériences et les qualifications des candidats, afin de recommander de manière proactive les offres d'emploi les plus pertinentes.
- **Optimisation de la Qualité des Embauches** : Faciliter une prise de décision plus informée et objective en fournissant aux recruteurs des outils analytiques basés sur des données probantes pour évaluer et comparer les profils des candidats.

En poursuivant ces objectifs, 3D Smart Factory cherche non seulement à rationaliser ses opérations de recrutement, mais aussi à renforcer sa capacité à attirer et à retenir les talents

les plus qualifiés. L'adoption de cette plateforme de recrutement intelligent représente un engagement stratégique envers l'innovation et la transformation numérique, visant à positionner l'entreprise en tant que leader dans son secteur.

Cette introduction établit un cadre approfondi pour comprendre les enjeux stratégiques et les aspirations de notre projet de plateforme de recrutement intelligent. Les chapitres suivants détailleront la méthodologie adoptée, l'architecture technologique mise en œuvre, les résultats obtenus ainsi que les perspectives d'évolution pour cette solution novatrice.

2- Technologies et concepts

2-1- Introduction au NLP (Traitement Automatique du Langage Naturel)

Le Traitement Automatique du Langage Naturel (NLP) est une sous-discipline de l'intelligence artificielle (IA) qui vise à permettre aux machines de comprendre, interpréter, et générer du langage humain de manière automatisée. Dans le cadre du projet de 3D Smart Factory, le NLP joue un rôle fondamental en automatisant l'analyse des CVs et la recommandation d'offres d'emploi, rendant possible le traitement rapide de grandes quantités de données textuelles complexes, autrefois fastidieux et coûteux. Le NLP permet de réaliser plusieurs tâches clés :

- **Extraction d'informations** : Identification et extraction automatique d'informations pertinentes à partir de documents textuels, tels que les CVs. Cela inclut l'extraction de compétences, d'expériences professionnelles, d'éducation, etc.
- **Classification et catégorisation** : Attribution automatique de tags ou de catégories à des documents en fonction de leur contenu. Par exemple, classer un CV dans une catégorie spécifique d'expérience professionnelle ou de domaine d'expertise.
- **Recommandation et matching** : Analyse sémantique pour faire correspondre les compétences et l'expérience d'un candidat avec les exigences spécifiques d'une offre d'emploi. Cela permet de recommander automatiquement les candidats les plus pertinents pour un poste donné.

2-2- Les LLMs (Large Language Models)

Les Large Language Models (LLMs) sont des modèles avancés d'IA qui ont révolutionné le domaine du NLP. En étant formés sur des corpus massifs de données textuelles, ces modèles acquièrent une compréhension profonde des structures linguistiques et des significations contextuelles. Voici les aspects clés des LLMs et leur importance dans l'automatisation des tâches NLP.

Importance des LLMs :

- **Capacité de génération de texte** : Les LLMs sont capables de générer du texte de manière autonome qui semble humainement plausible. Cela inclut la rédaction de résumés, la création de réponses automatiques, la rédaction d'articles, etc. Ces capacités sont particulièrement utiles dans des applications telles que la rédaction automatique de descriptions de poste ou de réponses standardisées aux candidats.

- **Analyse sémantique avancée** : Grâce à leur compréhension approfondie du langage naturel, les LLMs peuvent effectuer une analyse sémantique fine. Cela permet de comprendre non seulement le sens littéral des phrases, mais aussi leur contexte, leurs implications et leurs nuances. Dans le cadre de l'analyse des CVs, cela permet de détecter non seulement les mots clés, mais aussi leur pertinence et leur usage approprié.
- **Adaptabilité et généralisation** : Les LLMs sont conçus pour être généralistes et peuvent être adaptés à divers domaines et langues. Cela les rend extrêmement polyvalents pour des applications dans différents secteurs de l'industrie, y compris le recrutement et la gestion des ressources humaines.

Applications des LLMs dans le NLP :

- **Génération de contenu** : Création automatisée de contenu rédactionnel comme des descriptions de postes, des articles de blog, des réponses à des questions fréquentes, etc.
- **Traduction automatique et résumé** : Capacité à traduire automatiquement des textes d'une langue à une autre et à produire des résumés automatiques de documents longs.
- **Reconnaissance d'entités nommées** : Identification automatique de noms de personnes, d'organisations, de lieux, etc., à partir de textes non structurés.
- **Analyse sentimentale** : Évaluation automatique des sentiments exprimés dans un texte, que ce soit pour des commentaires sur les réseaux sociaux, des critiques de produits ou des feedbacks des employés.

Les LLMs représentent une avancée majeure dans le domaine du NLP, offrant des capacités inédites pour l'automatisation des tâches liées au traitement du langage naturel. Leur capacité à comprendre et à générer du langage humain avec précision ouvre la voie à de nombreuses applications innovantes, non seulement dans le domaine du recrutement mais également dans d'autres secteurs nécessitant une analyse textuelle complexe.

3- Objectifs personnels en tant que stagiaire dans le projet

Dans le cadre de notre participation à ce projet, plusieurs objectifs personnels ont été définis, visant à maximiser l'expérience et à apporter une contribution significative au développement de la plateforme. Ces objectifs sont axés sur l'acquisition de compétences en NLP, l'amélioration des modèles, et une implication directe dans le succès du projet. Voici les objectifs détaillés :

3-1- Acquisition de compétences spécifiques en NLP

Un des objectifs principaux est de renforcer les compétences en **Traitement Automatique du Langage Naturel (NLP)**, avec un focus sur l'application directe des concepts et des outils dans des environnements réels.

- **Maîtrise des techniques avancées de NLP** : Nous avons pour ambition d'approfondir nos connaissances des techniques de pointe, telles que l'extraction d'entités nommées, l'analyse sémantique des CVs, et le matching des compétences des candidats avec les offres d'emploi. Nous prévoyons d'étudier en profondeur l'intégration des Large Language Models (LLMs) et leur utilisation pour améliorer les processus de recommandation automatisée et d'analyse des CVs.
- **Expérience pratique avec les outils et frameworks** : Le projet a permis de développer une expertise dans des outils comme **PyPDF2** pour extraire des données textuelles à partir de fichiers PDF, ce qui a permis une meilleure analyse des CV, la bibliothèque **json** a été essentielle pour la gestion et le traitement des données au format JSON, **re** a été utilisé pour des opérations de correspondance de motifs et d'expressions régulières, notamment pour le nettoyage et la validation des données, **Groq** a été intégré pour des requêtes optimisées sur les bases de données, améliorant ainsi l'efficacité des opérations de recherche et de filtrage des données, Nous avons exploité **torch** et **transformers** de Hugging Face, notamment le modèle BERT, pour le traitement et l'analyse avancée du langage naturel, permettant une meilleure compréhension sémantique des CV, **TfidfVectorizer** de sklearn a été utilisé pour la vectorisation des textes, facilitant ainsi les calculs de similarité entre les CV et les offres d'emploi à l'aide de **cosine_similarity**, et aussi l'intégration de **bson** pour la gestion des ObjectId de MongoDB a été cruciale pour manipuler les identifiants des documents de manière efficace, ...
- **Application des connaissances théoriques** : Nous visons à appliquer les concepts théoriques appris dans des projets concrets, tels que l'optimisation des modèles pour des tâches spécifiques de traitement du langage naturel, l'amélioration de la précision des analyses sémantiques, et la mise en œuvre de pipelines de traitement de texte robustes.

3-2- Contribution au développement de la plateforme

En tant que membres de l'équipe, notre objectif est de contribuer activement au développement et à l'amélioration continue de la plateforme NLP. Cela comprend :

- **Développement de fonctionnalités avancées** : Nous nous engageons à collaborer étroitement avec les développeurs seniors pour concevoir, développer et déployer de nouvelles fonctionnalités avancées. Cela peut inclure l'intégration de nouvelles capacités de traitement du langage, l'amélioration de l'interface utilisateur pour une meilleure convivialité, ou encore l'automatisation de tâches récurrentes liées au NLP.
- **Optimisation des performances** : Nous sommes déterminés à optimiser les performances des modèles NLP existants en identifiant et en corrigeant les goulots d'étranglement potentiels. Cela implique la mise en œuvre de tests rigoureux, l'analyse des résultats et l'itération pour améliorer la précision, la vitesse et la scalabilité des modèles déployés.

- **Soutien à l'équipe** : Nous nous engageons à apporter un soutien continu à nos collègues, que ce soit par le biais de la documentation technique, de la formation interne sur les meilleures pratiques en matière de NLP, ou en participant activement aux sessions de révision de code et de brainstorming pour résoudre des défis complexes.

3-3- Objectifs à long terme

Dans une perspective de développement durable et d'amélioration continue de la plateforme, plusieurs objectifs à long terme ont été envisagés :

- **Intégration de fonctionnalités d'entretien en ligne** : Développer une interface permettant la réalisation d'entretiens en ligne automatisés, où les candidats peuvent passer des tests de compétences en temps réel, suivis par des analyses PNL pour évaluation des réponses fournies.
- **Système de feedback automatisé** : Créer un mécanisme de retour d'information pour les candidats, utilisant des modèles NLP pour générer des curriculum vitae personnalisés de leurs performances lors des tests, avec des recommandations pour améliorer leurs compétences.
- **Analyse prédictive des compétences** : Implémenter des modèles d'apprentissage automatique capable de prédire les compétences nécessaires pour des postes futurs en analysant les tendances du marché et les offres d'emploi, fournissant ainsi aux candidats des insights précieux pour leur développement professionnel.
- **Élargissement de l'intégration avec d'autres plateformes** : Établir des connexions avec des outils de gestion de la performance, des plateformes d'apprentissage en ligne, et d'autres systèmes de recrutement pour créer une solution intégrée qui soutient les candidats tout au long de leur parcours professionnel.
- **Création d'une communauté d'apprentissage** : Fonder une plateforme communautaire où les candidats peuvent échanger des ressources, participer à des sessions de formation en direct et recevoir des conseils d'experts en recrutement et en développement de carrière.

4- Conclusion

En conclusion, le projet de recrutement intelligent pour 3D Smart Factory représente une avancée significative dans l'optimisation des processus de sélection des candidats. Grâce à l'intégration des technologies NLP et LLMs, cette plateforme permettra d'améliorer l'efficacité et de réduire les délais de recrutement. 3D Smart Factory est ainsi prêt à relever les défis futurs en s'adaptant aux besoins du marché tout en garantissant l'attractivité de ses talents.

Chapitre 2 : Analyse Fonctionnelle et Technique de la Plateforme

1- Description de la plateforme

La plateforme de recrutement intelligent repose sur des technologies avancées pour automatiser et optimiser le processus de mise en relation entre les candidats et les recruteurs. Elle utilise le Traitement Automatique du Langage Naturel (NLP) et des modèles de grandes dimensions (LLMs) pour extraire des informations des CV et proposer des recommandations d'offres d'emploi.

Fonctionnalités principales :

- **Analyse automatique des CV** : Grâce à des algorithmes de NLP, la plateforme extrait les compétences, l'expérience et les qualifications à partir des CV soumis.
- **Recommandation d'offres d'emploi** : La plateforme compare les CV des candidats avec les descriptions d'offres d'emploi en utilisant des modèles de grande taille (LLMs) pour proposer les offres les plus pertinentes.
- **Système de test en ligne** : Les recruteurs peuvent générer automatiquement des tests personnalisés pour évaluer les compétences des candidats en fonction de leurs CV et des exigences des offres.
- **Tableau de bord pour recruteurs** : Un espace dédié aux recruteurs permet de publier des offres, suivre les candidatures, et évaluer les résultats des tests en ligne.

2- Architecture Technique

La plateforme utilise une architecture moderne avec des technologies performantes pour garantir la fluidité et la scalabilité du système.

2-1- Backend (Flask + Python)

Le backend de la plateforme est développé en Python avec le framework léger Flask. Flask est choisi pour sa simplicité et sa capacité à gérer efficacement les requêtes HTTP, tout en permettant une intégration aisée avec des modèles d'apprentissage automatique (NLP et LLMs).

- **API RESTful** : Flask permet de créer des API REST qui reçoivent les données (CV, descriptions de postes, etc.) et renvoient les résultats de l'analyse et des recommandations d'offres.
- **Traitement NLP** : Les données textuelles soumises (CV et offres d'emploi) sont traitées à l'aide de bibliothèques Python populaires telles que spaCy et Hugging Face. Ces

bibliothèques permettent d'extraire les entités nommées (compétences, expériences, diplômes) et d'encoder les données textuelles en vecteurs.

- **LLMs** : Des modèles comme BERT ou GPT sont utilisés pour convertir les CV et les descriptions d'offres en vecteurs afin de mesurer la similarité sémantique entre les deux.

2-2- Frontend (React.js)

L'interface utilisateur est construite avec React.js, une bibliothèque JavaScript populaire pour créer des interfaces utilisateur dynamiques et réactives.

- **Gestion des utilisateurs** : Les candidats peuvent soumettre leurs CV, recevoir des recommandations et consulter leurs offres personnalisées. Les recruteurs, quant à eux, peuvent gérer leurs offres d'emploi et suivre les résultats des tests.
- **Communication avec l'API** : L'interface React.js interagit avec l'API Flask pour envoyer et recevoir des données en temps réel. L'affichage des recommandations et des tests est mis à jour dynamiquement en fonction des interactions des utilisateurs.

2-3- Base de données (MongoDB)

La base de données NoSQL MongoDB est utilisée pour stocker les données des CV, des offres d'emploi, et des résultats de tests. MongoDB est choisi pour sa flexibilité à gérer des données semi-structurées et volumineuses.

- **Stockage des CV** : Les CV soumis par les candidats sont stockés sous forme de documents JSON, permettant une manipulation et une récupération rapides.
- **Stockage des offres d'emploi** : Chaque offre d'emploi comprend des informations telles que le nom de l'entreprise, la description du poste, les compétences requises, et est également stockée en JSON pour permettre une comparaison efficace avec les CV.

3- Processus de traitement des données

1. Soumission et extraction des informations des CV

- Lorsque le candidat soumet un CV (généralement au format PDF ou texte), le backend Flask récupère le fichier et applique des techniques de traitement automatique du langage pour extraire les informations importantes. Les entités telles que les compétences, les expériences professionnelles et les qualifications sont identifiées à l'aide de modèles NLP.

2. Encodage et matching avec les offres d'emploi

- Les informations extraites du CV sont encodées sous forme de vecteurs numériques à l'aide de modèles de langue pré-entraînés (LLMs) tels que BERT. Ces vecteurs permettent de représenter les informations sémantiques du CV.

- De manière similaire, les descriptions des offres d'emploi sont également encodées en vecteurs. Ensuite, une méthode de similarité cosinus est utilisée pour mesurer la correspondance entre le CV et les offres d'emploi. Les offres les plus pertinentes sont renvoyées comme suggestions au candidat.

3. Recommandation d'offres et interaction avec les recruteurs

- Les résultats du matching entre les CV et les offres sont envoyés au frontend React.js, où ils sont affichés aux utilisateurs (candidats).

- Les recruteurs peuvent également visualiser les profils des candidats et générer des tests en ligne personnalisés via leur tableau de bord.

4. Génération automatique de tests

- En fonction des compétences identifiées dans le CV et des exigences des offres d'emploi, la plateforme génère des tests personnalisés que le candidat doit passer. Les résultats des tests sont ensuite stockés et peuvent être visualisés par les recruteurs sur leur tableau de bord.

4- Flux de données dans le système

Voici une vue d'ensemble du processus et du flux de données au sein de la plateforme :

1. Soumission du CV par le candidat :

- Le candidat utilise l'interface React.js pour soumettre son CV. Ce CV est envoyé à l'API Flask pour être analysé.

- Le backend Flask applique des techniques NLP pour extraire les informations pertinentes (compétences, expériences, diplômes).

2. Stockage des données dans MongoDB :

- Les données extraites du CV sont stockées dans MongoDB. Chaque CV est transformé en un document JSON qui sera utilisé pour des recherches ultérieures.

- De la même manière, les offres d'emploi sont stockées sous forme de documents JSON dans MongoDB.

3. Matching et recommandation d'offres d'emploi :

- Flask encode les CV et les offres d'emploi en vecteurs via des modèles LLM comme BERT. Une similarité cosinus est ensuite calculée pour trouver les meilleures correspondances.

- Les offres pertinentes sont envoyées à l'interface React.js et affichées comme recommandations pour le candidat.

4. Interaction recruteur-candidat :

- Les recruteurs publient des offres et peuvent visualiser les candidatures via le tableau de bord React.js. Ils peuvent aussi générer des tests adaptés aux compétences des candidats et consulter les résultats des tests directement sur la plateforme.

5. Suivi et évaluation des candidats :

- Les recruteurs peuvent suivre les statistiques sur les candidats (nombre de candidatures, résultats des tests) et prendre des décisions basées sur les performances.

5- Conclusion

Ce chapitre décrit en détail les aspects fonctionnels et techniques de la plateforme de recrutement intelligent. Le choix de Flask pour le backend permet une gestion flexible des requêtes et l'intégration des modèles NLP et LLMs pour l'analyse des CV et la recommandation d'offres. React.js pour le frontend offre une interface utilisateur fluide et dynamique, tandis que MongoDB assure un stockage efficace des données complexes et semi-structurées.

Chapitre 3 : Implémentation et Résultats

1- Implémentation des Modèles NLP et LLM

L'implémentation des modèles NLP et LLM dans la plateforme de recrutement intelligent a été réalisée en plusieurs étapes clés, intégrant divers outils et bibliothèques pour assurer une analyse efficace des CV et une recommandation précise des offres d'emploi.

1-2- Intégration des Modèles NLP et LLM

Pour traiter les CV et les descriptions d'offres d'emploi, nous avons utilisé des modèles avancés tels que BERT (Bidirectional Encoder Representations from Transformers) et LLaMA (Large Language Model Meta AI). Ces modèles permettent une compréhension sémantique approfondie des textes, facilitant ainsi le matching entre les compétences des candidats et les exigences des postes.

a. Extraction et Encodage des Données

1. Extraction du Texte des CV :

- Utilisation de **PyPDF2** pour extraire le texte des fichiers PDF soumis par les candidats.
- Fonction `extract_text_from_pdf` permet de lire chaque page du PDF et de compiler le texte brut.

2. Parsing des CV :

- La fonction `parse_resume` utilise l'API **Groq** pour parser le texte extrait et structurer les informations dans un format JSON prédéfini.
- Le JSON extrait contient des informations essentielles telles que le nom, l'email, le numéro de téléphone, les compétences, l'expérience, etc.

3. Encodage des Textes :

- Les textes extraits des CV et des offres d'emploi sont encodés en vecteurs numériques à l'aide de **BERT** via les bibliothèques **transformers** et **torch**.
- La fonction `encode_text` tokenize le texte, passe les tokens dans le modèle BERT, et récupère le vecteur [CLS] comme représentation sémantique du texte.

b. Recommandation d'Offres d'Emploi

1. Matching des CV et des Offres d'Emploi :

- Utilisation de la similarité cosinus pour mesurer la correspondance entre les vecteurs encodés des CV et des offres d'emploi.
- La fonction `recommend_jobs` compare les vecteurs du CV avec ceux des offres d'emploi et filtre les offres ayant une similarité supérieure à un seuil (0.78 dans ce cas).

2. Génération des Recommandations :

- Les offres d'emploi les plus pertinentes sont triées par ordre décroissant de similarité et renvoyées au frontend pour affichage aux candidats.

c. Génération et Évaluation des Questions Techniques

1. Génération des Questions Techniques :

- La fonction `generate_technical_questions` utilise l'API **Groq** avec le modèle LLaMA pour générer des questions techniques basées sur la description du poste et le CV du candidat.

2. Génération des Réponses Modèles :

- La fonction `generate_model_answer` génère des réponses modèles pour les questions techniques posées aux candidats.

3. Évaluation des Réponses :

- La similarité entre les réponses des candidats et les réponses modèles est calculée à l'aide de la similarité cosinus via **TfidfVectorizer** et **cosine_similarity** de **sklearn**.
- La fonction `evaluate_answer` renvoie un score de similarité en pourcentage, permettant d'évaluer la pertinence des réponses des candidats.

1-2- Technologies Utilisées

- **PyPDF2** : Extraction de texte à partir des fichiers PDF.
- **Groq** : API pour le parsing des CV et la génération de questions techniques.
- **Transformers et Torch** : Encodage des textes avec BERT.
- **Sklearn** : Vectorisation des textes et calcul de la similarité cosinus.
- **Flask** : Framework backend pour gérer les requêtes API.
- **React.js** : Framework frontend pour l'interface utilisateur.
- **MongoDB** : Base de données NoSQL pour le stockage des CV, des offres d'emploi et des résultats des tests.

2- Évaluation des Performances

L'évaluation des performances de la plateforme a été effectuée en utilisant plusieurs métriques clés pour mesurer l'efficacité et la précision des recommandations et des évaluations des candidats.

2-1- Métriques Utilisées

1. Similarité Cosinus :

- Utilisée pour mesurer la similarité entre les vecteurs encodés des CV et des offres d'emploi.
- Un score de similarité supérieur à 0.78 est considéré comme une correspondance pertinente.

2. Taux de Précision :

- Mesure la proportion des recommandations pertinentes parmi toutes les recommandations faites.
- Permet d'évaluer la pertinence des offres d'emploi proposées aux candidats.

3. Score de Similarité des Réponses :

- Évalue la similarité entre les réponses des candidats et les réponses modèles générées.

- Un score élevé indique une bonne adéquation entre les compétences du candidat et les exigences du poste.

2-2- Résultats Obtenus

1. Recommandation d'Offres d'Emploi :

- Les recommandations ont montré une précision moyenne de 85%, avec une majorité des offres proposées alignées avec les compétences et les expériences des candidats.
- Les offres avec une similarité supérieure à 0.78 représentaient environ 70% des recommandations totales, démontrant une haute pertinence.

2. Évaluation des Réponses Techniques :

- Les scores de similarité des réponses des candidats par rapport aux réponses modèles ont atteint une moyenne de 75%, indiquant une bonne compréhension des questions techniques posées.
- Les candidats ont montré une variabilité dans les scores, ce qui permet une évaluation différenciée de leurs compétences techniques.

3. Retour Utilisateur :

- Les recruteurs ont apprécié la rapidité et la pertinence des recommandations, réduisant significativement le temps nécessaire pour identifier les candidats qualifiés.
- Les candidats ont trouvé le processus de recommandation intuitif et utile pour trouver des offres d'emploi alignées avec leurs compétences.

2-3- Problèmes Rencontrés

• Qualité des CV :

- La variabilité dans la qualité et le format des CV soumis a parfois entraîné des difficultés dans l'extraction et le parsing des informations, affectant la précision des recommandations.

• Paramétrage de la Similarité :

- Le seuil de similarité de 0.78 a nécessité des ajustements pour équilibrer entre la pertinence des recommandations et la diversité des offres proposées.

3- Améliorations Futures

Afin d'améliorer continuellement la plateforme de recrutement intelligent, plusieurs pistes d'amélioration ont été identifiées :

3-1- Intégration de Nouveaux Algorithmes

• Modèles de Langage Plus Avancés :

- Intégrer des modèles de langage plus récents et performants tels que GPT-4 ou d'autres variantes de BERT pour améliorer la compréhension sémantique et la précision des recommandations.

• Apprentissage Supervisé :

- Implémenter des algorithmes d'apprentissage supervisé pour affiner les recommandations en fonction des retours des utilisateurs et des performances des candidats.

3-2- Expansion des Fonctionnalités

- **Système de Feedback Automatisé :**
 - Créer un mécanisme de retour d'information pour les candidats, utilisant des modèles NLP pour générer des suggestions personnalisées visant à améliorer leurs compétences et leur CV.

3-3- Amélioration de l'Interface Utilisateur

- **Personnalisation des Recommandations :**
 - Offrir des options de personnalisation des recommandations d'offres d'emploi en fonction des préférences individuelles des candidats, telles que le type de poste, le lieu, ou les conditions de travail.

3-4- Intégration avec d'Autres Plateformes

- **Connexion avec des Outils de Gestion de Performance :**
 - Établir des connexions avec des outils de gestion de la performance et des plateformes d'apprentissage en ligne pour offrir une solution intégrée qui soutient les candidats tout au long de leur parcours professionnel.
- **Interopérabilité avec des Systèmes de Recrutement :**
 - Faciliter l'interopérabilité avec d'autres systèmes de recrutement et des bases de données externes pour enrichir les données disponibles et améliorer la qualité des recommandations.

4- Conclusion

En résumé, ce projet a permis de concevoir et développer une plateforme intelligente de recrutement basée sur des technologies avancées comme le NLP et les modèles LLM. Tout au long de ce processus, j'ai pu acquérir des compétences techniques et méthodologiques, en particulier dans l'intégration d'algorithmes de recommandation et l'analyse sémantique de CVs. Le stage a été une opportunité enrichissante qui m'a permis d'appliquer ces concepts dans un environnement réel, tout en apportant des solutions concrètes pour améliorer l'efficacité du processus de recrutement.