Labellisation automatique des offres d'emploi JOCAS

Travail réalisé entre septembre et novembre 2024

Noé Amar

Contexte du projet

Base JOCAS : plus de 6 millions d'offres d'emploi collectées depuis 2016.

Problème actuel :

- La base est riche mais encore difficile à exploiter.
- Pas de structure ni de thèmes identifiés dans les offres.
- Impossible pour l'instant d'analyser les tendances du marché de l'emploi.

Besoin : *Un modèle automatique, rapide et robuste* capable de structurer cette base.

Objectif du modèle

Objectif à court terme : labelliser automatiquement chaque offre selon 6 thématiques RH :

- Diversité
- Rémunération / Avantages
- Opportunités professionnelles
- Culture / Valeurs d'entreprise
- Leadership / Management
- Équilibre vie pro – vie perso

Contraintes fortes :

- Très faible coût par offre
- Labelisation en quelques secondes
- Généralisable à toute la base (+6M offres)

Objectif final : analyser l'évolution de ces thèmes dans le temps, par secteur, région, période post-Covid, etc.

Mois 1 — Prise en main du problème

Travail réalisé :

- Exploration de la base JOCAS et compréhension des formats de données.
- Reprise / nettoyage des scripts développés l'année précédente.
- Analyse des premières structures textuelles (offres, profils, descriptions. . .).
- Étude de quelques milliers d'offres déjà labelisées par Eliot.

Objectif de cette phase :

- Obtenir une *vision claire de la donnée* (volume, structure, qualité).
- Construire un premier modèle supervisé pour voir si les labels existants peuvent être appris et généralisés.
- Tester l'idée d'une *distillation* : reproduire les labels d'Eliot grâce à un modèle ML.

→ *Résultat attendu : un premier modèle de base, servant de point de comparaison.*

Exploration de la base JOCAS

1. Répartition temporelle des offres (extrait)

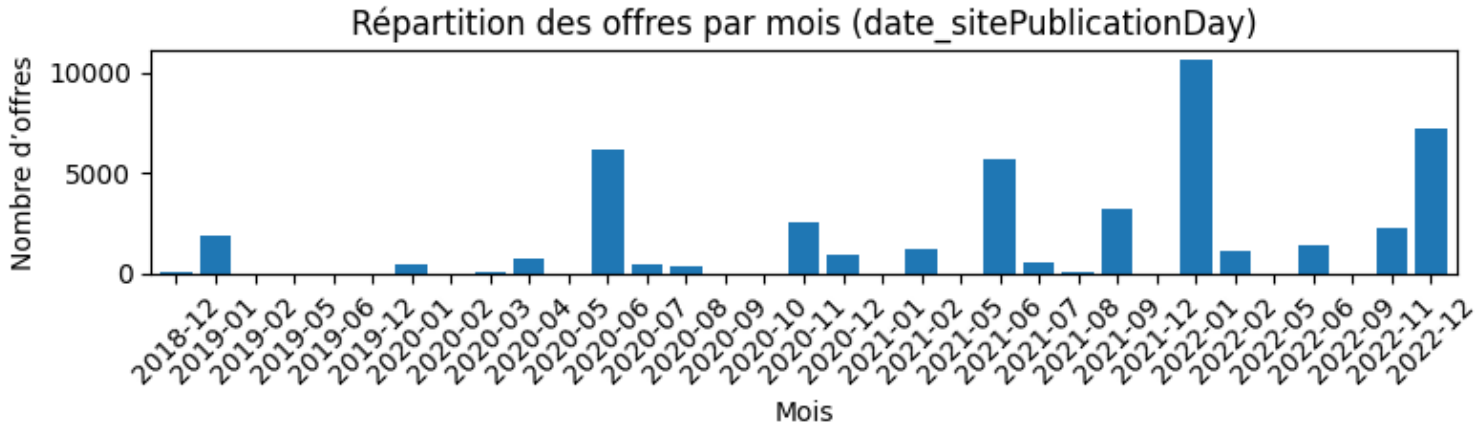

Figure: Offres par mois sur la plage d'étude (50 000 premieres offres) 2019-2022

→ Forte variabilité : certaines périodes d'activité intense

2. Qualité et complétude des données

- Certaines variables sont très complètes (titre, contrat, localisation. . .)
- Mais d'autres sont **quasi absentes** (salaire, entreprise SIREN, durée. . .)
- Les variables textuelles (description_) sont essentielles pour la labellisation.

Conclusion : le texte est la source d'information la plus fiable → approche NLP.

Analyse des données manquantes

Variables avec peu de données manquantes (fiables) :

- `job_title`, `contractType`, `location_departement`
- $\Rightarrow$ peuvent être utilisées comme features structurées.

Variables très incomplètes ($>$ 70% de valeurs nulles) :

- `salary_min`, `salary_max`, `education_level`, `SIREN...`
- $\Rightarrow$ inutilisables en apprentissage supervisé classique.

Conséquence méthodologique :

- Les **champs textuels** deviennent la seule source exploitable.
- Approche retenue : **modèles NLP**, embeddings, fine-tuning, distillation.

$\rightarrow$ *La qualité du texte justifie l'approche par labellisation automatique via LLM.*

Premiers essais de modélisation supervisée

Objectif : vérifier si les labels existants (donnés par GPT-4 l'an dernier) étaient suffisamment cohérents pour entraîner un modèle prédictif.

Modèle testé :

- **XGBoost** avec embeddings **TF-IDF** + quelques variables structurées
- *Entraînement multi-label, validation croisée*

Résultats (moyenne sur 5 folds) :

- F1-micro : **0.44**
- F1-macro : **0.48**

$\rightarrow$ *Le modèle n'arrive pas à généraliser : labels trop bruités / trop incohérents.*

Tentative avec un modèle plus puissant

Stratégie : tester un **transformer** fine-tuné (BERT / CamemBERT), puis lancer un **essai de RLHF** :

- Mise en place d'un dataset spécifique (format instruction $\rightarrow$ *réponse*)
- Préparation du training set façon *chat format*
- Entraînement via **GRPO / DPO** (Reward Modeling)

Résultats GRPO (après fine-tuning) :

- F1-micro : **0.62**
- F1-macro : **0.64**
- Mais **aucune amélioration sur certaines classes**

Problème clef :

- *Mise en place très lourde* (formatage données + pipeline RLHF)
- Gains limités $\rightarrow$ le bruit dans les labels restait trop important

Conclusion : le vrai problème n'était pas le modèle, mais la qualité des données.

Pourquoi les modèles ne convergeaient pas ?

Problème principal : la qualité des labels existants.

Constats :

- Labels générés automatiquement avec GPT-4 l'an dernier.
- Absence de cohérence dans le raisonnement utilisé.
- Deux offres similaires pouvaient recevoir des labels différents.
- $\rightarrow$ *Le modèle n'arrivait pas à apprendre de régularité.*

Performances en validation croisée :

- F1-micro : **0.45 – 0.50**
- F1-macro : **0.40 – 0.45**

Conclusion : *Le problème ne venait pas du modèle, mais des données. Il fallait* **relabelliser proprement** *pour pouvoir généraliser.*

Changement de stratégie — Mois 2

Nouvelle idée : reconstruire ENTIEREMENT la labellisation avec un LLM fiable.

- Utilisation de **GPT-5** (meilleures capacités de raisonnement).
- Labellisation complète avec détection d'incertitude.
- **Ajout d'un score de confiance** (0 - 100).
- Si incertitude $\Rightarrow$ stocké dans *to_review.jsonl* pour traitement manuel.

$\rightarrow$ *Objectif : ne garder que des labels "sûrs" pour entraîner un vrai modèle ML.*

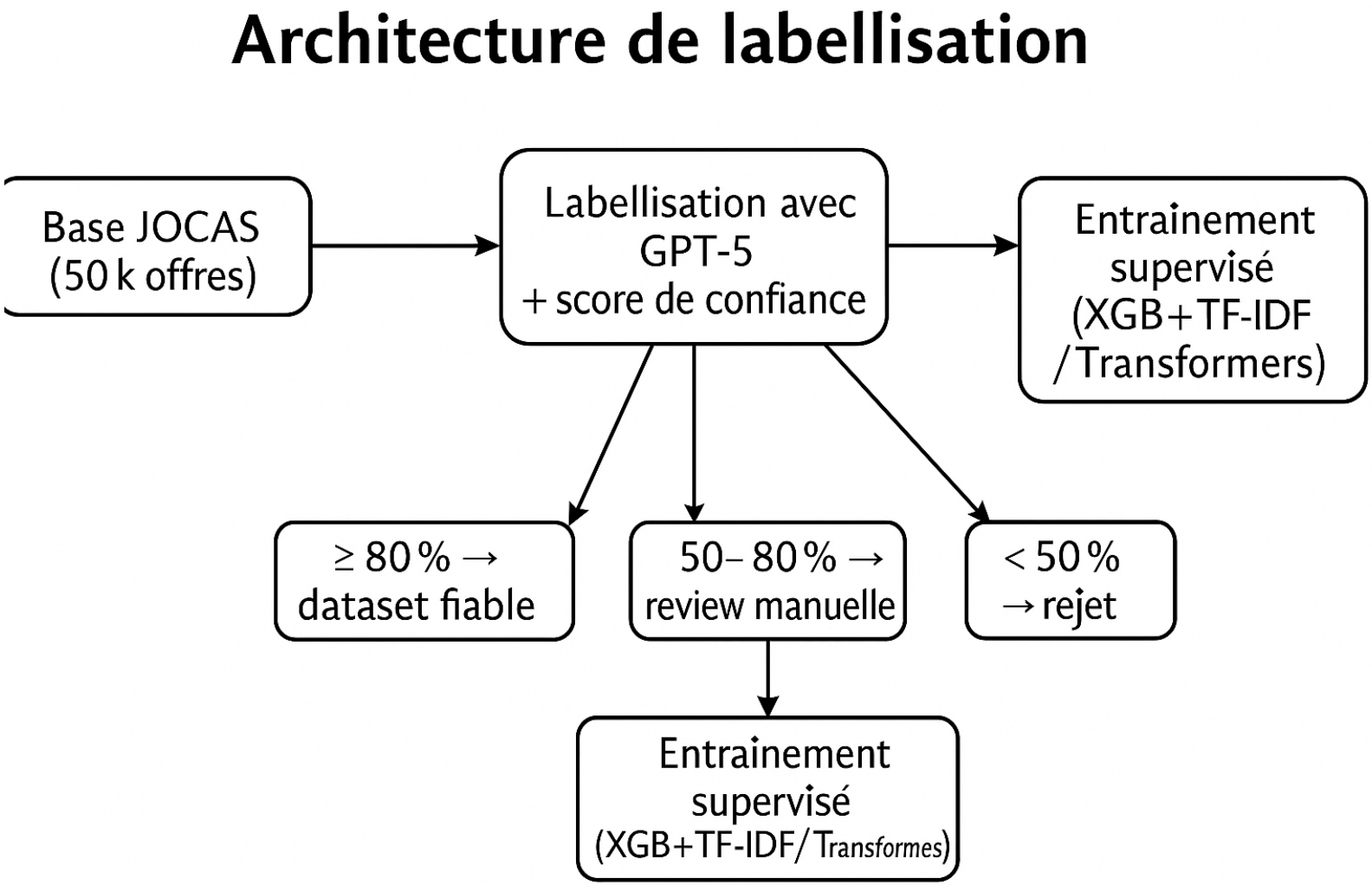

Figure: Pipeline de labelling

Premiers entraînements (XGB + TF-IDF)

TF-IDF = représentation vectorielle simple du texte basée sur la fréquence des mots.
Modèle utilisé : *XGBoost + MultiOutputClassifier*
Avec seuil de confiance $\geq 50\%$:

- F1-micro ≈ 0.67
- F1-macro ≈ 0.61

Mais avec seuil $\geq 80\%$ (env. 2000 labels fiables) :

- **F1-micro mean : 0.8830**
- **F1-macro mean : 0.8541**

Comment interpréter ces performances ?

Exemple : F1 = 0.88

$\Rightarrow$ si on classe **100 offres**, le modèle se trompe **environ 12 fois**.

- Très bon pour un modèle non transformer.
- Certaines classes sont naturellement difficiles (ex : Diversité / Work-Life Balance).
- Mais la confiance du LLM semble corrélée à la "clarté" sémantique du texte.

Conclusion : le score de confiance est un véritable *filtre qualitatif*.

Évolution temporelle : volume + labels

Contexte : les données ne sont pas continues (*absence d'offres certains mois*) $\rightarrow$ un lissage est nécessaire pour observer des tendances globales.

1 label analysé :

WORK_LIFE_BALANCE — tendance post-Covid.

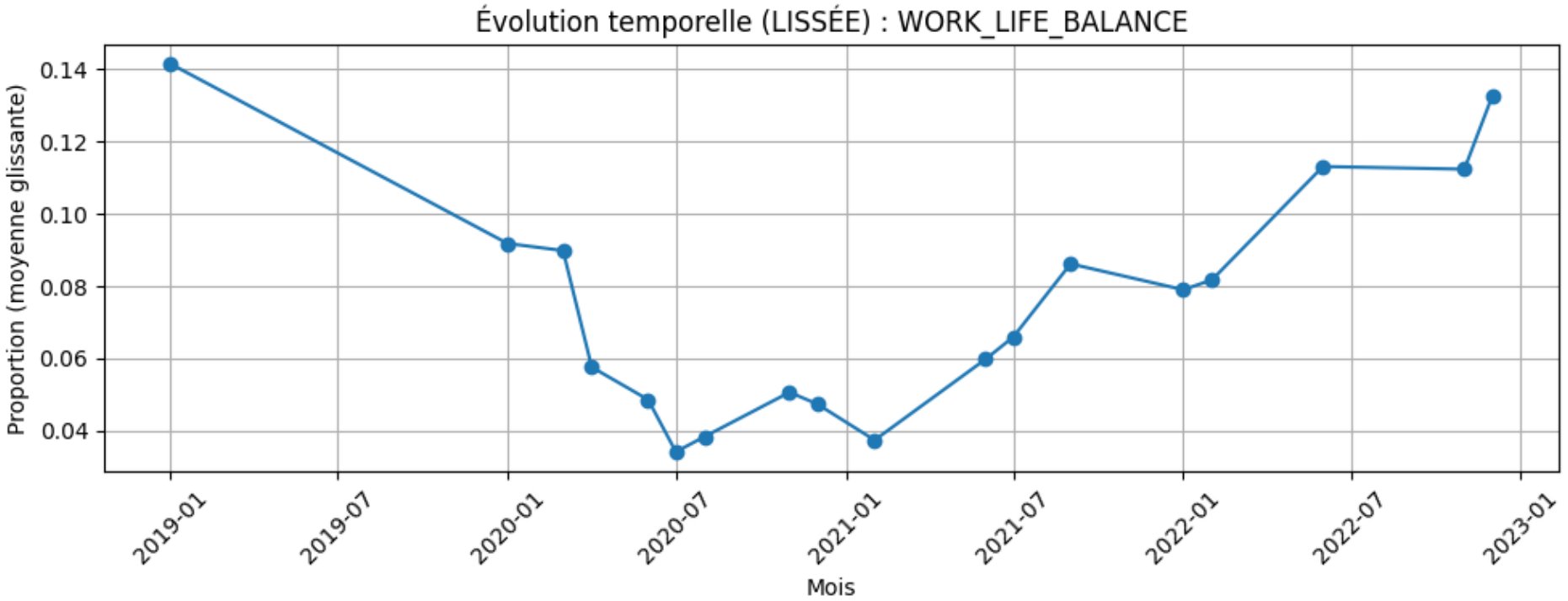

Figure: Work life balance label

Perspectives — Décembre

Objectif court terme :

- Atteindre F1 ≥ 0.90 en continuant à labelliser.
- Remplacer TF-IDF par un **BERT fine-tuné** (ou RoBERTa).
- Entraîner sur 10k labels fiables.
- Préparer pipeline pour **labelliser plus de 6M offres JOCAS**.
- Reprendre l'application web d'Eliot pour l'adapter

Conclusion

- La méthodologie “score de confiance” fonctionne.
- J’ai désormais un pipeline **rapide, scalable et fiable**.
- Le modèle actuel peut labelliser des milliers d’offres en quelques secondes.
- Base solide pour analyser les **dynamiques temporelles du marché du travail**.

→ **La base JOCAS peut enfin devenir exploitable à grande échelle.**

Questions / Discussion

Points possibles à discuter :

- Fiabilité des labels générés par le LLM
- Pertinence de la stratégie : *labeliser d'abord, analyser ensuite*
- Amélioration du modèle (BERT / embeddings denses)
- Scalabilité sur 6 millions d'offres
- Analyse longitudinale : évolution des thèmes RH dans le temps