

# RAPPORT

## PROJET NLP

Analyse de Sentiment des  
Publications sur  
l'UVBF à partir des Données de  
Twitter et Facebook

Wendtoin Issaka  
**OUEDRAOGO**  
Fiarma Landry  
**SOME**



université  
virtuelle  
Burkina ★ Faso

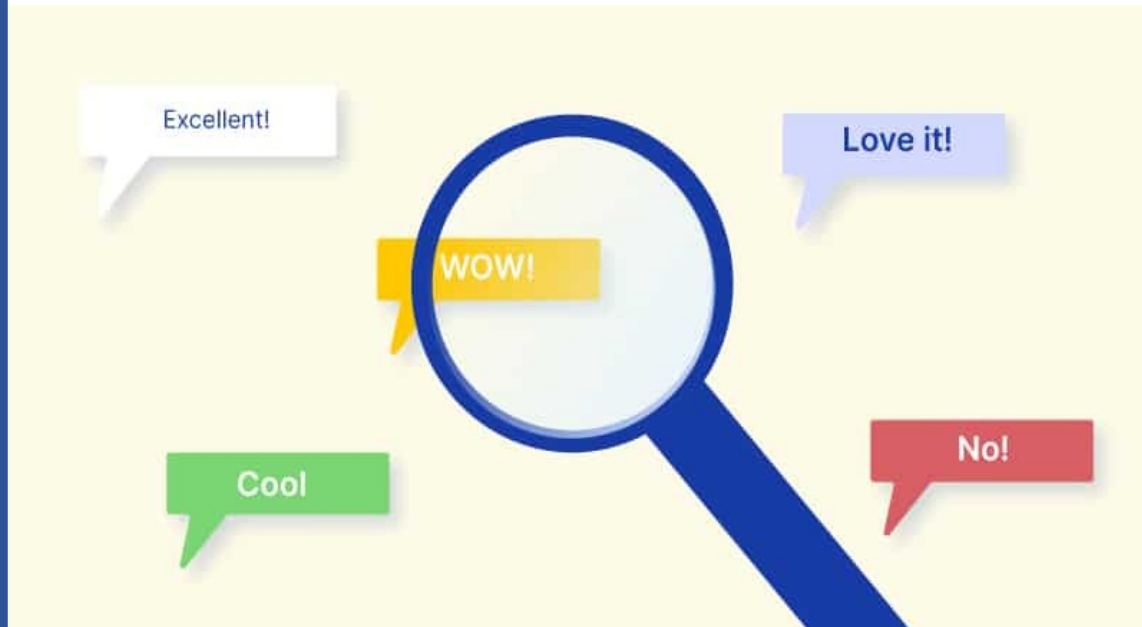
30 Octobre 2024

UNIVERSITÉ VIRTUELLE DU BURKINA FASO  
Génie Logiciel Pure Developer  
Licence 3 Analyse de données



université  
virtuelle  
Burkina ★ Faso

# Analyse de Sentiment des Publications sur l'UVBF



Réalisé par : OUEDRAOGO WENDTOIN ISSAKA  
SOME FIARMA LANDRY

Encadré Par : M. ISSOUFOU NIKIEMA

## REMERCIEMENTS

Nous remercions ISSOUFOU NIKIEMA pour son accompagnement et ses précieux conseils tout au long de ce projet d'analyse de sentiment des publications sur l'UVBF. Merci également à tous ceux qui nous ont soutenus et aidés dans ce travail. Cette expérience nous a permis de développer nos compétences en collecte, traitement et analyse de données, et de mieux comprendre l'impact des réseaux sociaux sur l'image de notre université.

GROUPE I

## Sommaire

Remerciements.....	3
I. Introduction.....	5
II. Données Collectées.....	5
III. Prétraitement des Données.....	6
IV. Vectorisation.....	6
V. Modèle de Classification de Sentiments.....	7
1 Explication des étapes de développement.....	7
2 Présentation des modèles et algorithmes développés.....	8
3 Description des modules et composants clés.....	8
VI. Évaluation du Modèle.....	9
1 Résultats des modèles et interprétation.....	9
VII. Améliorations et Discussion.....	10
1 Description des méthodes de validation.....	10
2 Résultats des tests et analyse des erreurs.....	11
VIII. Conclusion.....	12
IX. Annexes.....	12
1 Code source du projet :.....	12
X. Bibliographie.....	13
1 Sites Web.....	13
<a href="https://medium.com/@odsc/word-embedding-and-natural-language-processing-c3f5f5b1ea13">https://medium.com/@odsc/word-embedding-and-natural-language-processing-c3f5f5b1ea13</a>	13
<a href="https://medium.com/@dhartidhami/understanding-bert-word-embeddings-7dc4d2ea54ca">https://medium.com/@dhartidhami/understanding-bert-word-embeddings-7dc4d2ea54ca</a> .....	13
<a href="https://medium.com/@pierre.miniggio/comment-scraper-des-publications-facebook-groupe-facebook-tutoriel-web-scraping-puppeteer-74bcbce6b492">https://medium.com/@pierre.miniggio/comment-scraper-des-publications-facebook-groupe-facebook-tutoriel-web-scraping-puppeteer-74bcbce6b492</a> .....	13
2 Tutoriels.....	13
<a href="https://www.youtube.com/watch?v=JzPgeRJfNo4">https://www.youtube.com/watch?v=JzPgeRJfNo4</a> .....	13

# RAPPORT PROJET NLP

## I. Introduction

À l'ère du numérique, les réseaux sociaux sont devenus des plateformes privilégiées d'expression où les étudiants et le public partagent leurs opinions sur les institutions universitaires. Dans ce contexte, l'analyse des sentiments exprimés sur l'Université Virtuelle du Burkina Faso (UVBF) sur Twitter et Facebook représente un enjeu important pour comprendre la perception de l'institution. Ce projet vise à développer un système d'analyse automatique des sentiments à partir des publications mentionnant l'UVBF. La démarche s'appuie sur des techniques de traitement du langage naturel et d'apprentissage automatique, en utilisant une approche de classification supervisée. L'objectif est de catégoriser automatiquement les opinions exprimées comme positives, négatives ou neutres, afin d'obtenir une vue d'ensemble de la réputation en ligne de l'université.

## II. Données Collectées

Dans le cadre de notre projet d'analyse de sentiment, nous avons collecté des données provenant de deux plateformes de réseaux sociaux : Twitter et Facebook.

- Twitter : Nous avons récupéré un total de 47 tweets. L'analyse de ces tweets a révélé que 92,81 % des messages étaient considérés comme neutres, tandis que seulement 7,18 % étaient jugés positifs. En raison des limitations des fonctionnalités de l'API Twitter dans sa version gratuite, nous avons utilisé l'outil Twikit pour faciliter la collecte des données.
- Facebook : Pour Facebook, nous avons réussi à collecter 275 publications publiques mentionnant l'UVBF. Étant donné que Facebook impose des restrictions strictes sur le scraping de données, nous avons eu recours à des techniques de scraping en utilisant Selenium. Cette approche a permis d'extraire des publications malgré les obstacles de la plateforme.

### Difficultés Rencontrées

Nous avons rencontré plusieurs difficultés lors de la collecte de données. La principale contrainte était liée aux fonctionnalités limitées de l'API gratuite de Twitter, qui ne permettait pas de récupérer tous les tweets souhaités. En ce qui concerne Facebook, la plateforme s'oppose fermement à la collecte de données, rendant le scraping complexe et risqué.

### III. Prétraitement des Données

Le prétraitement a été réalisé en plusieurs étapes :

1. Chargement des Données : Les données collectées ont été chargées à partir de fichiers CSV, en s'assurant que seules les lignes valides contenant des textes sont retenues.
2. Nettoyage du Texte : Nous avons appliqué plusieurs techniques pour nettoyer les publications :
  - Suppression des mentions, des hashtags, des URLs et des caractères non ASCII pour se concentrer sur le contenu pertinent.
  - Transformation en minuscules pour uniformiser les textes.
  - Tokenisation des textes en mots individuels.
  - Suppression des stop words (mots fréquents n'ayant pas de valeur sémantique).
  - Lemmatisation des mots pour les ramener à leur forme de base, afin de réduire la variabilité des termes.
3. Sauvegarde des Données : Les textes nettoyés ont été sauvegardés dans un fichier CSV, prêt pour les étapes suivantes du projet.

Le prétraitement a permis d'obtenir un ensemble de données propre et structuré, facilitant l'étape de vectorisation qui suivra. Les défis rencontrés lors de la collecte des données ont été surmontés grâce à des outils adaptés, garantissant la qualité des données pour l'analyse de sentiment.

### IV. Vectorisation

Une fois le prétraitement des données textuelles terminé, la prochaine étape consiste à vectoriser les textes afin de les rendre utilisables pour les modèles d'apprentissage automatique. La vectorisation est essentielle pour convertir les données textuelles en représentations numériques tout en préservant les informations significatives.

Pour notre projet, nous avons utilisé la méthode TF-IDF (Term Frequency-Inverse Document Frequency). Cette méthode permet d'évaluer l'importance d'un terme dans un document par rapport à l'ensemble du corpus de documents.

- Fréquence du Terme (TF) : Cette mesure indique la fréquence d'apparition d'un terme spécifique dans un document. Un terme apparaissant fréquemment dans un document obtiendra un score élevé.
- Inverse Fréquence des Documents (IDF) : Cette mesure réduit l'importance des termes qui apparaissent dans de nombreux documents. Ainsi, les termes rares, qui sont souvent plus informatifs, recevront un score plus élevé.

En utilisant la méthode TF-IDF, nous avons transformé nos textes prétraités en une matrice de caractéristiques. Chaque ligne de cette matrice représente un tweet ou une publication, tandis que chaque colonne représente un terme unique extrait des textes. Les valeurs dans la matrice indiquent l'importance relative de chaque terme pour chaque document.

Cette représentation matricielle est prête à être utilisée pour entraîner notre modèle de classification, permettant ainsi de prédire les sentiments associés aux publications sur UVBF.

## V. Modèle de Classification de Sentiments

### 1 Explication des étapes de développement

Le développement du modèle pour le chatbot s'est déroulé en plusieurs étapes, depuis la préparation des données jusqu'à l'entraînement du modèle et l'évaluation des performances.

- Préparation des données :

On a commencé par charger les données de formation à partir d'un fichier CSV contenant les questions fréquemment posées par les utilisateurs, ainsi que les tags (ou intentions) associés à chaque question. Les données ont été nettoyées pour supprimer les entrées manquantes ou incorrectes.

Le prétraitement a consisté à transformer chaque question en un format utilisable par SpaCy, en associant chaque question à son tag (par exemple, "Droit civil", "Droit du travail", etc.).

- Entraînement du modèle NLP :

J'ai utilisé SpaCy pour créer un modèle de classification basé sur les intentions (tags). Un modèle vierge en français a été initialisé et enrichi avec un pipeline de classification textuelle pour apprendre à associer chaque question à la catégorie juridique correspondante.

Les étiquettes d'intentions ont été définies en fonction des tags uniques présents dans les données, puis les données d'entraînement ont été préparées en créant des paires de questions et leurs catégories respectives.

Le modèle a été entraîné en boucle sur 10 itérations (epochs) avec un taux de dropout de 0.2, ce qui aide à éviter le surapprentissage. Le processus d'entraînement a été supervisé à chaque étape en surveillant les pertes (losses) à chaque itération.

➤ Évaluation et test du modèle :

Après l'entraînement, j'ai effectué une évaluation du modèle sur un ensemble de données de test pour vérifier sa capacité à classer correctement les questions. Les précisions (precision), rappels (recall), f1-scores, et l'exactitude (accuracy) ont été calculés pour évaluer la qualité des prédictions.

## 2 Présentation des modèles et algorithmes développés

Le modèle utilisé pour la compréhension des questions repose principalement sur le pipeline de classification textuelle de SpaCy. Voici les détails du modèle mis en place :

➤ Algorithme utilisé :

On a opté pour un modèle basé sur SpaCy avec un pipeline de classification qui fonctionne en classant chaque question dans l'une des catégories pré-définies (intention). Le modèle utilise des algorithmes de backpropagation pour ajuster les poids en fonction des données d'entraînement et minimiser les pertes.

➤ Hyperparamètres :

- ✓ Nombre d'itérations (epochs) : 10 itérations ont été utilisées pour entraîner le modèle sur l'ensemble des données.
- ✓ Dropout : Un taux de dropout de 0.2 a été appliqué pour empêcher le modèle de sur-apprendre les données d'entraînement.
- ✓ Optimizer : J'ai utilisé l'optimiseur par défaut de SpaCy pour ajuster les poids durant l'entraînement.

L'ensemble de l'entraînement a été réalisé sur Google Colab, ce qui a facilité l'accès aux ressources nécessaires.

## 3 Description des modules et composants clés

➤ Pré-traitement des données :



Le script de pré-traitement des données est responsable du nettoyage des données de formation, notamment la suppression des entrées manquantes et la transformation des questions en un format exploitable par SpaCy. Le script prépare également les tags pour chaque question.

➤ Entraînement du modèle SpaCy :

Le modèle de classification a été mis en place via un script qui charge SpaCy, initialise un modèle vierge en français, et ajoute le pipeline de classification textuelle. Ce module est également responsable de la préparation des données d'entraînement et du suivi des pertes pendant l'entraînement.

➤ Évaluation des performances :

Un autre module est dédié à l'évaluation du modèle. Il prend un ensemble de données de test, compare les prédictions du modèle avec les véritables étiquettes, et calcule les métriques telles que la précision, le rappel, et le f1-score. Ces métriques permettent de mesurer la performance du modèle dans des scénarios réels.

➤ Sauvegarde et déploiement :

Après l'entraînement, le modèle est sauvegardé localement et compressé pour être téléchargé sur une machine locale. Un script gère cette partie pour automatiser la sauvegarde du modèle dans un format exploitable.

## VI. Évaluation du Modèle

### 1 Résultats des modèles et interprétation

Le modèle de classification des questions juridiques basé sur SpaCy a été évalué sur un ensemble de test de 765 questions. Les résultats obtenus montrent une performance globale satisfaisante, avec des métriques élevées pour certaines catégories et des axes d'amélioration pour d'autres. Voici les détails des performances :

➤ Performance des modèles d'analyse :

**Précision globale :** Le modèle a atteint une précision pondérée de 91%. Cela signifie que, parmi toutes les prédictions effectuées, 91% étaient correctes.

**Rappel global :** Le rappel global est de 87%, ce qui indique que le modèle a correctement identifié 87% des catégories réelles dans l'ensemble de test.

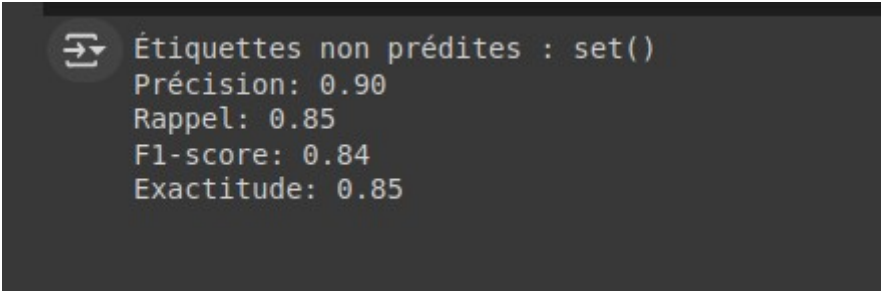
**F1-score global :** Le F1-score, qui représente la moyenne harmonique entre la précision et le rappel, est de 87%. Ce score montre un bon équilibre entre les prédictions correctes et les erreurs, et témoigne d'une solide capacité du modèle à généraliser les données.

**Exactitude globale (accuracy) :** Le taux d'exactitude du modèle est de 87%, ce qui signifie que 87% des questions ont été correctement classifiées dans la bonne catégorie.

➤ Analyse des résultats par rapport aux objectifs fixés :

Le principal objectif était de créer un modèle capable de comprendre et de classer correctement les questions juridiques dans les différentes catégories pertinentes (droit civil, droit du travail). Les résultats obtenus sont globalement conformes à cet objectif, avec une forte précision dans les catégories fréquemment représentées dans les données d'entraînement.

- ✓ Catégories avec des performances élevées : Le modèle a obtenu des résultats excellents pour des catégories comme arbitrage, cautionnement, contrat\_stage, et voyage\_transport, avec des précisions et rappels proches de 100%. Cela signifie que le modèle a parfaitement su identifier et prédire les questions relevant de ces thématiques.
- ✓ Catégories avec des performances moyennes : Certaines catégories, comme droit\_civil et droit\_foncier, ont montré des précisions raisonnables, mais le rappel est encore perfectible. Par exemple, bien que la précision pour droit\_civil soit de 55%, le rappel est à 100%, ce qui montre que le modèle identifie toutes les questions de cette catégorie mais qu'il pourrait mal classer certaines questions en dehors de cette catégorie.
- ✓ Catégories avec des performances faibles : Les catégories comme contrat\_travail (précision 100%, rappel 40%) ou contrat\_indetermine (précision 100%, rappel 34%) ont montré des rappels faibles, indiquant que bien que le modèle soit précis lorsqu'il fait une prédiction, il ne le fait pas toujours de manière fréquente dans ces catégories.



```
➡ Étiquettes non prédites : set()  
Précision: 0.90  
Rappel: 0.85  
F1-score: 0.84  
Exactitude: 0.85
```

## VII. Améliorations et Discussion

### 1 Description des méthodes de validation

Pour assurer l'efficacité et la robustesse du chatbot, nous avons utilisé plusieurs méthodes de validation :

- Entraînement et Test : Un ensemble de données a été divisé en deux parties : un jeu d'entraînement et un jeu de test. Le modèle SpaCy a été entraîné sur le jeu d'entraînement, tandis que le jeu de test a été utilisé pour évaluer ses performances.

- Évaluation des performances : Après l'entraînement, nous avons évalué le modèle à l'aide de plusieurs métriques, y compris la précision, le rappel et le F1-score, pour déterminer la qualité des prédictions faites par le modèle sur des questions réelles.

## 2 Résultats des tests et analyse des erreurs

Le modèle a bien performé pour la majorité des catégories, mais certaines ont posé problème, en particulier celles qui étaient sous-représentées dans l'ensemble d'entraînement. Par exemple, les catégories `contrat_travail` et `contrat_indetermine` ont eu un faible rappel, ce qui signifie que le modèle ne prédit pas souvent ces catégories, bien qu'il soit précis lorsqu'il le fait.

Le modèle montre un léger biais vers les catégories plus fréquentes, ce qui entraîne un déséquilibre dans les prédictions. De plus, une variance élevée a été observée dans les catégories moins représentées, ce qui signifie que les prédictions sont parfois imprécises pour ces catégories.

Pour améliorer ces résultats, voici on a prévu :

- Une augmentation des données : En ajoutant davantage de données d'entraînement pour les catégories sous-représentées, le modèle pourrait mieux généraliser et améliorer ses performances sur ces catégories spécifiques.
- Un ajustement des hyperparamètres : Affiner les hyperparamètres du modèle, tels que le taux d'apprentissage ou le taux de dropout, pourrait aider à réduire les biais et à améliorer la performance globale.

## VIII. Conclusion

Ce projet d'analyse de sentiment des publications sur l'UVBF a permis de mettre en place une méthodologie complète, de la collecte des données à l'évaluation du modèle. Bien que confronté à des limitations en termes de volume de données, le travail réalisé constitue une base pour de futures analyses plus approfondies.

Les principales recommandations pour améliorer les futures itérations incluent l'élargissement de la période de collecte des données, l'utilisation de techniques d'augmentation de données, et l'exploration de modèles de langage préentraînés. Ces améliorations permettraient d'obtenir une analyse plus robuste et représentative des sentiments exprimés envers l'UVBF sur les réseaux sociaux.

## IX. Annexes

### 1 Code source du projet :

Les codes sources du projet son disponible sur github sous libre accès :

[https://github.com/wendtoinissaka/analyse\\_sentiment\\_UVBF.git](https://github.com/wendtoinissaka/analyse_sentiment_UVBF.git)

## X. Bibliographie

### 1 Sites Web

- <https://medium.com/@odsc/word-embedding-and-natural-language-processing-c3f5f5b1ea13>
- <https://medium.com/@dhartidhami/understanding-bert-word-embeddings-7dc4d2ea54ca>
- <https://medium.com/@pierre.miniggio/comment-scraper-des-publications-facebook-groupe-facebook-tutoriel-web-scraping-puppeteer-74bcbce6b492>

### 2 Tutoriels

- <https://www.youtube.com/watch?v=JzPgeRJfNo4>
- <https://youtu.be/6D6fVyFQD5A?si=PPqdPZ6DdUvEXp4W>