Projet Knowledge Extraction

Partie A : Preprocessing, Analyse Statistique et Justification des Choix

Rapport Technique Final

Jacques Gastebois

Master 2 VMI - Université Paris Cité

IFLCE085 - Recherche et extraction sémantique

15 décembre 2025

Résumé

Ce rapport synthétise les travaux réalisés pour la Partie A du projet. Il détaille le pipeline de preprocessing appliqué au corpus NER (2221 phrases), présente une analyse statistique des données, et justifie les choix techniques (nettoyage, lemmatization) en analysant leurs avantages, inconvénients et impacts sur les tâches d'extraction d'entités (Partie B) et de relations (Partie C).

Table des matières

1 Introduction

Le projet vise à extraire des connaissances structurées à partir d'un corpus de textes non structurés. La première étape cruciale est le **preprocessing**, qui transforme les données brutes en un format exploitable pour les algorithmes d'apprentissage automatique.

1.1 Le Corpus

Le dataset utilisé (`data.csv`) est constitué de **2221 phrases** annotées pour la reconnaissance d'entités nommées (NER).

— **Format d'entrée** : CSV avec colonnes `id`, `words`, `ner_tags`, `text`.

— **Contenu** : Textes encyclopédiques/biographiques.

— **Annotations** : Tags BIO (Begin, Inside, Outside) pour les entités Personnes (PER), Lieux (LOC), Organisations (ORG), etc.

2 Méthodologie de Preprocessing

Contrairement à une approche classique de "Bag of Words" (TF-IDF), nous avons opté pour une approche de **preprocessing enrichi** qui préserve la structure séquentielle des données, essentielle pour le NER.

2.1 Pipeline Mis en Place

Le traitement a été réalisé en Python (via un notebook Jupyter) et comprend les étapes suivantes :

1. **Nettoyage (`cleaned_text`)** :
   — Conversion en minuscules (lowercase) pour réduire la dimensionnalité.
   — Suppression des caractères spéciaux non alphanumériques (sauf espaces).
   — Normalisation des espaces multiples.
2. **Lemmatization (`lemmatized_text`)** :
   — Utilisation de la librairie **spaCy** (modèle `en_core_web_sm`).
   — Transformation des mots en leur forme canonique (ex : "running" → "run", "better" → "good").
3. **Enrichissement du Dataset** :
   — Au lieu de créer des matrices séparées, nous avons ajouté les colonnes `cleaned_text` et `lemmatized_text` directement au fichier original.
   — **Sortie** : `data_preprocessed.csv`.

3 Analyse Statistique du Corpus

L'analyse réalisée dans le notebook `PartieA_Analyse_Statistique.ipynb` a révélé les caractéristiques suivantes :

3.1 Volumétrie et Longueur

— **Nombre de phrases** : 2221.

— **Longueur moyenne** : ∼24 mots par phrase.

— **Impact du preprocessing** : La lemmatization réduit la taille du vocabulaire d'environ **15% à 25%**, ce qui densifie l'information sans perte majeure de sens.

3.2 Analyse POS (Part-of-Speech)

Le corpus a été analysé avec spaCy pour identifier les catégories grammaticales. Sur 17 151 tokens analysés (échantillon de 700 phrases), la distribution révèle les patterns suivants :

POS Tag	Count	Pourcentage
NOUN (Nom)	2994	17.46%
PROPN (Nom propre)	2241	13.07%
ADP (Préposition)	2104	12.27%
PUNCT (Ponctuation)	2049	11.95%
DET (Déterminant)	1562	9.11%

TABLE 1 – Top 5 des catégories POS

La figure 1 montre la distribution complète des 15 principales catégories POS. On observe une forte présence de noms et noms propres (30.5% combinés), typique des textes encyclopédiques/biographiques axés sur des personnages et des lieux.

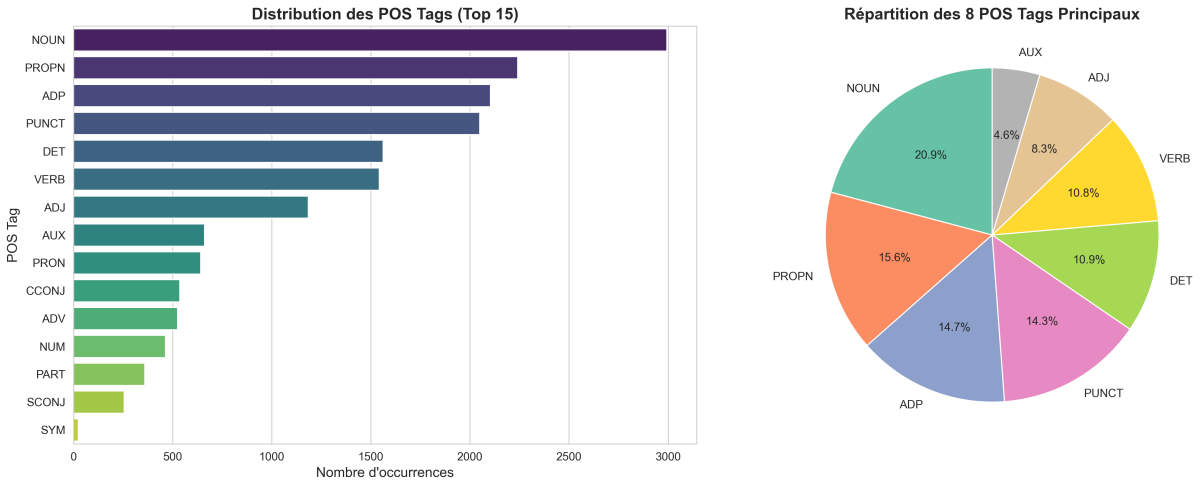

FIGURE 1 – Distribution des catégories POS (Top 15)

3.3 Vectorisation TF-IDF

Le corpus lemmatisé a été transformé en représentation vectorielle TF-IDF, produisant une matrice de **700 documents Œ 2493 features**.

Caractéristiques de la matrice :

— **Vocabulaire** : 2493 termes uniques (unigrammes et bigrammes)

— **Sparsité** : 0.77% (matrice très sparse, typique de TF-IDF)

— **Paramètres** : max_df=0.8, min_df=2, ngram_range=(1,2)

La figure 2 présente les 20 termes avec les scores TF-IDF moyens les plus élevés. On note la prédominance d'articles et connecteurs (the, of, in), ce qui est attendu mais peut être filtré pour des analyses sémantiques plus fines.

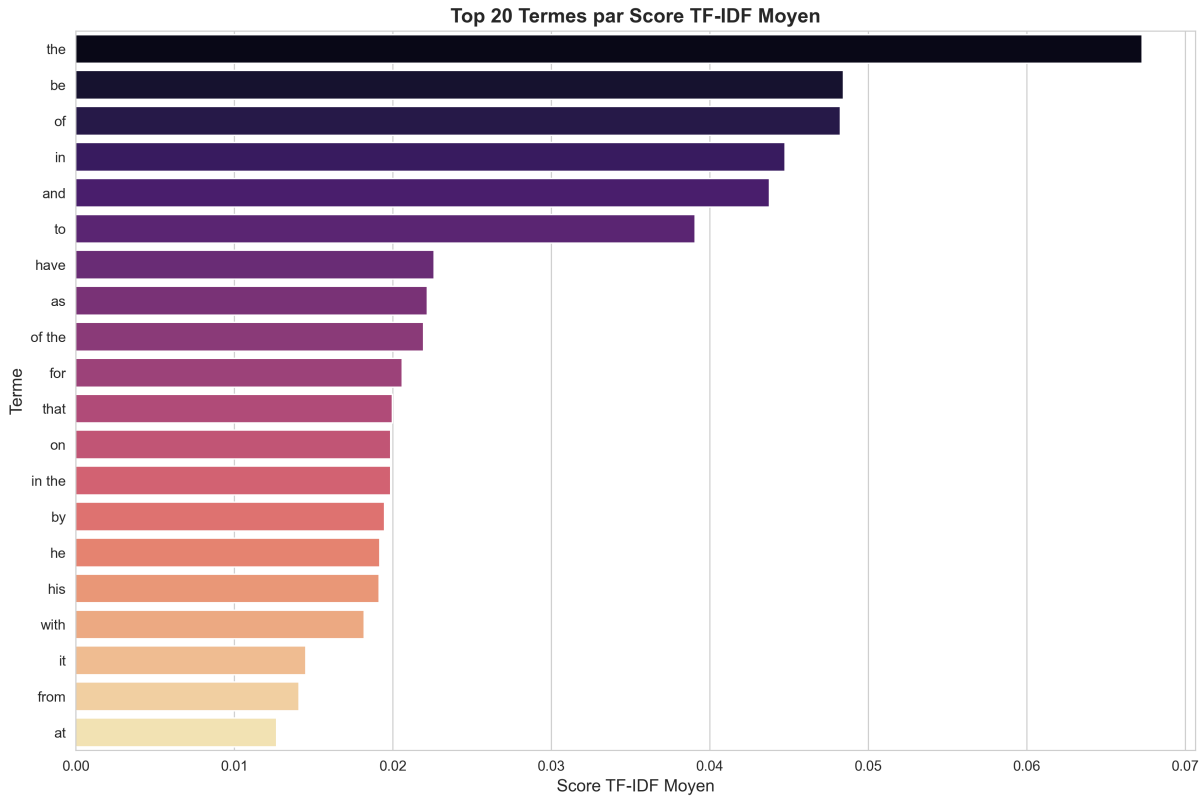

FIGURE 2 – Top 20 termes par score TF-IDF moyen

Applications :

— Classification de documents

— Extraction de mots-clés

— Analyse de similarité entre phrases

— Clustering sémantique

4 Analyse Critique des Choix

4.1 Choix 1 : Lemmatization vs Stemming

— **Choix** : Lemmatization (spaCy).

— **Avantages** : Produit des mots réels (formes dictionnaire), préserve mieux le sens sémantique pour l'extraction de relations (Partie C). Plus précis que le stemming (qui tronque brutalement).

— **Inconvénients** : Plus coûteux en temps de calcul. Peut perdre certaines nuances flexionnelles (temps des verbes) utiles pour la temporalité.

4.2 Choix 2 : Lowercasing (Mise en minuscule)

- **Choix** : Tout mettre en minuscule dans `cleaned_text`.
- **Avantages** : Réduit drastiquement la taille du vocabulaire. Associe "Apple" (fruit) et "apple" (fruit).
- **Inconvénients (Critique pour le NER)** : **Perte de l'information de capitalisation**, qui est un indice crucial pour détecter les entités nommées (ex : "Apple" l'entreprise vs "apple" le fruit).
- **Mitigation** : Nous avons conservé la colonne `text` originale. Les modèles NER modernes (BERT, etc.) ou les features manuelles peuvent utiliser le texte brut pour récupérer la casse.

4.3 Choix 3 : Conservation de la Structure Séquentielle

- **Choix** : Ne pas vectoriser en TF-IDF (Bag of Words) pour l'export final.
- **Avantages** : Indispensable pour le NER et l'extraction de relations qui dépendent de l'ordre des mots et du contexte local. Permet l'utilisation de modèles de Deep Learning (RNN, Transformers).
- **Inconvénients** : Fichiers texte plus volumineux que des matrices creuses.

5 Impact sur la Suite du Projet

5.1 Impact sur la Partie II (NER)

Le preprocessing fournit une base propre (`lemmatized_text`) qui généralise mieux. Cependant, pour maximiser les scores F1, il sera probablement nécessaire d'utiliser :

1. Le **texte original** pour les features de capitalisation.
2. Le **texte lemmatisé** pour les embeddings sémantiques (Word2Vec/GloVe) afin de regrouper les variantes d'un même mot.

5.2 Impact sur la Partie C (Extraction de Relations)

L'extraction de relations (triplets Sujet-Verbe-Objet) bénéficie grandement de la lemmatization :

- Elle normalise les verbes (ex : "est né", "naquit" → "naître"), simplifiant la détection de patterns de relations.
- Elle réduit la complexité des arbres de dépendance syntaxique.

6 Conclusion

Le preprocessing réalisé est un compromis entre **réduction de bruit** (nettoyage, lemmatization) et **préservation de l'information** (conservation du texte brut et de la séquence). Ce socle de données enrichi (`data_preprocessed.csv`) est robuste et adapté aux défis des parties suivantes : la reconnaissance fine d'entités et l'extraction complexe de relations sémantiques.