Projet de Réduction de dimension

Master 2 Data science : Santé, assurance, finance

2024 - 2025

AH-MOUCK Laetitia

Analyse de sentiments via mélange d'experts et représentation des critiques de films de IMDb

---

Table des matières

1 Introduction

L'analyse de sentiments est un domaine clé du NLP, appliqué à diverses tâches telles que la détection des émotions dans les commentaires clients, les opinions sur les réseaux sociaux et l'évaluation des produits. Ce projet se concentre sur la classification des critiques de films issues de IMDb, en identifiant si une critique est positive ou négative.

Nous allons utiliser un modèle Mixture of Experts (MoE), une approche qui divise le problème en plusieurs sous-modèles spécialisés, combinés via un mécanisme d'agrégation. Cette approche permet d'améliorer la précision en attribuant différentes parties des données à des experts adaptés. Ce modèle sera comparé à d'autres modèles plus classiques comme le MLP, le CNN ou encore la régression logistique.

2 Prétraitement des données

L'ensemble de données IMDb contient 50 000 critiques de films, étiquetées comme positives ou négatives. Nous sélectionnons un sous-échantillon de 2 000 critiques (1 000 positives, 1 000 négatives) pour des raisons d'efficacité.

2.1 Preprocessing des données

Les données ont été traitées de la façon suivante :

— Suppression des balises HTML et des caractères spéciaux.
— Conversion en minuscules pour uniformiser le texte.
— Suppression des mots vides (stopwords) pour éliminer les mots non significatifs.
— Lemmatisation pour réduire les mots à leur racine.
— Tokenisation pour séparer chaque mot.

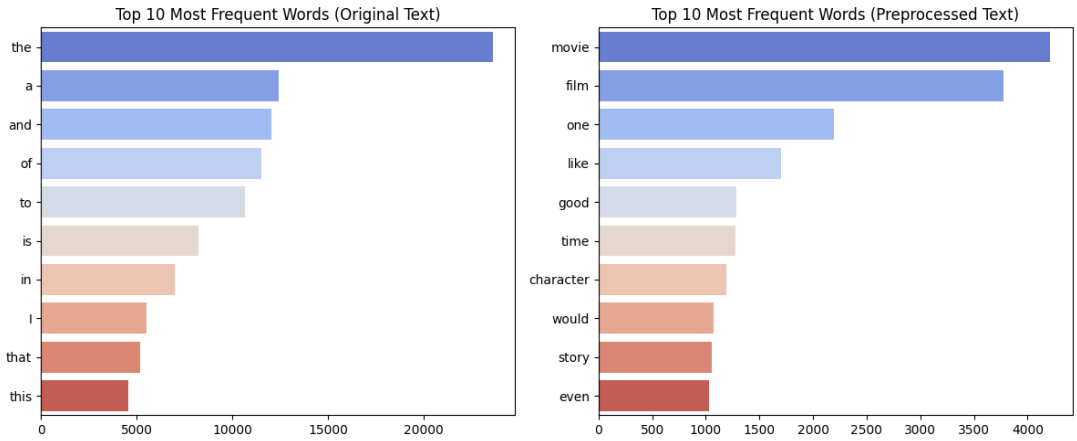

FIGURE 1 – Top 10 des mots les plus fréquents avant et après traitement des données

D'après les graphes, le traitement des données a bien fonctionné. Avant traitement, la plupart des mots les plus utilisés étaient des mots non significatifs. Après traitement, les mots les plus utilisés sont plus en accord avec la thématique des critiques de films.

Voici un exemple de critique avant traitement :
I am completely shocked that this show had been cancelled.Ity only lasted one year.I just recently started watching it and I love it.Its a show that could of gone as far as Friends went.It had the humour and was

extremely enjoyable.

It is about 2 brothers and 2 sisters living under one roof without their parents.Kurt(Joey Lawrence) plays the part of the oldest sibling and takes on the more fatherly role.

This should of lasted much more than year as it was fantastic. Amazing show with all the best actors.

10/10

Après traitement, nous avons :
completely shocked show cancelled ity lasted one year recently started watching love show could gone far friend went humour extremely enjoyable brother sister living one roof without parent kurt joey lawrence play part oldest sibling take fatherly role lasted much year fantastic amazing show best actor

3 Extraction de caractéristiques

Pour extraire les caractéristiques de nos données, le texte brut est converti en représentation numérique via deux méthodes : TF-IDF et le Word Embedding avec Word2Vec.

Le modèle `TF-IDF` est une méthode de pondération utilisée pour évaluer l'importance d'un mot dans un document par rapport à un corpus. Cette approche permet de transformer un texte en vecteurs numériques, où les mots fréquents dans un document mais peu fréquents dans l'ensemble du corpus sont considérés comme plus importants.

Le modèle `Word2Vec` repose sur l'idée que des mots ayant des significations similaires sont représentés par des vecteurs proches dans un espace vectoriel. En utilisant des techniques de réseaux de neurones, `Word2Vec` capture des relations complexes entre les mots, telles que la similarité sémantique et les relations contextuelles, ce qui n'est pas le cas de TF-IDF. Cette méthode est particulièrement efficace pour des applications telles que la recherche sémantique et l'analyse de sentiment, où il est nécessaire de comprendre le sens profond des mots et de leurs relations dans un texte.

Nous combinons les deux résultats en ajoutant les résultats de Word2vec aux résultats de TF-IDF. Cela permet d'avoir une meilleure représentation des textes pour les tâches NLP et d'avoir à la fois les mots-clés avec TF-IDF et le contexte et les synonymes avec Word2Vec.

4 Visualisation

Afin de visualiser les données, nous les projetons dans un espace à faible dimension en utilisant t-SNE et UMAP.

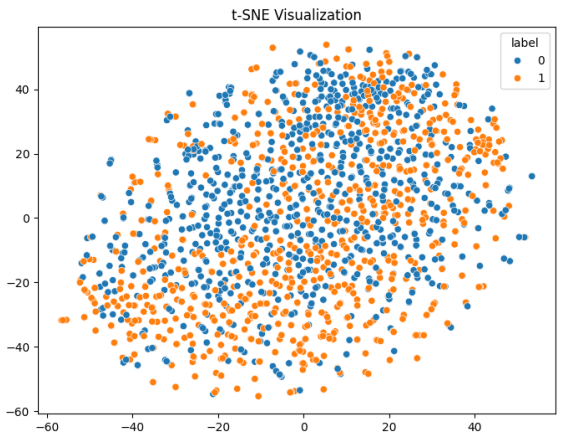

FIGURE 2 – Visualisation des données avec t-SNE

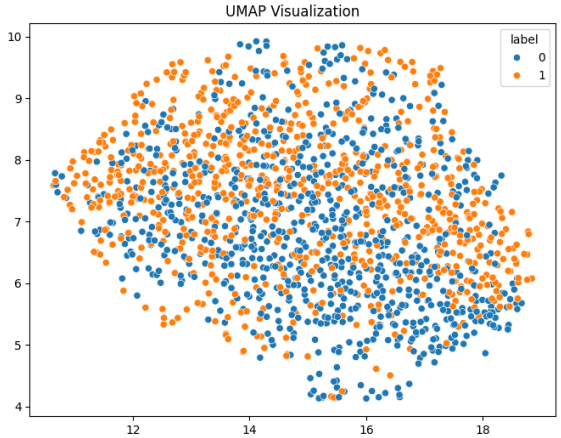

FIGURE 3 – Visualisation des données avec UMAP

Les données se regroupent un peu selon leur label, cependant les points restent assez mélangés. Il n'y a pas de séparation bien nette des deux groupes.

5 Modélisation

Pour chaque modèle, on vérifie bien qu'il n'y a pas de sur apprentissage grâce aux courbes de perte et d'accuracy pour l'entraînement et la validation.

5.1 Mixture of Experts (MoE)

5.1.1 Structure du modèle MoE :

— **Couche d'entrée** : Dimension d_{input}
— **Nombre d'experts** : $N_{\text{experts}} = 2$ (paramétrable)
— **Architecture des experts** : Réseaux à 2 couches Linear avec fonction d'activation ReLU
— **Couche de gating** : Mécanisme d'attention softmax

5.1.2 Caractéristiques Techniques

Paramètre	**Valeur**
Dimension d'entrée	d_{input}
Nombre d'experts	2 (par défaut)
Dimension cachée	256
Fonction d'activation	ReLU
Gate	softmax
Optimizer	Adam, lr=0.01
Critère	BCEWithLogitsLoss

Nous avons utilisé l'optimizer Adam et la fonction de perte *BCEWithLogitsLoss*. Elle combine une couche de sigmoïde et la perte d'entropie croisée binaire (*Binary Cross-Entropy*) en une seule opération, ce qui améliore la stabilité numérique. Elle est définie comme suit :

$$\mathscr{L}(x,y) = \frac{1}{N}\sum_{i=1}^{N}[-y_i \cdot \log(\sigma(x_i)) - (1-y_i)\cdot \log(1-\sigma(x_i))] \tag{1}$$

où :
— x (logits) est la sortie non normalisée du modèle (sans application de sigmoïde), de dimension N (nombre d'échantillons).
— y est la cible binaire (étiquette), avec $y_i \in \{0,1\}$.
— $\sigma(\cdot)$ est la fonction sigmoïde, définie par :

$$\sigma(x_i) = \frac{1}{1+e^{-x_i}} \tag{2}$$

5.1.3 Résultats

On voit que les classes sont plutôt bien prédites avec le mélange d'experts.

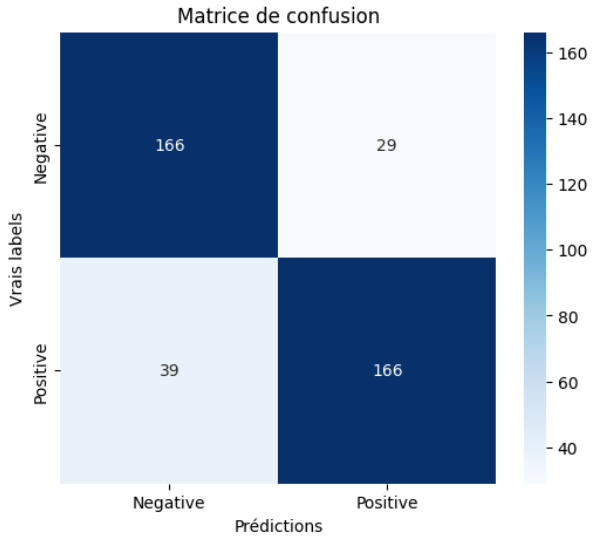

FIGURE 4 – Matrice de confusion pour le MoE

5.2 MLP

5.2.1 Structure du modèle

Le modèle implémenté est un Perceptron Multicouche (MLP) entièrement connecté avec les caractéristiques suivantes :

- **Couche d'entrée** : Dimension d_{input}
- **Couches cachées** :
  - Couche Linear 1 : 256 neurones
  - Couche Linear 2 : 128 neurones
- **Couche de sortie** : 1 neurone (classification binaire)
- **Fonctions d'activation** : ReLU entre les couches

Optimizer : Adam, lr=0.001
Critère : BCEWithLogitsLoss

5.2.2 Résultats

Avec le MLP, les résultats sont un peu mieux comme on peut le voir pour la prédiction de la classe positive.

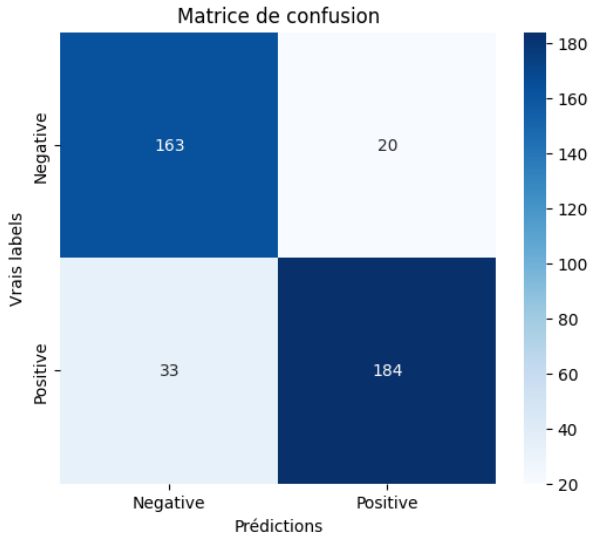

FIGURE 5 – Matrice de confusion pour le MLP

5.3 Régression logistique

On fait une régression logistique classique.

5.3.1 Résultats

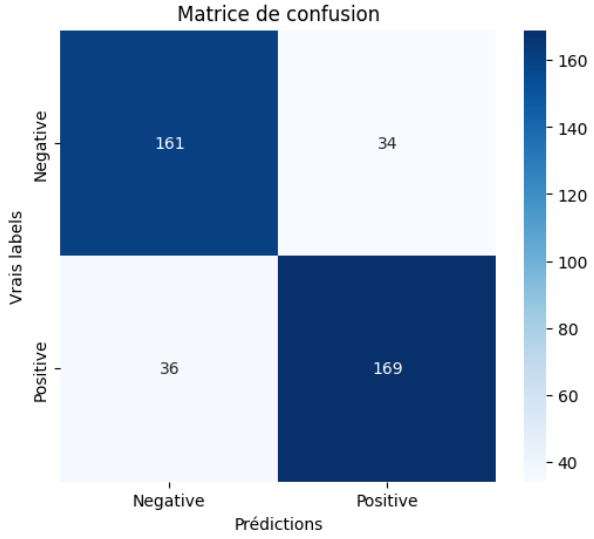

FIGURE 6 – Matrice de confusion pour la régression

Avec la régression logistique, les résultats sont aussi un peu mieux que pour le modèle MoE.

5.4 CNN

5.4.1 Structure du modèle

Le CNN est composé d'une couche de convolution avec fonction ReLu et maxpooling. On fait passer 3 filtres de taille 256.

5.4.2 Résultats

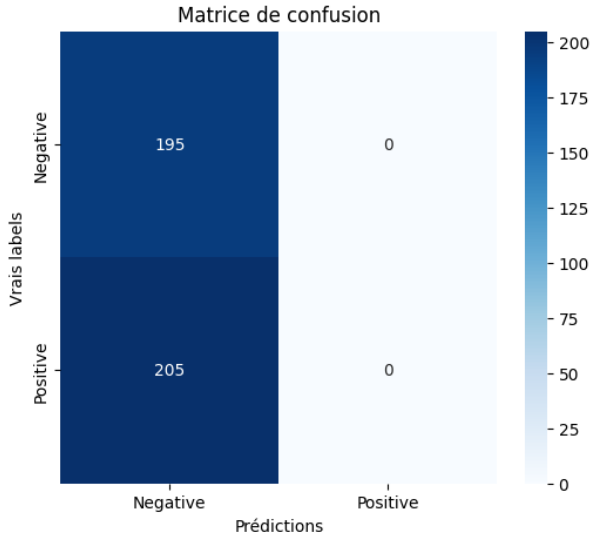

FIGURE 7 – Matrice de confusion pour le CNN

Il y a un problème avec le CNN qui classe tout d'un seul côté.

5.5 LSTM

Les **Long Short-Term Memory** (LSTM) sont un type particulier de réseau de neurones récurrents (RNN) conçus pour capturer des dépendances à long terme. Contrairement aux RNN standard, les LSTM contiennent des *portes* qui contrôlent le flux d'information.

5.5.1 Structure du Modèle

Le modèle séquentiel implémenté comprend :
— Couche d'entrée : Input$(1, n_{\text{features}})$
— Couche LSTM : 128 unités
— Dropout : 50% pour la régularisation
— Couche Dense : 64 neurones ReLU
— Dropout : 30% additionnel
— Couche de sortie : Dense$(n_{\text{classes}}, \text{softmax})$

— Optimiseur : Adam ($\alpha = 0.001$)
— Fonction de coût : Entropie croisée catégorielle sparse

$$\mathscr{L} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{y}_i) \tag{3}$$

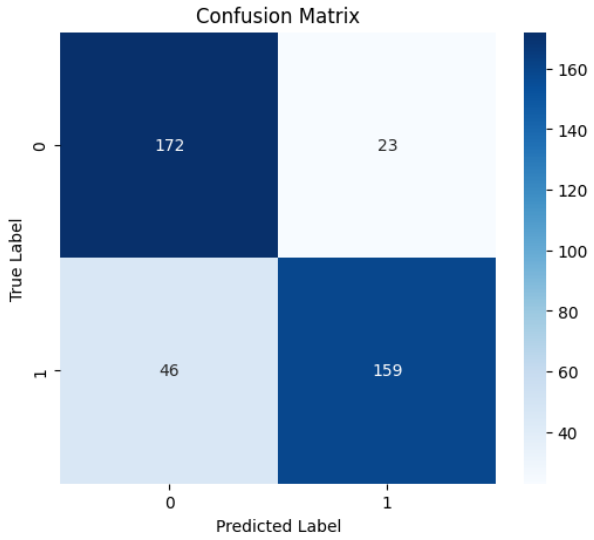

FIGURE 8 – Matrice de confusion pour LSTM

6 Comparaison des modèles

L'évaluation est réalisée avec les métriques suivantes : l'accuracy, la précision, le rappel et le F1-score.

Model	Accuracy	Precision	Recall	F1 Score
MoE	0.8050	0.843243	0.760976	0.800000
Logistic Regression	0.8250	0.832512	0.824390	0.828431
MLP	0.8300	0.851282	0.809756	0.830000
CNN	0.4875	0.000000	0.000000	0.000000
LSTM	0.8275	0.832367	0.827500	0.827177

FIGURE 9 – Comparaison des modèles

D'après le tableau, c'est le MLP qui a le meilleur résultat en termes d'accuracy, avec LSTM et la régression logistique qui ne sont pas très loin derrière.

7 Analyse

On veut savoir pour différents types de critiques s'il y a un expert qui est choisi plus qu'un autre. J'ai regardé pour les critiques qui sont longues ou courtes en faisant la moyenne des poids pour chaque classe court ou long, en définissant 0=Court (<200 mots), 1=Long (>=200 mots).

On peut voir que l'on est sur du 50/50, donc les experts n'ont pas vraiment l'air de se spécialiser. Il en est de même pour les avis positifs et négatifs.

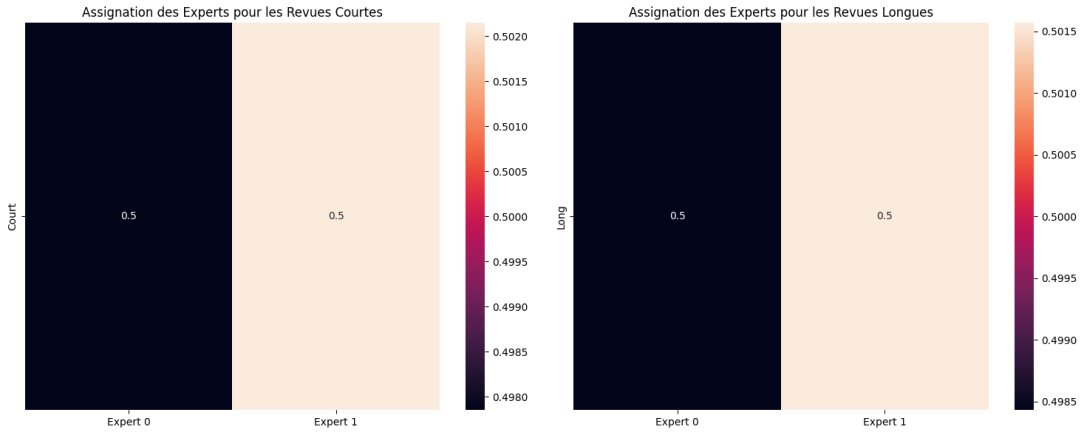

FIGURE 10 – Choix de l'expert selon la longueur de la review

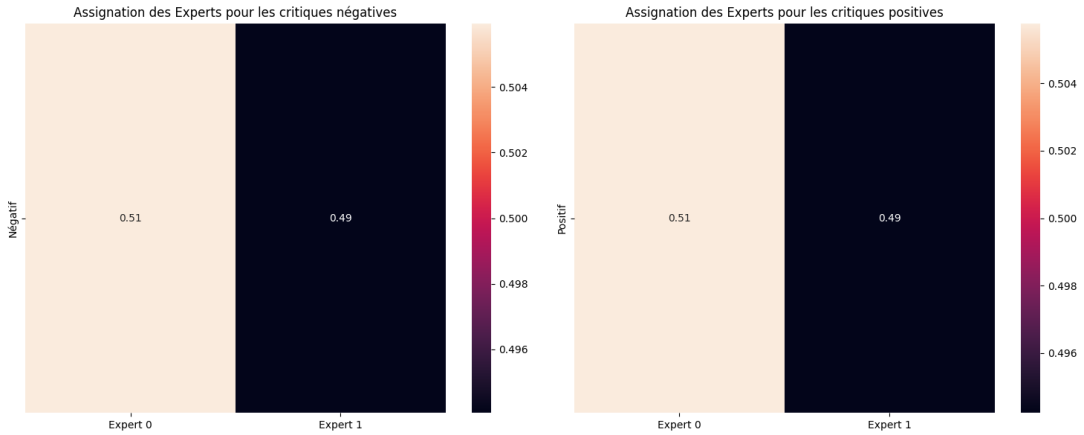

FIGURE 11 – Choix de l'expert selon avis positif ou négatif

8 Conclusion

Le MoE n'a pas forcément été le plus performant des modèles. Des pistes d'amélioration incluent l'utilisation de modèles pré-entraînés plus avancés. Peut-être que le preprocessing n'était pas assez bien fait étant donné la visualisation des données. On pourrait également changer de gate puisque c'est lui qui permet de choisir un expert ou l'autre. On pourrait prendre un gate avec plus de couches donc plus profond pour permettre de mieux distinguer les différentes features d'un expert.