

|  |  |  |
| --- | --- | --- |
| **Organisation** | : | Université Virtuelle du Burkina |
| **Filière** | : | Fouilles de Données et Intelligence Artificielle |
| **Niveau** | : | Master I |
| **Promotion** | : | 2022-2023 |

**Natural Language Processing (NLP) - Part 2**

**Résumer de texte et comparaison de méthodes**

*GitHub :* [*https://github.com/kiema97/NPL2-Final-Project.git*](https://github.com/kiema97/NPL2-Final-Project.git)

*Novembre 2024*

|  |  |
| --- | --- |
| Membres du Groupe | |
| **Nom** | **Prénom** |
| KIEMA | Wend-denda Arsène |
| OUEDRAOGO | Patednewende Mahomed |

**Chargé du cours :** Dr KABORE Kader

# Introduction

Dans ce rapport, nous comparons deux approches principales de génération de résumés de texte : l'approche extractive et l'approche abstraite. Les performances des deux méthodes ont été évaluées à l'aide des métriques standard ROUGE et BLEU. Ce document inclut une analyse quantitative et qualitative, ainsi que des réflexions sur leurs applications pratiques.

# Méthodologie

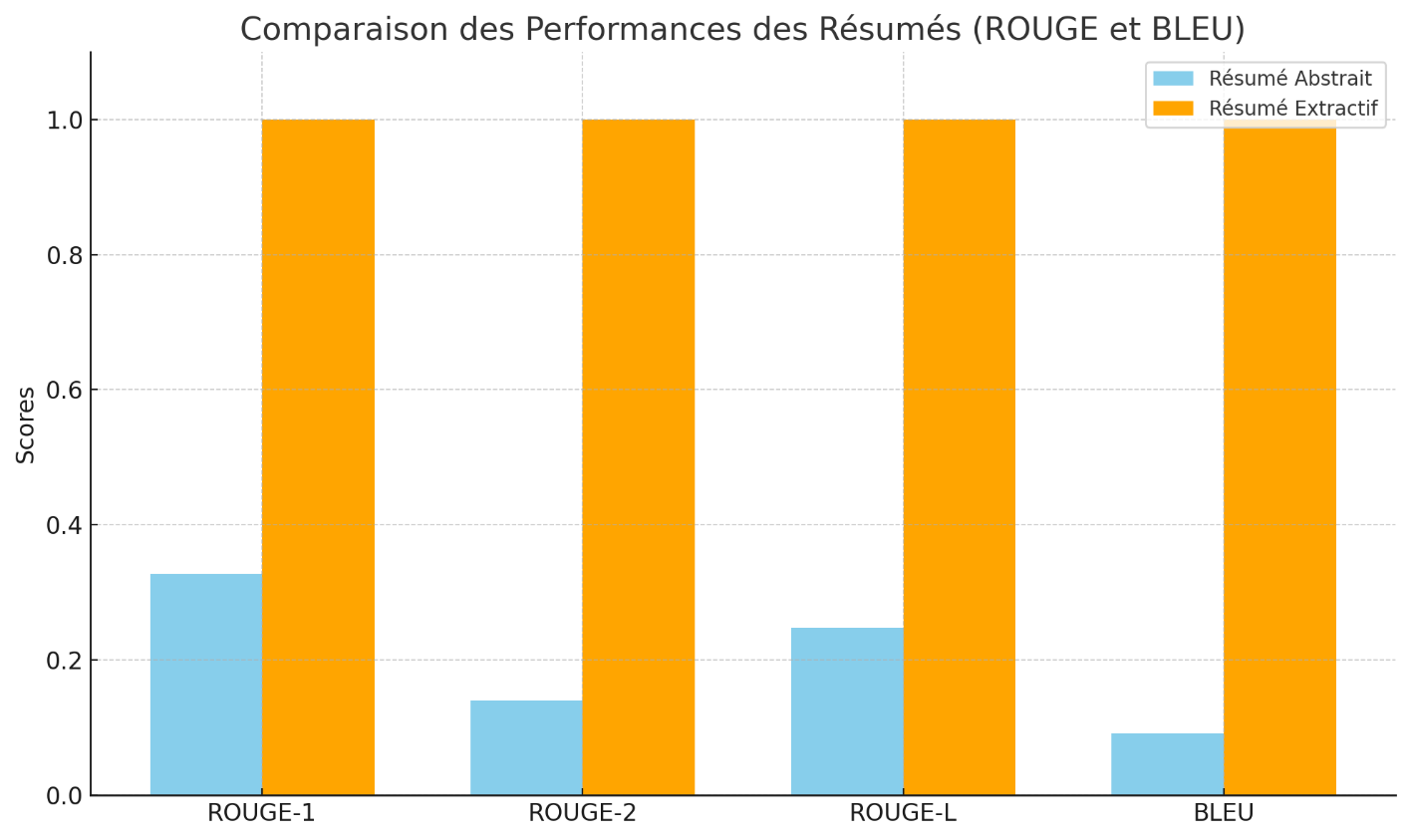
Pour cette étude, nous avons utilisé le modèle BART pour générer des résumés abstraits. L'approche extractive, quant à elle, consiste à extraire directement les phrases les plus pertinentes du texte source. Les performances des deux approches ont été comparées en utilisant :  
- ROUGE-1, ROUGE-2, et ROUGE-L  
- BLEU  
Les données proviennent du jeu de données CNN/DailyMail.

# Résultats Quantitatifs

Les scores obtenus pour les deux approches sont résumés dans le tableau ci-dessous :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Approche | ROUGE-1 | ROUGE-2 | ROUGE-L | BLEU |
| Abstraite | 0.3278 | 0.1401 | 0.2474 | 0.0921 |
| Extractive | 1.0000 | 1.0000 | 1.0000 | 1.0000 |

Un graphique comparatif des scores est présenté ci-dessous pour mieux visualiser les différences entre les deux approches.



# Analyse des Résultats

Les scores de l'approche extractive sont parfaits (1.0) pour toutes les métriques, car cette méthode extrait directement des phrases du texte source, ce qui garantit une correspondance exacte avec les références. Cependant, cette approche ne peut pas reformuler ou synthétiser l'information.

L'approche abstraite, bien que moins performante numériquement, montre sa capacité à générer des résumés originaux. Les scores ROUGE et BLEU inférieurs reflètent les défis associés à la génération de texte fluide et cohérent tout en préservant la fidélité à l'information source.

# Analyse des Limites

## **Limites des modèles pour résumés extractifs par rapport aux modèles pour résumés abstraits**

* **Approche limitée au texte existant** : Les modèles extractifs sélectionnent uniquement des phrases ou des segments directement présents dans le texte source, ce qui peut entraîner un manque de fluidité ou une redondance dans les résumés. Ils ne peuvent pas reformuler ou simplifier des concepts complexes.
* **Adaptabilité limitée** : Ils ne parviennent pas à condenser ou interpréter le contenu, ce qui les rend moins efficaces pour des textes longs ou denses. Par exemple, dans des cas où une reformulation serait nécessaire pour un lecteur général, les modèles extractifs échouent à apporter une valeur ajoutée.
* **Qualité inférieure pour des données bruitées** : Si le texte source contient des phrases mal structurées ou moins pertinentes, les modèles extractifs risquent d’inclure ces éléments, réduisant la qualité du résumé.

## **Limites des modèles encoder-decoder dans les tâches de résumé de texte**

* **Dépendance à la qualité des données d'entraînement** : Les modèles encoder-decoder, comme BART ou T5, sont fortement influencés par les données utilisées lors de l'entraînement. Si les données ne sont pas représentatives ou contiennent des biais, le modèle peut générer des résumés inappropriés.
* **Perte d'information** : Sur des textes très longs, les mécanismes d'attention ont une capacité limitée à encoder tous les détails pertinents, ce qui peut entraîner l'omission de points importants.
* **Problèmes de cohérence** : Bien que les modèles puissent produire des résumés abstraits cohérents, ils peuvent parfois inclure des informations inexactes ou inventer des détails qui ne sont pas présents dans le texte source.
* **Coût computationnel élevé** : Ces modèles nécessitent des ressources importantes en termes de mémoire et de puissance de calcul, ce qui peut limiter leur applicabilité à des environnements contraints en ressources.

## **Comment améliorer la performance du modèle sur des textes complexes ou longs ?**

* **Utilisation de mécanismes d’attention améliorés** : Des variantes comme les mécanismes d’attention hiérarchiques ou les Transformers optimisés pour des séquences longues (par exemple Longformer, BigBird) peuvent mieux gérer les textes complexes.
* **Pré-traitement des données** : Diviser les textes longs en segments significatifs, tout en conservant leur relation logique, peut aider à capturer des informations importantes sans surcharge du modèle.
* **Raffinement des hyperparamètres** : Optimiser les paramètres comme la longueur maximale des sorties, les seuils de faisceau (beam search), et les critères d’arrêt anticipé peut améliorer la qualité des résumés.
* **Enrichissement des données d'entraînement** : Utiliser des ensembles de données avec des exemples complexes ou multi-domaines peut améliorer la généralisation des modèles. Des techniques comme l'apprentissage par transfert ou l'entraînement sur des données synthétiques annotées peuvent aussi s’avérer utiles.
* **Evaluation humaine combinée aux métriques automatiques** : Les métriques comme ROUGE et BLEU ne capturent pas toujours la qualité sémantique d’un résumé. Inclure des évaluations humaines permet d'identifier des erreurs que les métriques automatiques peuvent manquer.

# Conclusion

Ce rapport a mis en évidence les forces et les limites des approches extractive et abstraite dans la tâche de résumé de texte. L'approche extractive est idéale pour les scénarios où une précision absolue est requise, tandis que l'approche abstraite convient mieux pour les cas où des résumés originaux et reformulés sont nécessaires. Le choix de la méthode dépend donc des besoins spécifiques de l'application.