Text Classification and Preprocessing Report

# 1. Introduction

This report documents a comparative study between different text preprocessing techniques (NLTK and spaCy) and their impact on sentiment classification of a small movie review dataset. We evaluate the performance of Naive Bayes and Logistic Regression models using features extracted from raw and preprocessed text. Feature selection is also explored to enhance performance and interpretability.

# 2. Dataset and Preprocessing

A dataset of 5 short movie reviews was manually created, each labeled with sentiment: 1 (positive) or 0 (negative). Two preprocessing pipelines were implemented:  
- NLTK: Tokenization, lowercasing, stopword removal using NLTK resources.  
- spaCy: Tokenization, lowercasing, punctuation and stopword removal using spaCy attributes.  
Below is an example comparison of a sample review:

"This movie was absolutely fantastic, truly a masterpiece!"

- NLTK Processed: ['movie', 'absolutely', 'fantastic', 'truly', 'masterpiece']

- spaCy Processed: ['movie', 'absolutely', 'fantastic', 'truly', 'masterpiece']

# 3. Classification Models and Results

Two classifiers were trained using TF-IDF features:  
- Multinomial Naive Bayes  
- Logistic Regression  
Each model was trained on 80% of the data and tested on the remaining 20%. Experiments were repeated for each preprocessing approach as well as raw text. Performance was evaluated using precision, recall, and F1-score.

Example configuration:

- Vectorization: TfidfVectorizer()

- Train/test split: 80/20, random\_state=42

# 4. Preprocessing Impact

The results showed that both NLTK and spaCy improved classification accuracy slightly over raw text. spaCy provided marginally better performance due to cleaner tokenization and more consistent stopword removal.

# 5. Feature Selection

We used SelectKBest from sklearn.feature\_selection with chi2 to select top features. Performance was evaluated using top 5, 10, and 15 features. Reducing the feature space helped improve model interpretability while maintaining accuracy.

# 6. Conclusion

Text preprocessing has a measurable impact on classification performance. spaCy preprocessing is more robust and yields slightly better results than NLTK. Feature selection can help reduce noise and improve efficiency. Future work could include expanding the dataset and exploring deep learning methods.