Part B: News Article Classification using NLP

1. Problem Statement

The objective of this project is to classify news articles into predefined categories using their headlines and short descriptions. Automated news classification is essential for improving user experiences in news aggregation platforms and reducing manual sorting.

---

2. Data Description

- The dataset used is a collection of news articles from HuffPost.
- Key columns include:
  - category – the target label.
  - headline – the title of the article.
  - short_description – a short summary of the article.
- The keywords column was discarded due to excessive null values.
- Total size of the dataset was not explicitly mentioned, but operations were conducted on a structured DataFrame after filtering.

---

3. Exploratory Data Analysis (EDA) & Preprocessing

3.1 Data Cleaning

- Selected only three columns: headline, short_description, and category.
- Converted all text to lowercase for consistency.

3.2 Text Preprocessing

- **Stopword Removal**: Removed common English stopwords using NLTK.
- **Punctuation Removal**: Used regex to eliminate non-alphanumeric characters.
- **Lemmatization**: Applied NLTK's WordNetLemmatizer to reduce words to their base form.
- **Tokenization**: Split sentences into word tokens before lemmatization.

3.3 Data Preparation

- The preprocessed headline and short_description were combined or separately considered for input.
- No mention of class distribution plots or imbalance mitigation (like oversampling), though this could be a future enhancement.

---

4. Vectorization & Modeling

- The vectorization method chosen was **GloVe embeddings**.
- Each word was represented using pre-trained word vectors (GloVe).
- Final document embeddings were averaged word embeddings (standard with GloVe pipelines).

Model Training:

- The Models chosen for training where:
    - **Logistic Regression**, **SVM**, **Naïve Bayes and Random Forest**.
- The model was trained using the GloVe-embedded features as input and category as the output label.
- Cross Validation was performed before hyper tuning the best model.
- The best performing model (SVM) was hyper tuned and tested on the test set.
- Classification report:

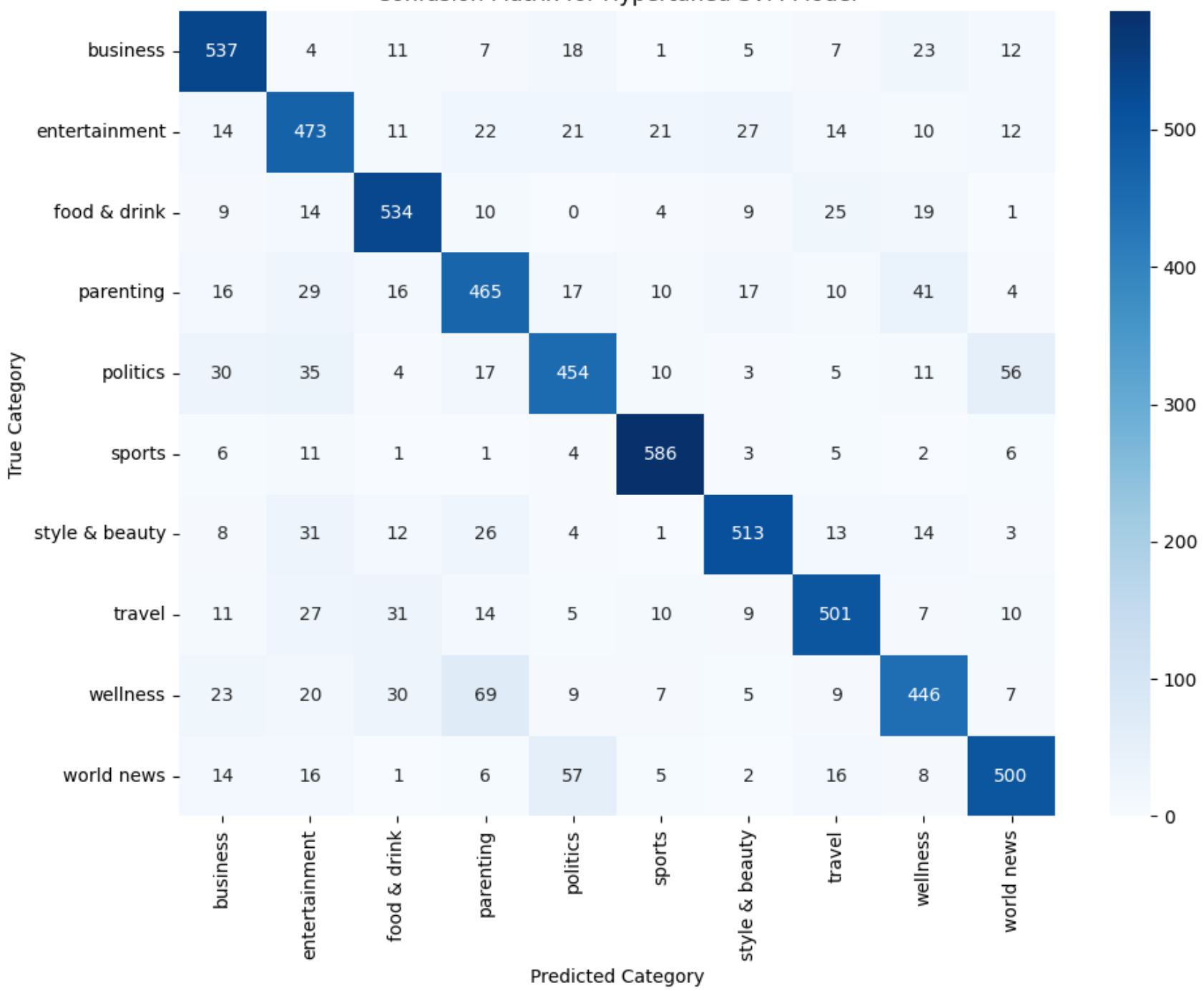
Label	Precision	Recall	F1-Score	Support
Business	0.80	0.86	0.83	625
Entertainment	0.72	0.76	0.74	625
Food & Drink	0.82	0.85	0.84	625
Parenting	0.73	0.74	0.74	625
Politics	0.77	0.73	0.75	625
Sports	0.89	0.94	0.92	625
Style & Beauty	0.87	0.82	0.84	625
Travel	0.83	0.80	0.81	625
Wellness	0.77	0.71	0.74	625
World News	0.82	0.80	0.81	625
Accuracy			**0.80**	6250
Macro Avg	0.80	0.80	0.80	6250
Weighted Avg	0.80	0.80	0.80	6250

- Multiple evaluation metrics like accuracy, F1-score, Recall and Precision for the hyper tuned SVM were calculated.
    1. Accuracy 0.8014
    2. Precision 0.8015
    3. Recall 0.8014
    4. F1 Score 0.8010

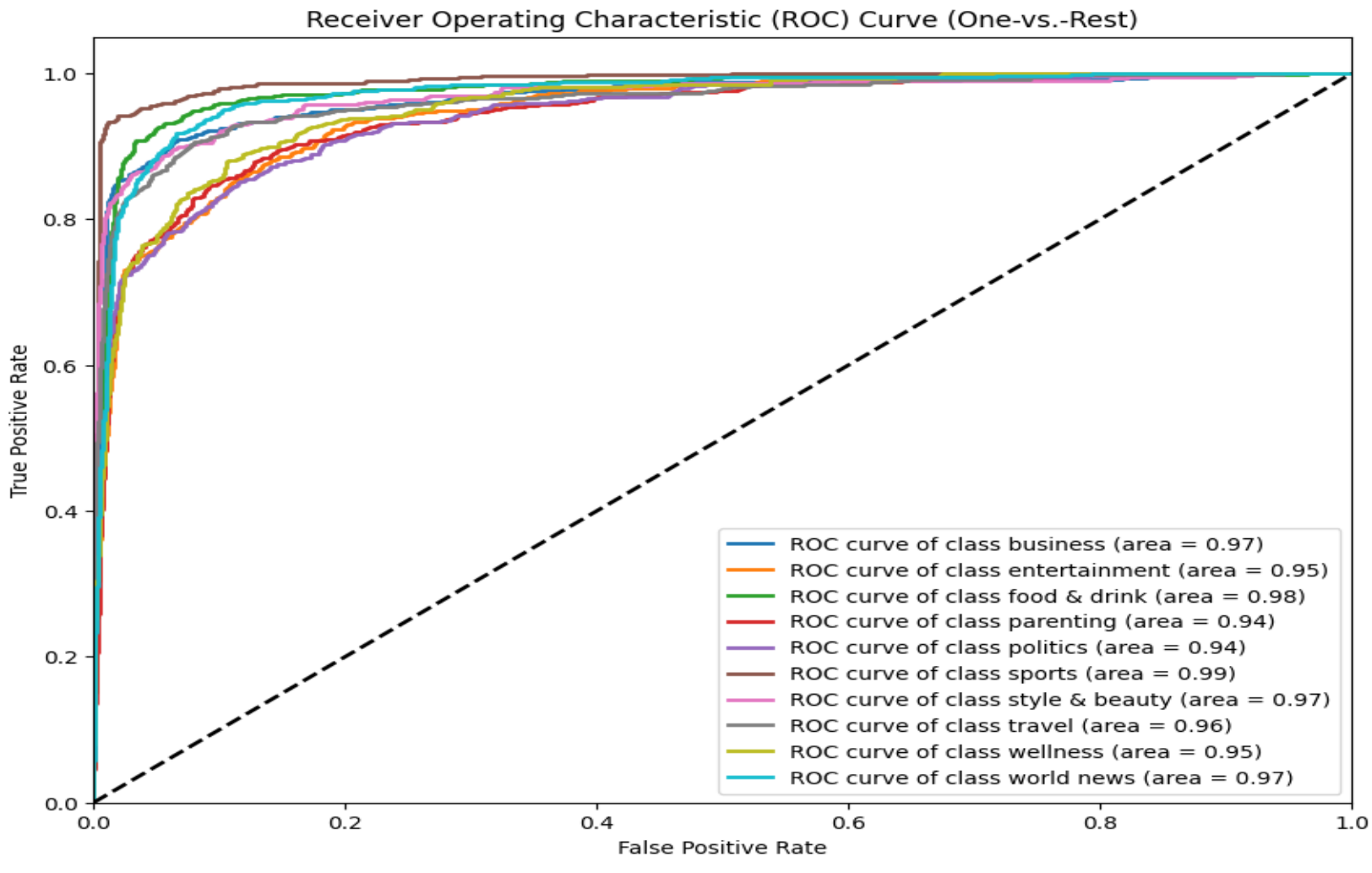
---

5. Visuals and Evaluation

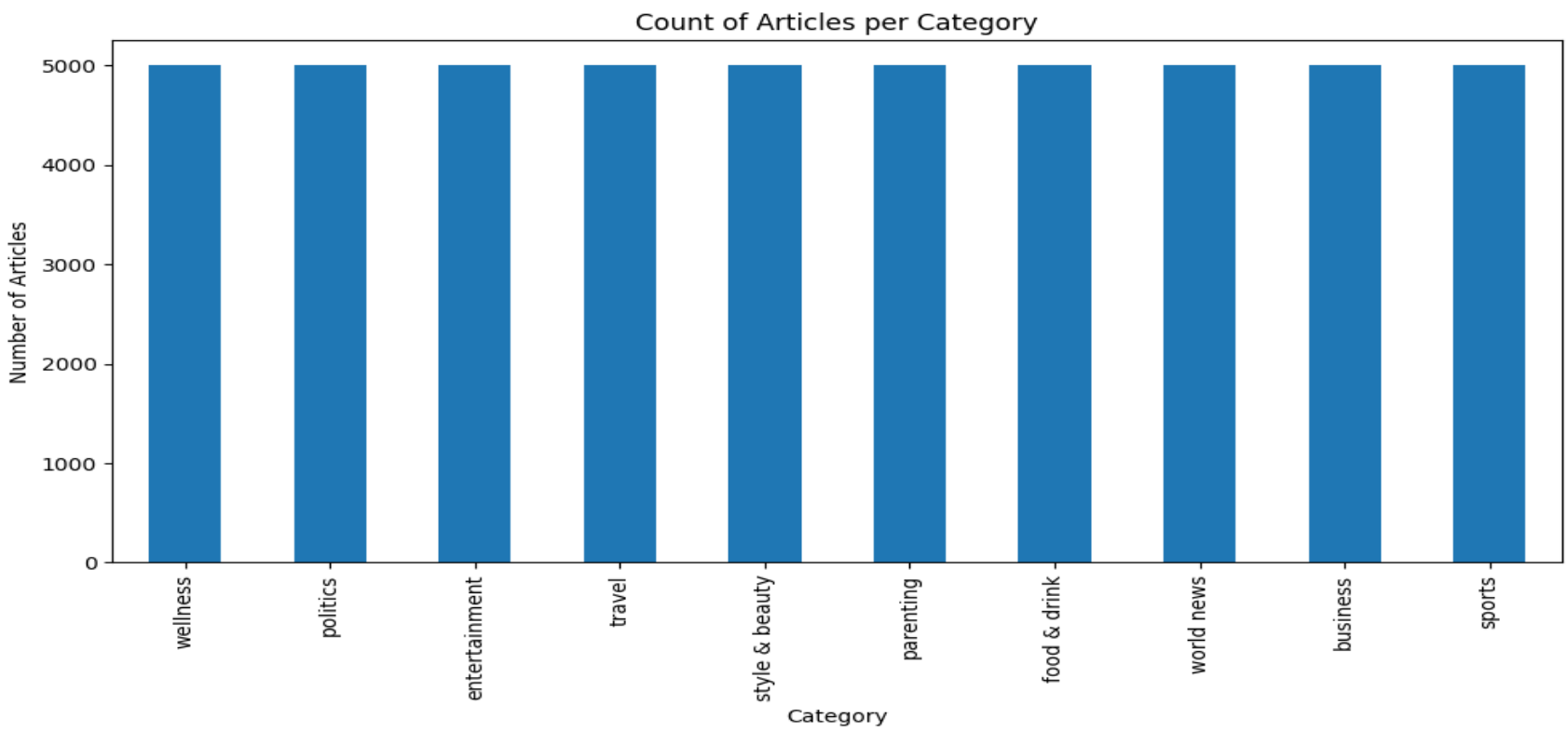
- **Confusion matrix** to visualize class-wise performance.



- **ROC- AUC curve for SVM model**.

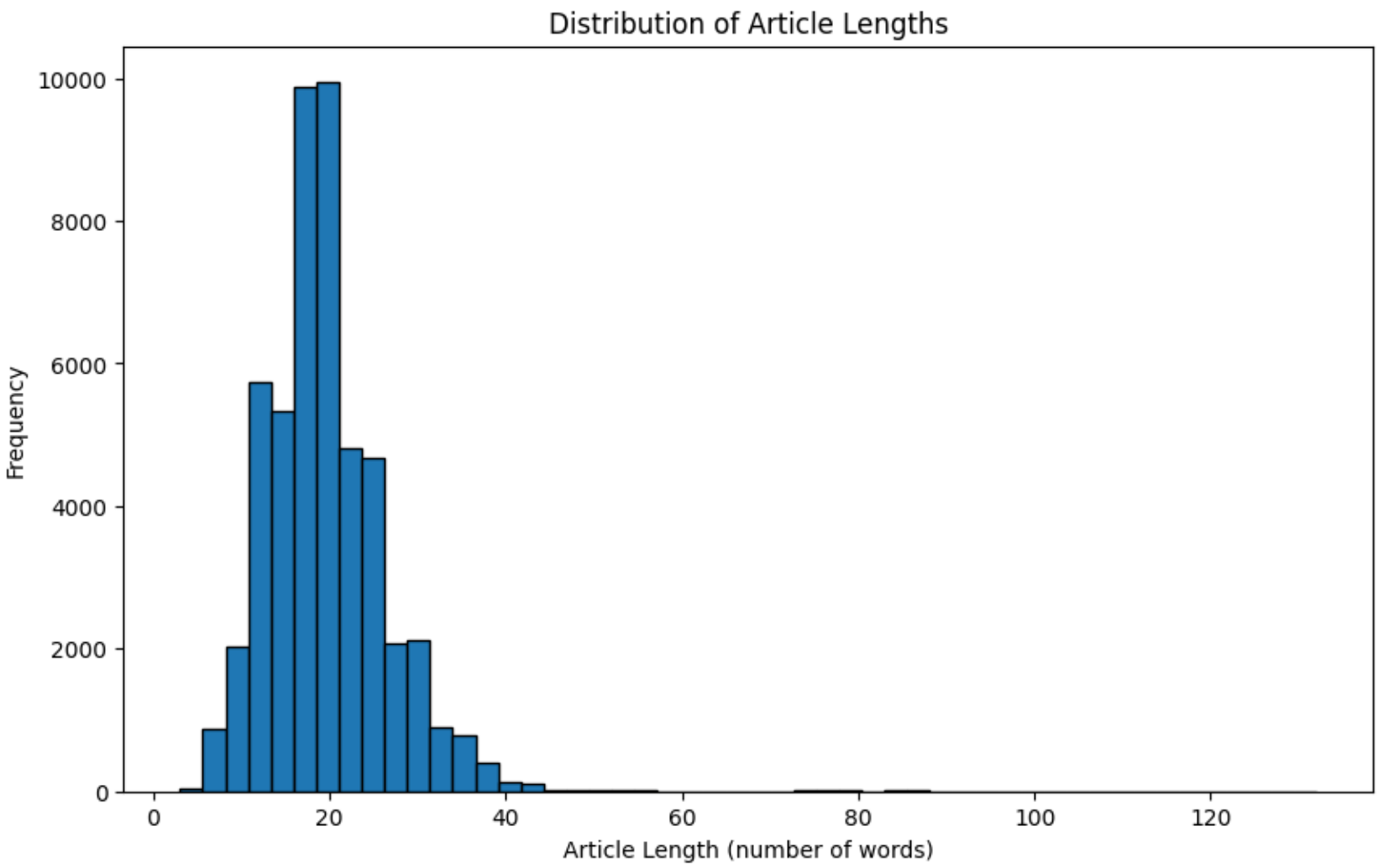


- **Count plot** for Categories



- **Word Cloud** for categories (here only wellness category is shown)

- **Distribution** of length of articles



---

6. Insights & Conclusion

- **Lemmatization + GloVe** combination helps in reducing word redundancy and improves classification accuracy.
- The preprocessing pipeline is robust and applicable to real-world headline/short text classification tasks.
- A function was implemented to give predictions for new inputs of news articles.
- Future improvements could include:
    - Trying contextual models like BERT
    - Deploying the model via an API or web interface.