

AG NEWS Category Classification Using NLP

Milestone 3: Project – Preprocessing and Transformation

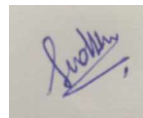
Student: Sudhish Subramaniam

+12368634776

subramaniam.su@northeastern.edu

Percentage of Effort Contributed by Student : 100%

Signature of Student :



Submission Date: _ : 22nd October, 2023

Introduction

In the rapidly evolving landscape of digital news consumption, the efficient categorization of news articles has become a paramount challenge for news platforms. The objective at hand is the development of a robust Natural Language Processing (NLP) model, a technological solution aimed at automating the categorization of news articles into distinct topics, namely "World," "Sports," "Business," and "Sci/Tech." The significance of this task lies in its potential to revolutionize the news industry by enhancing user experience and streamlining content recommendation processes.

In an era where information is abundant and attention spans are limited, ensuring that readers can effortlessly access news content that aligns with their interests is a pressing concern. My mission is to create a sophisticated NLP model capable of classifying news articles with precision, relying on the nuanced analysis of their content and context. This classification is not merely an organizational endeavor but a pivotal means of optimizing content delivery and personalizing news recommendations. By automating this intricate process, I aim to empower news platforms to engage readers more effectively, potentially leading to increased user retention and overall satisfaction. In doing so, I embark on a journey to redefine how news is presented and consumed in the digital age.

Problem Statement

The problem at hand is to automate the classification of news articles into predefined categories, specifically "World," "Sports," "Business," and "Sci/Tech." This task is essential for streamlining content organization and personalizing news delivery on digital platforms, ultimately improving user engagement and satisfaction.

Data Used

Dataset Overview

The dataset at the core of this project is the AG News Corpus, a meticulously curated collection of news articles derived from an extensive pool of over 1 million news articles sourced from more than 2000 news outlets. This vast and diverse collection of articles has been accumulated over the course of a year through the efforts of the ComeToMyHead academic news search engine, which has been in operation since July 2004.

Specifically, the dataset used for this project is the AG's news topic classification dataset, curated by Xiang Zhang. This dataset has gained prominence as a benchmark for text classification tasks and serves as a well-established resource for addressing the news categorization problem. It comprises 30,000 training samples and 1,900 test samples per class. The dataset's substantial size

and the balanced distribution of samples across categories make it an invaluable resource for training, fine-tuning, and evaluating Natural Language Processing (NLP) models, which are central to solving the news article categorization challenge.

Data Dependency

- **Class Balance Dependency:** The dataset's balanced distribution across "World," "Sports," "Business," and "Sci/Tech" categories is essential to prevent bias.
- **Textual Dependency:** The model relies on keywords and patterns in the news articles for categorization. Text content strongly influences predictions.
- **Train Test Quality Dependency:** Both training and test sets must be of high quality and representative to achieve reliable model performance.
- **Temporal Influence:** Temporal trends or seasonality in news topics may impact categorization accuracy.
- **Feature Influence:** The choice and quality of features (e.g., word embeddings) depend on the dataset's content.
- **Evaluation Metric Choice:** The dataset's class distribution affects the selection of appropriate evaluation metrics.

Managing these dependencies will ensure robust and accurate news article categorization.

Analysis

Data Preprocessing and transformation

The objective of this project is to automate the classification of news articles into predefined categories, namely "World," "Sports," "Business," and "Sci/Tech." This automated classification is crucial for improving content organization and personalizing news delivery on digital platforms, aiming to enhance user engagement and satisfaction.

Data preprocessing plays a vital role in any natural language processing (NLP) task. In this project, I carried out several preprocessing steps to prepare the news article data for classification. These steps included converting all text to lowercase, removing punctuation and numbers, eliminating

common English stopwords, applying lemmatization to reduce word dimensionality, tokenizing the text, and padding sequences for uniform length.

Following data preprocessing, I built a machine learning model for news article classification. The model utilized a vocabulary size of 10,000 and an embedding size of 32 to create word-level representations of the text data. This embedding layer helps capture semantic relationships between words in news articles.

In conclusion, I have successfully preprocessed and transformed news article data, making it ready for classification. Our machine learning model, which uses word embeddings and sequence padding, is prepared to classify news articles into "World," "Sports," "Business," and "Sci/Tech" categories. This classification will improve content organization and personalization on digital platforms, ultimately enhancing user engagement and satisfaction.

Explanatory data analysis

The dataset for this analysis consists of 120,000 samples in the training set and 7,600 samples in the test set, each comprising two columns: 'text' and 'label.' The 'label' column represents the category of news, with each of the four classes (0, 1, 2, and 3) containing 30,000 samples. This can be seen in Fig. 1. This balanced class distribution ensures a fair representation of different news categories. Moreover, there are no missing values in either the training or test dataset, indicating data integrity.

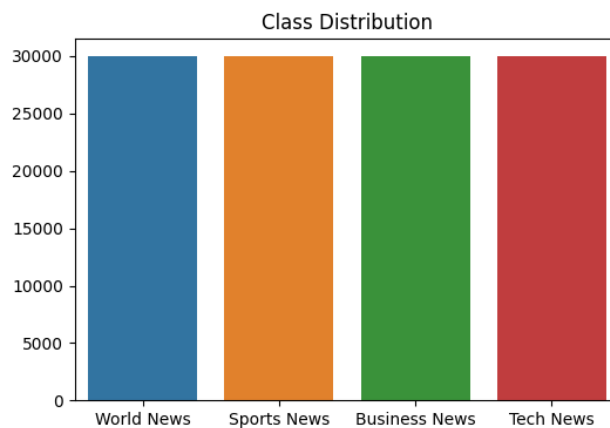


Fig. 1.

Word cloud visualizations were created for each news category, revealing the most common terms associated with each. In the 'World News' category, terms such as "prime minister," "iraq," and "israel" were prevalent. 'Sports News' featured terms like "game," "season," and "team." In the

'Business News' category, words like "company," "price," "oil," and "stocks" dominated. 'Science and Technology News' highlighted terms such as "microsoft," "google," "email," and "internet." These insights provide a preliminary understanding of the vocabulary within each category. These prevalent words are clearly seen in Fig. 2 (a), (b), (c), (d).

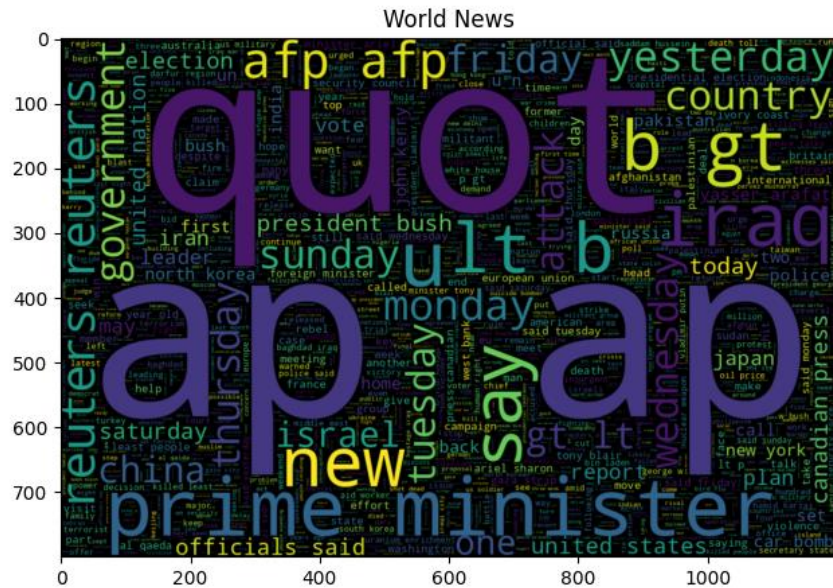


Fig 2 (a). Word Cloud of World News

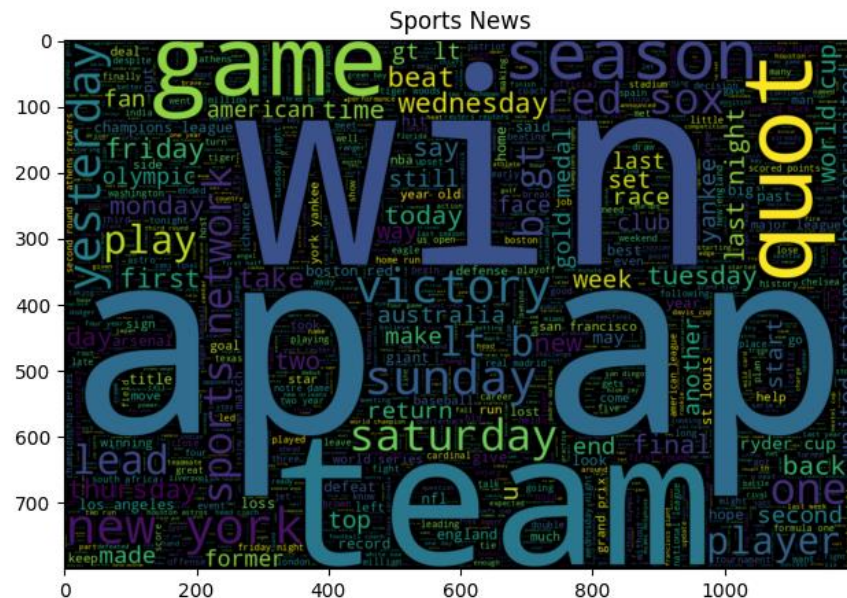


Fig 2 (b). Word Cloud of Sports News

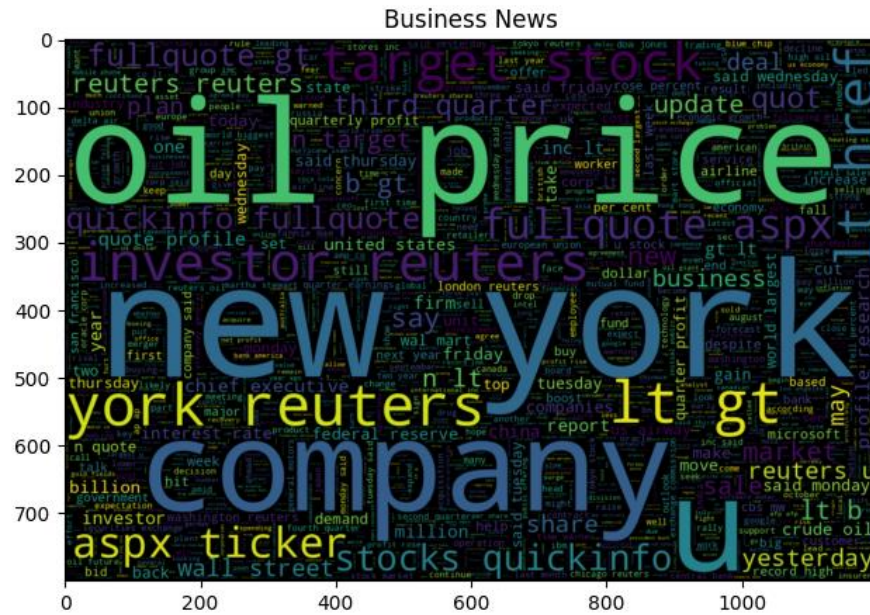


Fig 2 (b). Word Cloud of Business News

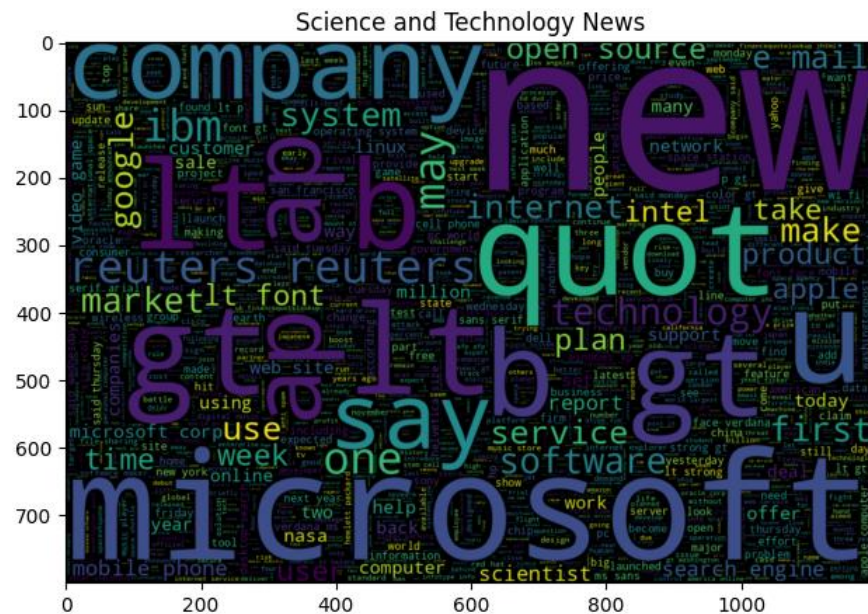


Fig 2 (d). Word Cloud of Science and Technology News

A Word2Vec model was trained on the text data to capture word embeddings. The model used a vector size of 100, a window of 5, and a minimum word count of 1. This model allows for the exploration of semantic relationships between words. For example, words like "prime" included

"shivraj," "keyuraphan," and others. Similarly, words related to "game" were "games," "play," and others. For "company," terms like "giant" and "firm" were identified. For “Microsoft”, terms like “windows”, “microsofts”, “oracle” were identified. The model can be a valuable resource for understanding word associations within the dataset. This can be seen in Fig. 3.

Words similar to 'prime': keyuraphan: 0.8244876265525818 pisanu: 0.7960994243621826 shivraj: 0.7827610373497009 giuseppe: 0.782672643661499 bijan: 0.7645360827445984 erakat: 0.7593415975570679 khorram: 0.7448946833610535 saeb: 0.7397149801254272 lapierre: 0.7395654320716858 shoichi: 0.7367920279502869	Words similar to 'game': games: 0.7263332009315491 play: 0.5799111723899841 tragicomedy: 0.5769765973091125 opus: 0.5761773586273193 franchise: 0.5448745489120483 matchup: 0.5407401919364929 postseason: 0.5395525693893433 accelerators: 0.5255386233329773 opener: 0.5225147604942322 madden: 0.5166462659835815	Words similar to 'company': giant: 0.78047776222229 firm: 0.7487821578979492 companies: 0.654605507850647 maker: 0.6283325552940369 unit: 0.6136001944541931 supplier: 0.6097885370254517 acquisition: 0.5923078656196594 companys: 0.5917797684669495 subsidiary: 0.5865916609764099 assets: 0.5835081934928894
Words similar to 'microsoft': windows: 0.6695334315299988 sp: 0.6645975708961487 xp: 0.6510486602783203 aol: 0.6262804269790649 microsofts: 0.6229841709136963 longhorn: 0.6201565265655518 symantec: 0.6129153966903687 oracle: 0.6039695739746094 telephoneprovider: 0.6029330492019653 vied: 0.5923817157745361		

Fig. 3. Similar Words using Word2Vec

Text length analysis was conducted, calculating word count, character count, and average word length for each sample in the training and test datasets (shown in Fig. 4). Correlation analysis (Show in Fig. 5) between these text length features, and the 'label' column indicated very weak correlations, suggesting that text length may not be a strong predictor of news category. This observation highlights the importance of other features in classifying news.

index	text	label	word_count	char_count	avg_word_length
0	feats n pension talks unions representing workers turner newall say disappointed talks sticken parent firm federal mogul	2	17	121	7.117647058823529
1	race second private team sets launch date human spaceflight space com space com toronto canada second team rocksteers competing million ansari x prize contest privately funded suborbital space flight officially announced first launch	3	36	252	7.0
2	ky company wins grant study peptides ap ap company founded chemistry researcher university louisville grant develop method producing better peptides short chains amino acids building blocks proteins	3	27	190	7.333333333333333
3	prediction unit helps forecast wildfires ap ap barely dawn mika fitzpatrick starts shift blur colorful maps figures endless charts already knows day bring lightning strike places expects winds pick moist places dry flames roar	3	34	226	6.647058823529412
4	calif aims limit farm related smog ap ap southern california smog fighting agency went emissions bovine variety friday adopting nation first rules reduce air pollution dairy cow manure	3	28	184	6.571428571428571

Fig. 4. Head rows with word count, character count and average word length for each news

```
<ipython-input-40-2891a86a63b>
correlation_matrix = train_
label                1.000000
word_count           0.008374
char_count           0.005026
avg_word_length      -0.009792
Name: label, dtype: float64
```

Fig. 5. Correlation between news and other columns

In the NER Analysis, I have observed the following:

1. The dataset seems to contain information about various organizations, including technology companies like Microsoft, Google, IBM, and others such as Reuters, NASA, and the United Nations. This indicates that the news articles might be related to developments in these organizations.
2. The presence of individuals like George W. Bush, Vladimir Putin, and Michael Phelps suggests that the news articles may cover political figures, celebrities, or prominent individuals in various fields.
3. Locations and countries such as Iraq, New York, and China are recognized. This suggests that the news dataset may include articles on international affairs and global events.
4. The identification of dates and times like "Tuesday," "August," and "last week" implies that the news articles might be categorized based on the time of occurrence or publication.

Based on this NER analysis, I can infer that the news dataset likely covers a wide range of topics, including technology, politics, international affairs, and events that occurred at specific times. Categorizing the news articles into these four categories could provide the model a structured approach for analyzing and organizing the dataset.

In summary, the analysis provides insights into the dataset's composition, word cloud visualizations for different news categories, and the training of a Word2Vec model to understand word similarities. The examination of text length features suggests that these factors may have limited predictive power for news category classification.

Feature engineering and feature selection

Feature engineering is the process of transforming raw text data into numerical features, enabling the training of machine learning models. The initial step involves using the TfidfVectorizer from scikit-learn to convert the text data into TF-IDF vectors, which serve as essential features. However, one noteworthy consideration is the limitation on the number of features due to system

constraints. The code imposes a cap of 4500 features to avoid system crashes, a common occurrence when dealing with high-dimensional data.

Following the TF-IDF vectorization, the code goes on to extract the vocabulary and calculates the word frequencies in the training dataset. The term frequencies are particularly informative as they reveal which words or terms hold the most significance within the dataset. Sorting the features by frequency in descending order, the code presents the top 50 features, shedding light on the most influential terms. This step is pivotal for feature selection and model interpretability.

Some most important features include: “new”, “said”, “world”, “company”, “Microsoft”, “iraq”, “oil”. These features are relevant to the world news and can be considered as most important features during modelling. This shows that the text is rightly processed.

To make the features ready for machine learning, the TF-IDF vectors are transformed into arrays and then converted into Pandas DataFrames. These DataFrames house the features for both the training and testing datasets. In conclusion, this code snippet emphasizes the importance of feature engineering in preparing text data for machine learning. By setting a limit of 4500 features, it effectively addresses system constraints, ensuring the stability of the computational environment.

References

1. Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level Convolutional Networks for Text Classification. Papers with Code. Retrieved from <https://paperswithcode.com/dataset/ag-news>
2. Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level Convolutional Networks for Text Classification. ArXiv. /abs/1509.01626
3. Havrillant L, Kreinovich V (2017) A simple probabilistic explanation of term frequency-inverse document frequency (TF-IDF) heuristic (and variations motivated by this explanation). Int J Gen Syst 46(1):27–36
4. Kim, SW., Gil, JM. Research paper classification systems based on TF-IDF and LDA schemes. Hum. Cent. Comput. Inf. Sci. 9, 30 (2019). <https://doi.org/10.1186/s13673-019-0192-7>
5. Ashwin N. (Jul 26, 2022). "Creating a TF-IDF Model from Scratch in Python." Medium. Retrieved from <https://medium.com/@ashwinnaidu1991/creating-a-tf-idf-model-from-scratch-in-python-71047f16494e>.

6. Bafna P, Pramod D, Vaidya A (2016) Document clustering: TF-IDF approach. In: IEEE int. conf. on electrical, electronics, and optimization techniques (ICEEOT). pp 61–66