New York Times Archives Data Analysis and Visualization using Prefect, DuckDB, and Streamlit

Github: **https://github.com/katiehutc/nytimes**

The Problem:

The core challenge was to extract meaningful and actionable insights from a massive, unstructured dataset of over 400,000 New York Times news headlines and abstracts spanning a decade (2015–2024). Traditional approaches often rely on brittle, dictionary-based keyword searches or require manual, time-consuming tagging. The goal was to overcome this by automatically discovering the underlying, shifting media narrative over time and making it accessible for trend analysis.

The Approach

The solution employed a modern, BERT-based topic modeling technique to automatically identify cohesive clusters of documents (topics). This unsupervised machine learning approach was integrated into a robust data pipeline. The resulting architecture focused on **scalability**, which is handling the large dataset through batch processing.

Architecture Overview

The system utilizes a three-tier architecture: **Orchestration**, **Processing**, and **Presentation**.

Orchestration Layer (Prefect)

- **Function:** Manages the entire data workflow, defining tasks and dependencies.
- **Tasks:** Responsible for data ingestion, model fitting, and batch transformation.
- **Benefit:** Provides logging, monitoring, and robust error handling for the entire pipeline.

Processing Layer (BERTopic & DuckDB)

- **Core Engine: BERTopic** uses Sentence-BERT embeddings to generate dense, contextual representations of each article, clustering them into high-quality topics.
- **Scalability:** Due to memory constraints with 400,000 documents, the transformation step was broken into smaller chunks (batches) to prevent memory overflow during the heavy embedding process.

- **Data Storage: DuckDB** serves as the analytical storage layer, receiving the transformed data (`topic_id`, `topic_name`, `date`) from the pipeline. Its fast, in-process analytical queries are ideal for subsequent dashboard visualization.

Presentation Layer (Streamlit)

- **Function:** An interactive dashboard that connects directly to the DuckDB file.
- **Visualization:** Displays topic volume over time, allowing end-users to filter and analyze the temporal trends of individual or multiple topics.

Technologies Used

- Prefect: manages the workflow, ensuring resilient data processing and error handling.
- Pandas: for efficient data manipulation.
- BERTopic: clusters semantic meaning from headlines to discover topics
- DuckDB: provides a high-performance, serverless SQL database optimized for analytical queries
- Streamlit and Plotly: power an interactive frontend dashboard for real-time trend exploration.
- Conda/Mamba: Manages the project's virtual environment and ensures reproducible dependency handling.

4. Detailed Explanation of How It Works

Clear Pipeline Flow: Ingestion → Processing → Storage → Analysis. The entire project operates as a sequential Data Pipeline orchestrated by Prefect.

Phase 1: Topic Modeling and Transformation (The Pipeline)

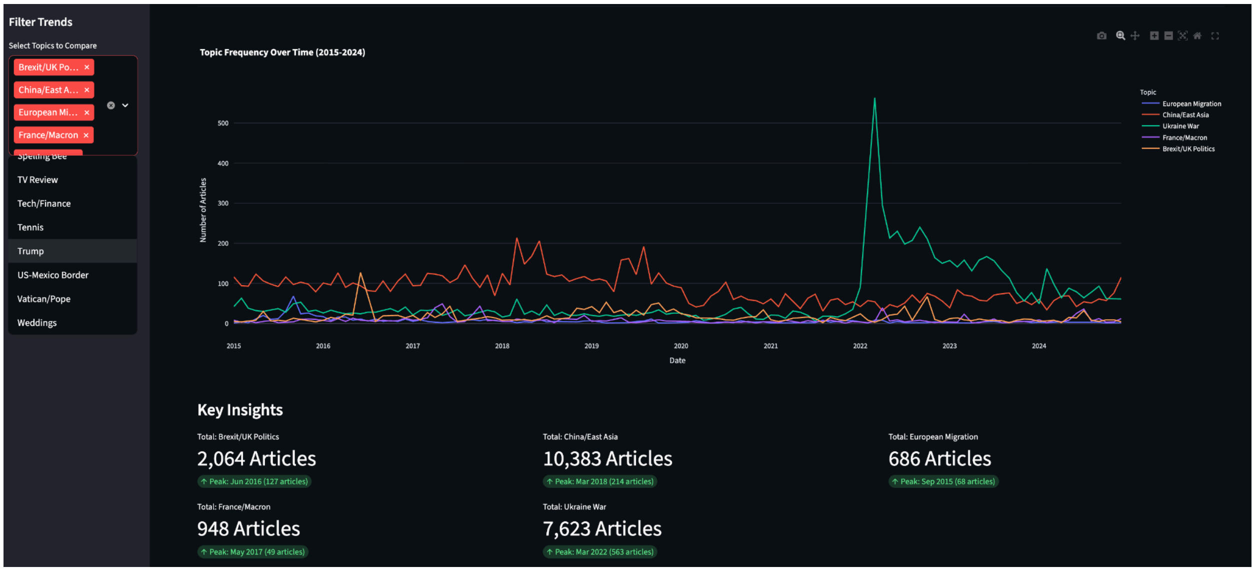
1. **Ingestion** (Raw Data Acquisition)
   - Source: Initial JSON files containing NYT headlines and abstracts (2015–2024).
   - Action: The Prefect flow loads the raw data into a Pandas DataFrame.
   - Output: Unstructured text data and associated metadata (date, headline).
2. **Model Training** (Processing): A smaller, representative sample of the documents (5% or 20,000 articles) is selected for model training. The **BERTopic** model is fitted to this sample.
   - **Embedding:** Documents are converted into dense numerical vectors using Sentence-BERT.
   - **Clustering:** UMAP is used for dimensionality reduction, followed by HDBSCAN for hierarchical clustering to identify topics.

3. **Topic Renaming:** The model is manually updated with clear, human-readable labels (e.g., 'COVID-19', 'Ukraine War') via SQL to replace the machine-generated keyword labels.
4. **Batch Transformation (Task):** The full dataset (400,000 documents) is processed in chunks (e.g., 5,000 articles per batch). For each chunk:
   - The pre-trained BERTopic model uses the transform() method to assign a topic_id and the corresponding topic_name to every article in the batch.
   - The transformed batch, including the original date and the new topic metadata, is immediately appended to the **processed_articles** table in the DuckDB file.

Phase 2: Interactive Visualization and Analysis

1. **Dashboard Launch:** The Streamlit application connects directly to the fully populated nyt_analysis_2.duckdb file.
2. **Data Query:** The app executes highly optimized SQL queries (using SELECT DISTINCT and filtering by topic_id) to retrieve the article counts grouped by topic and date.
3. **Visualization:** Matplotlib/Plotly is used to render the time-series data, allowing users to select multiple topics and instantly visualize how the media's focus on those topics evolved from 2015 to 2024.

Final Results



Key Insights

Total: Brexit/UK Politics
2,064 Articles
↑ Peak: Jun 2016 (127 articles)

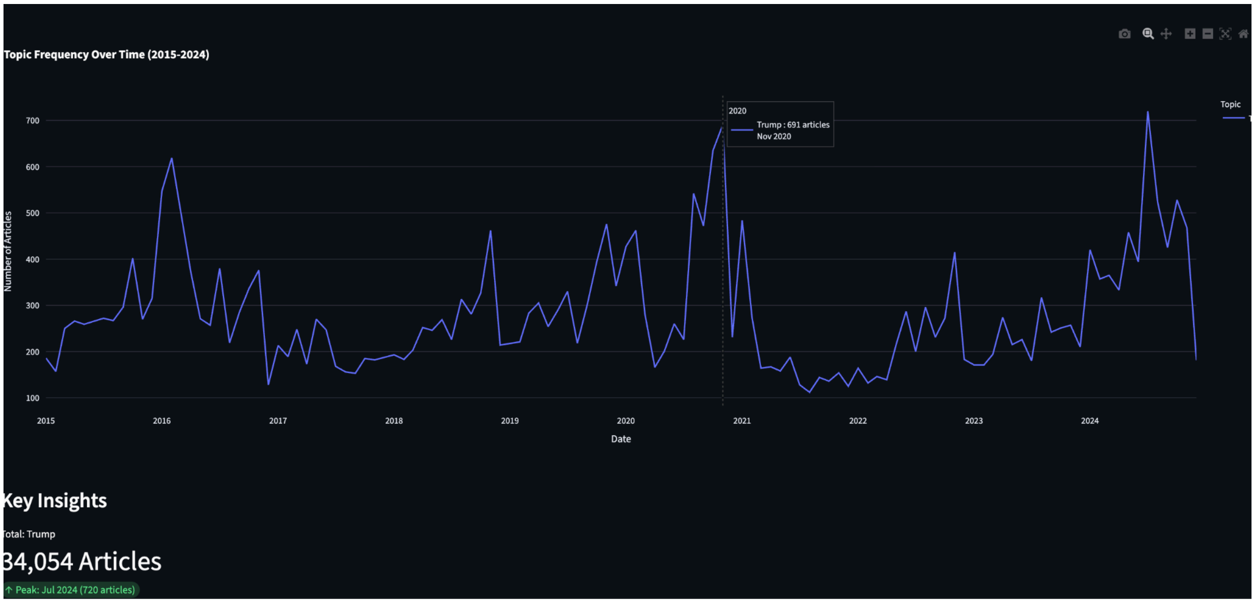
Total: China/East Asia
10,383 Articles
↑ Peak: Mar 2018 (214 articles)

Total: European Migration
686 Articles
↑ Peak: Sep 2015 (68 articles)

Total: France/Macron
948 Articles
↑ Peak: May 2017 (49 articles)

Total: Ukraine War
7,623 Articles
↑ Peak: Mar 2022 (563 articles)

In this example, we are comparing foreign news: Brexit/UK, China/East Asia, European Migration, France/Macron, and the Ukraine War. This analysis tells us which foreign issues are deemed most important across time. In the above diagram, China/East Asia was a dominant news topic until 2022 when the Ukraine War rapidly rose with the invasion of Ukraine from Russia. Other topics such as the UK or France are discussed much less frequently.



Key Insights

otal: Trump
34,054 Articles
↑ Peak: Jul 2024 (720 articles)

Additionally, we can investigate a singular topic at a time. Looking at Trump, he is one of the most frequent news topics from the past decade. We can see spikes and rises in concision with major events, such as the US Presidential elections, major policy decisions, and controversial speeches. The month with the most articles about Trump was the July 2024 "attempted assasination."