📊 Senticonomy: News Sentiment Analysis and Economic Impact Visualization

# 🧠 Project Overview

Senticonomy is a cutting-edge data analytics platform designed to collect, process, and analyze both historical and real-time news data from a variety of sources. The tool leverages natural language processing (NLP) and machine learning models to derive sentiment insights from news articles across various sectors, such as Business, Technology, Politics, and more. These insights can be used to gauge economic trends, monitor public opinion, track market sentiment, and detect potential economic shifts.

# 🎯 Objectives

The key objectives of the Senticonomy project are:  
1. Sentiment Analysis: Analyze the sentiment of both historical and real-time news data to uncover underlying economic and social trends.  
2. Market Prediction: Use sentiment insights to forecast market trends and predict the impact of news events on the economy.  
3. Public Sentiment Monitoring: Track public mood and opinion on policies, products, or specific industries to provide decision-makers with timely insights.  
4. Interactive Visualization: Provide an interactive and user-friendly dashboard for visualizing sentiment trends across different categories, making it accessible for business and public stakeholders.  
5. Automate News Ingestion: Automate the entire pipeline for daily news ingestion and sentiment analysis, ensuring that the platform is continuously updated with fresh data.

# 🔬 Scope

In Scope:  
- News Collection: Gather real-time and historical news data from various APIs (NewsAPI, GNews, Google News) to ensure comprehensive coverage of topics.  
- Sentiment Analysis: Apply NLP models like BERT, VADER, and TextBlob to classify sentiment polarity (positive, negative, neutral) in articles.  
- Clustering: Use unsupervised learning techniques such as KMeans, DBSCAN, or LDA for grouping articles based on topics or themes.  
- Visualization: Develop a dynamic, interactive dashboard for data visualization and sentiment trend analysis.  
- Automation: Implement AWS Lambda for scheduled updates, ensuring daily news ingestion and analysis with minimal manual intervention.  
  
Out of Scope:  
- Multilingual NLP: The current implementation supports only English-language news articles. Expanding this to other languages is out of scope.  
- Real-time Alerts: Although sentiment analysis can be done on-demand, real-time alerts are not part of the current system.  
- User Authentication Systems: There are no user authentication or access control systems in place for the dashboard.

# 🔀 Pipeline Overview

The project workflow can be described in the following stages:  
  
1. \*\*News Collection:\*\* Data is collected through various news APIs such as NewsAPI, GNews, and Google News.  
2. \*\*Data Storage:\*\* Raw data is initially stored in AWS S3 for archival purposes.  
3. \*\*Data Cleaning and Preprocessing:\*\* The data undergoes cleaning and preprocessing, including text normalization, tokenization, stopword removal, and lemmatization, before being structured in AWS RDS.  
4. \*\*Sentiment Analysis:\*\* Sentiment scores are calculated for each news article using a hybrid approach with models such as BERT, VADER, and TextBlob.  
5. \*\*Clustering:\*\* Similar articles are grouped using unsupervised learning techniques (e.g., KMeans, DBSCAN, LDA), helping to detect trends and emerging topics.  
6. \*\*Visualization:\*\* Results are visualized using an interactive dashboard powered by Streamlit, with visualization tools like Plotly and Matplotlib.  
7. \*\*Deployment:\*\* The entire system is deployed to AWS EC2 and S3 to enable scalability, automation, and cloud-based access.

# 💡 Key Features

- \*\*Multi-source Data Collection:\*\* Collect data from multiple news sources to ensure the diversity and breadth of topics.  
- \*\*Sentiment Classification:\*\* Analyze sentiment using advanced NLP models like BERT (for accuracy) and VADER/TextBlob (for speed).  
- \*\*Clustering for Emerging Trends:\*\* Group news articles into topics using clustering techniques to highlight significant emerging patterns.  
- \*\*Interactive Dashboards:\*\* The Streamlit-powered web interface enables users to explore sentiment trends across different categories and time frames.  
- \*\*Cloud Automation:\*\* AWS Lambda ensures that the pipeline is updated with fresh data daily, minimizing manual intervention.

# 🛠️ Technologies & Tools

The project uses a range of modern technologies for different stages of the pipeline:  
  
| Category | Technology |  
|---------------------|----------------------------------------------------|  
| \*\*Language\*\* | Python |  
| \*\*NLP Libraries\*\* | NLTK, SpaCy, BERT, TextBlob, VADER |  
| \*\*Data Collection\*\* | NewsAPI, GNews, Google News |  
| \*\*Data Storage\*\* | AWS S3, AWS RDS |  
| \*\*Data Visualization\*\* | Plotly, Matplotlib, Streamlit |  
| \*\*Automation\*\* | AWS Lambda, EC2 |  
| \*\*Version Control\*\* | Git, GitHub |

# 📈 Project Metrics

The following metrics are used to assess the performance and effectiveness of the project:  
  
| Metric | Description |  
|----------------------|-------------------------------------------------------|  
| \*\*Data Quality\*\* | Completeness, deduplication, and null checks in the dataset |  
| \*\*Sentiment Accuracy\*\*| Precision and recall for sentiment classification models |  
| \*\*Clustering Effectiveness\*\*| Silhouette score, topic coherence, and cluster interpretability |  
| \*\*Web App Usability\*\* | Load time, smoothness of interactions, and accessibility |  
| \*\*Automation Efficiency\*\* | Lambda success rate, processing time for daily updates |

# 📂 Dataset Information

- \*\*Sources:\*\* NewsAPI, GNews, Google News, Kaggle  
- \*\*Format:\*\* JSON, CSV  
- \*\*Fields:\*\*  
 - `Headline`: The title of the news article  
 - `Content`: The body of the news article  
 - `Date`: The publication date of the article  
 - `Source`: The originating news outlet  
 - `Category`: The category of news (e.g., Business, Technology, Politics)  
 - `Sentiment Score`: The sentiment analysis score (positive, negative, neutral)  
- \*\*Processing:\*\* Data is preprocessed using techniques like tokenization, text normalization, and stopword removal. Sentiment scores are then calculated using BERT, VADER, and TextBlob.

# 📦 Deliverables

- \*\*Modular Python Scripts:\*\* Scripts for news collection, preprocessing, analysis, and visualization  
- \*\*AWS Infrastructure:\*\* Setup for S3, RDS, Lambda, EC2  
- \*\*Processed Dataset:\*\* The dataset cleaned and structured in AWS RDS  
- \*\*Streamlit Web Application:\*\* A user-friendly interface to explore sentiment analysis results  
- \*\*Documentation:\*\* Detailed README and API usage guides

# 📜 Guidelines for Contributors

- \*\*Modular Code:\*\* Ensure that the code is well-organized into reusable functions or classes for maintainability.  
- \*\*Version Control:\*\* Commit code regularly to Git with descriptive commit messages. Follow best practices for branching and pull requests.  
- \*\*Security:\*\* Use `.env` files to store sensitive information like API keys securely. Never hard-code keys into scripts.  
- \*\*Logging and Error Handling:\*\* Implement proper logging for debugging and exception handling to catch errors during execution.  
- \*\*Efficiency:\*\* Optimize for cost and performance when using AWS services. Monitor and adjust usage based on project needs.

# 🔖 Technical Tags

#NLP #SentimentAnalysis #AWS #Python #APIs #Clustering #NewsAnalytics #Streamlit #DataVisualization

# 🤝 Business Use Cases

The Senticonomy platform can be used across various business and public sectors to:  
  
- \*\*Financial Market Prediction:\*\* Track market sentiment based on news coverage and sentiment fluctuations to make better investment decisions.  
- \*\*Public Sentiment Tracking:\*\* Monitor how the public reacts to political, social, and economic policies, enabling organizations to adjust strategies accordingly.  
- \*\*Brand & Media Monitoring:\*\* Monitor sentiment towards specific brands or products, tracking shifts in public perception.  
- \*\*Policy Impact Assessment:\*\* Evaluate the effectiveness of policies by analyzing the sentiment of news coverage over time.

# 📜 Final Notes

“Sentiment is the soul of society. Senticonomy gives it a voice, a shape, and a dashboard.”  
  
\*\*Author:\*\* Santhosh Kumar  
\*\*Contact:\*\* [Your GitHub Profile or Email]  
\*\*License:\*\* MIT / Apache 2.0 (Choose accordingly)