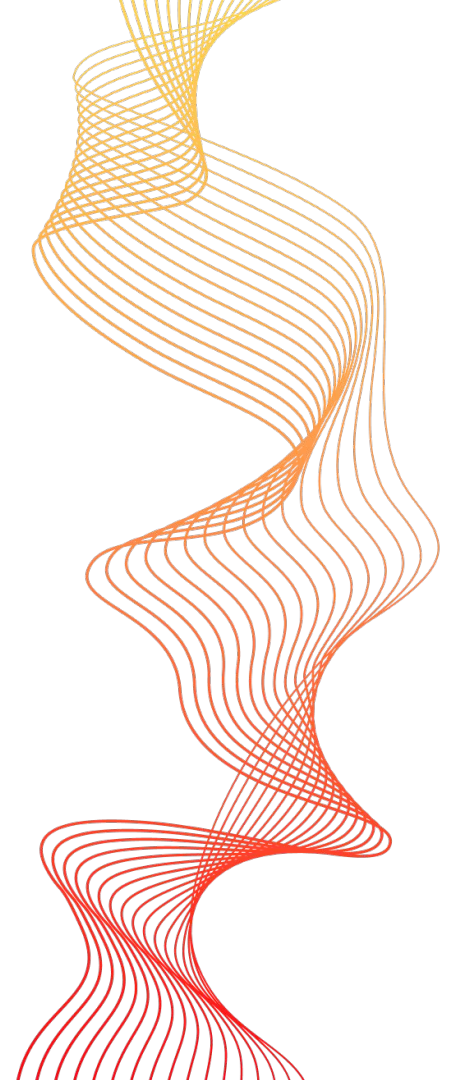




NewsScope : A News Search and Sentiment Analysis App

This presentation introduces NewsScope, a news search and sentiment analysis app built with Python, AI21 Labs API, Gnews API and Streamlit.

Deep Dreamers





Introduction

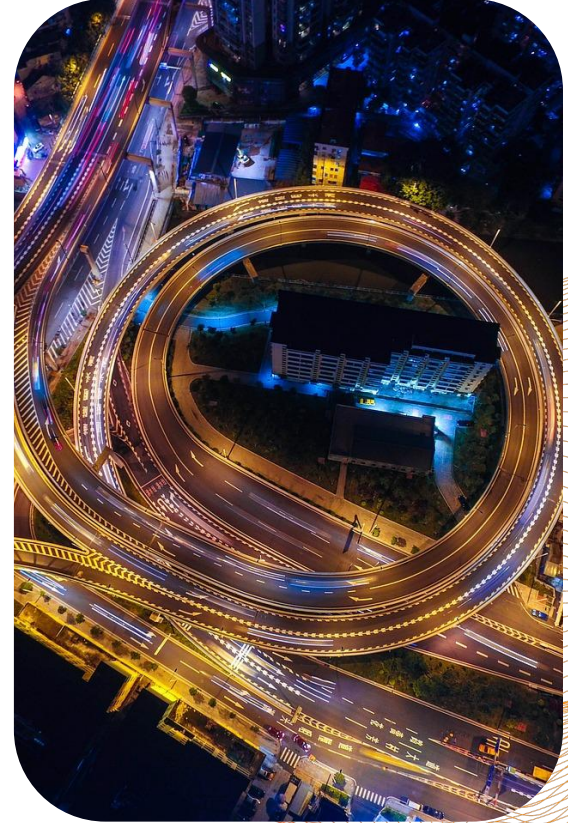
- NewsScope is a news search, summarization, and sentiment analysis app built with Python and Streamlit.
- The app uses the GNews API to fetch news articles and applies TextBlob for sentiment analysis.
- AI21's Summarization API is also employed to generate a brief summary of the article.





Features

- NewsScope has two pages: one for searching news articles and one for displaying top headlines.
- The user can specify the language, country, and maximum number of results for the search.
- The app displays the title, summary, sentiment, and URL for each article found.





Functionality

- The app fetches articles based on user input and processes them using sentiment analysis and summarization.
- The sentiment analysis component uses TextBlob to assess the overall sentiment of the article.
- The summarization component employs AI21's Summarization API to generate a brief summary of the article.





Technical Requirements

- The project requires installation of several Python libraries including Streamlit, TextBlob, and Requests.
- Installation and setup instructions are available on the project's GitHub page.

```
TYPE,  
TYPE,  
MODE_TYPE,  
_TYPE,  
ReactSymbols';  
Element from 'shared/isValidElementType';  
function typeOf(object: any) {  
  if (object === 'object' && object != null) {  
    $$typeof = object.$$typeof;  
    switch ($$typeof) {  
      case REACT_ELEMENT_TYPE:  
        const type = object.type;  
        switch (type) {  
          case REACT_FRAGMENT_TYPE:  
          case REACT_PROFILER_TYPE:  
          case REACT_STRICT_MODE_TYPE:  
          case REACT_SUSPENSE_TYPE:  
            return type;  
          default:  
            const $$typeofType = type && type.$$typeof;  
            switch ($$typeofType) {  
              case REACT_CONTEXT_TYPE:  
              case REACT_FORWARD_REF_TYPE:  
              case REACT_LAZY_TYPE:
```



Thank you for your time 😊