AI Usage Report: ESG News Analyzer

Project Overview

Project Name: ESG News Analyzer & Sentiment Tracker

Executive Summary

1. AI Tools Used in Development

1.1 GitHub Copilot (Development Assistant)

Purpose: Code generation, development assistance

Usage Areas:

- Code Generation: Assisted in writing boilerplate code for React components, Express routes, and MongoDB schemas
- Debugging: Helped identify and fix bugs in API endpoints, sentiment analysis flow, and data visualization
- Code Refactoring: Suggested improvements for code organization and best practices

1.2 Claude (Architecture)

Purpose: Technical consultation and architectural decisions

Usage Areas:

- Discussed database schema design for NewsArticle model
- Researched sentiment analysis models and ESG categorization approaches
- Explored charting library options (chose Chart.js and Recharts)

2. AI Integration in the Application

2.1 Google Gemini AI (Core Feature)

Purpose: Intelligent ESG categorization and comprehensive summary generation

Implementation Details:

- API: Google Generative AI (@google/generative-ai v0.21.0)
- Model: gemini-2.0-flash-exp / gemini-2.5-flash-lite

- Integration: Backend services (services/esgCategorizationService.js, services/summaryService.js)

Features Powered by Gemini:

A. ESG Categorization System

Input: Article title, description, and content

Processing: Gemini analyzes text against ESG framework (Environmental, Social, Governance)

Output: Array of applicable categories, primary category, and explanation

Example: "Tesla announces solar expansion" → Environmental (primary), with detailed reasoning

B. AI-Powered ESG Summary Generation

Functionality: Comprehensive analysis of company's ESG standing based on news articles

Context: Analyzes up to 20 recent articles with sentiment and category data

Output: Structured summary including:

- Overall ESG standing (2-3 sentences)
- Key concerns and positive highlights
- Trending topics with importance levels
- Stakeholder recommendations

Use Cases:

- Quick executive overview of company ESG performance
- Identifying critical ESG risks and opportunities
- Tracking sentiment trends across ESG dimensions

2.2 Hugging Face Transformers.js (Core Feature)

Purpose: Real-time sentiment analysis of news articles

Implementation Details:

- Library: @huggingface/transformers (v3.8.1)
- Model: Xenova/distilbert-base-uncased-finetuned-sst-2-english (DistilBERT)

- Integration: Backend service (services/sentimentService.js)
- Runs Locally: No external API calls required

Features Powered by Transformers:

A. Sentiment Classification System

Input: Article text (title + description)

Processing: BERT-based model analyzes sentiment with confidence scoring

Output: Sentiment label (Positive/Negative/Neutral)

Logic:

- Confidence ≥ 0.6 → Model's prediction
- Confidence ≤ 0.4 → Inverse prediction
- Confidence 0.4-0.6 → Neutral classification

Performance: Singleton pattern for efficient model reuse across requests