Report
Sentiment Analysis of Financial News

Name: Devershika Mohane
PRN: 23070126032
AIML A

Objective

The main objective of this project is to develop a machine learning-based system that analyzes the sentiment of financial news articles and classifies them as *positive*, *negative*, or *neutral*.

To achieve this, the project involves:

- Performing comparative analysis of multiple machine learning algorithms to identify the most effective model for text sentiment classification.
- Applying natural language processing (NLP) techniques for text cleaning, tokenization, and feature extraction using TF-IDF.
- Deploying the final trained model on AWS EC2, enabling real-time sentiment predictions through a web-based interface that can handle multiple user queries efficiently.

The ultimate goal is to bridge the gap between qualitative financial news and quantitative market analysis by automating the sentiment classification process using scalable cloud-based AI infrastructure.

Overview

The financial world thrives on information — stock prices, trading behavior, and investor decisions are often driven by news events and market sentiment. However, analyzing massive volumes of financial news manually is not only slow but also subject to human interpretation and bias. This project leverages Natural Language Processing (NLP) and Machine Learning (ML) to automate this process. Through NLP, textual data from news articles is cleaned, tokenized, and vectorized to make it machine-readable. Several ML models are then trained to learn sentiment patterns in financial language. After comparing different algorithms, it was found that Linear SVM (Support Vector Machine) provided the best performance in terms of accuracy and generalization. To ensure practical usability, the model was deployed using Amazon EC2, providing an accessible, cost-effective, and scalable solution that can analyze sentiment in real time for any given financial news input. This project demonstrates how machine learning and cloud technologies can work together to provide intelligent decision support tools for investors, analysts, and financial researchers.

Problem Statement

Financial news headlines and articles contain valuable insights into market movements, but analyzing them manually is inefficient and prone to bias.

The problem is to **design and implement an automated sentiment analysis system** capable of accurately classifying financial text data, thereby assisting analysts and investors in making informed decisions based on market sentiment trends.

Architecture

The architecture of the system includes the following major components:

1. Data Collection:
   - Financial news articles and headlines were collected from open repositories and reliable datasets such as Kaggle and Yahoo Finance.
   - Each data point consisted of the text content and an associated sentiment label.

2. Data Preprocessing:
   - All text was cleaned by removing punctuation, digits, and special symbols.
   - Tokenization and lemmatization were applied using NLTK to normalize words.
   - Stopwords were filtered out to retain only the most meaningful words.

3. Feature Extraction:
   - Text data was converted into numerical vectors using TF-IDF (Term Frequency–Inverse Document Frequency).
   - This technique helped highlight important words that carry greater significance in financial contexts.

4. Model Training:
   - Multiple algorithms — Logistic Regression, Naive Bayes, Random Forest, and Support Vector Machine (SVM) — were trained and tested.
   - Each model was evaluated using accuracy, precision, recall, and F1-score metrics.

5. Model Evaluation and Selection:
   - After detailed comparative analysis, Linear SVM was chosen as the final model because of its superior classification performance and robustness to overfitting.

6. Deployment:
   - The trained SVM model was serialized using Pickle and deployed on AWS EC2.
   - A Flask-based web application was hosted, enabling users to input any financial news headline and receive an instant sentiment prediction.

AWS Service Used – EC2

Amazon Elastic Compute Cloud (EC2) was utilized as the primary AWS service for deploying the project.

- EC2 instances hosted the trained Linear SVM model and the web application.
- Flask was used as the backend framework to handle requests and deliver real-time sentiment outputs.
- The EC2 environment provided flexibility for scaling resources according to incoming traffic and ensured smooth performance for the web interface.
- Security groups were configured to restrict unauthorized access while allowing users to interact with the hosted application via HTTP requests.

Implementation

The complete implementation process involved the following stages:

1. Data Preprocessing:
    - Libraries such as *pandas*, *NLTK*, and *scikit-learn* were used for text cleaning, tokenization, and lemmatization.
    - Text was transformed into TF-IDF vectors to capture word importance.
2. Model Training and Evaluation:
    - Multiple models were trained and evaluated.
    - Linear SVM achieved the best accuracy and F1-score.
    - Hyperparameter tuning was done using grid search for optimal regularization parameters.
3. Model Serialization:
    - The finalized SVM model was serialized using *pickle.dump()* for deployment.
4. Deployment on AWS EC2:
    - The Flask web application was hosted on EC2 with the serialized model integrated into the backend.
    - The frontend accepted user inputs (financial news text) and returned the sentiment result (Positive, Negative, Neutral).
5. Testing:
    - The system was tested on unseen data to validate generalization and latency.
    - The average response time for real-time inference was within acceptable limits (<2 seconds).

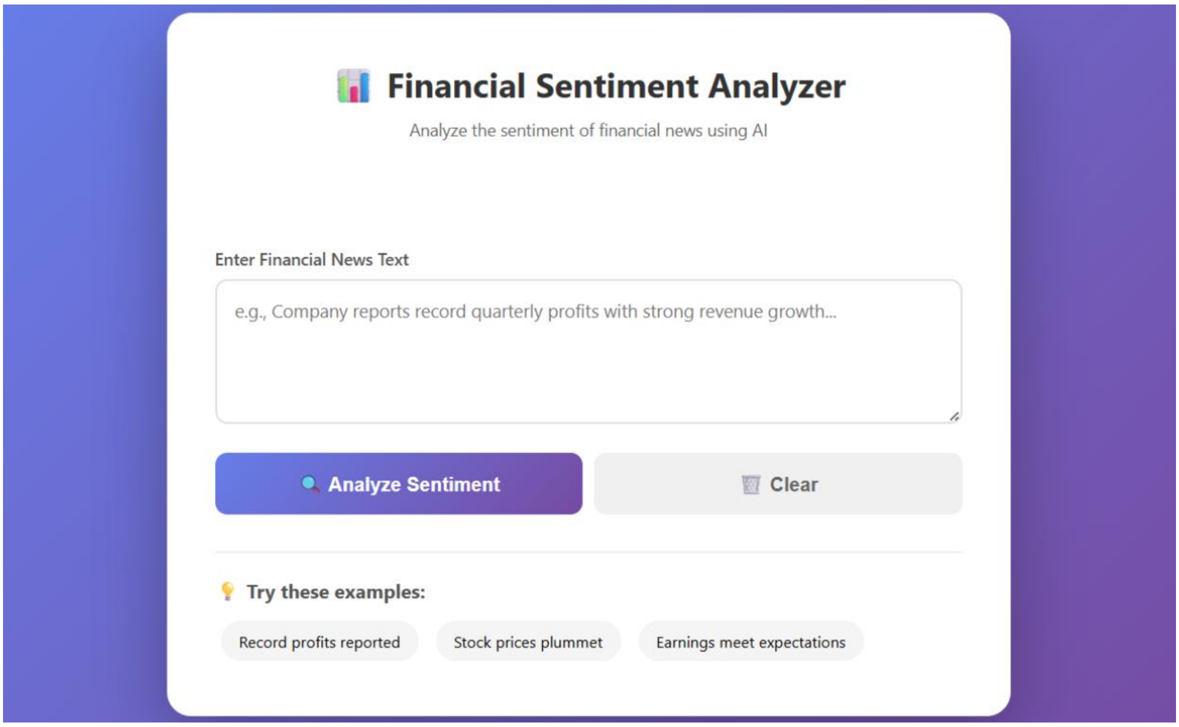
Challenges

- Data Imbalance: The dataset had more neutral samples, causing skewed model learning.
- Domain-Specific Vocabulary: Financial jargon required custom text preprocessing to improve accuracy.
- Deployment Optimization: Ensuring that the EC2 instance efficiently handled concurrent requests while maintaining low latency.
- Hyperparameter Tuning: Achieving optimal regularization parameters for SVM to avoid overfitting.

Conclusion

The project successfully developed and deployed a machine learning-based financial sentiment analysis system using AWS EC2. Through rigorous experimentation and comparative model analysis, Linear SVM was identified as the best-performing algorithm, providing high accuracy and interpretability.

Deployment on EC2 enabled real-time, scalable, and secure sentiment predictions, making the system practical for real-world financial applications. This work highlights the synergy between AI and cloud computing, showcasing how automation can enhance financial analysis and decision-making. Future improvements may include integrating deep learning models such as BERT or FinBERT, real-time data streams, and multi-language support for broader financial news coverage.



📊 Financial Sentiment Analyzer

Analyze the sentiment of financial news using AI

Enter Financial News Text

e.g., Company reports record quarterly profits with strong revenue growth...