Social Media Sentiment Analysis for Product Reviews

Project Duration

2 Months (8 Weeks)

Team Composition and Roles

- **Team Member 1:**
  Project Manager & Data Engineer
  • Responsible for planning, task delegation, and ensuring data pipelines are in place for collecting review data from various social media platforms (Twitter, Reddit, YouTube, etc.).

- **Team Member 2:**
  NLP Specialist & Model Developer
  • Develops sentiment classification models using NLP and fine-tuned transformer models (e.g., BERT). Handles model training, tuning, and evaluation.

- **Team Member 3:**
  Data Analyst & Visualization Lead
  • Conducts EDA, word frequency and sentiment trend analysis. Creates sentiment distribution charts and dashboards.

- **Team Member 4:**
  Frontend & Reporting Developer
  • Builds interactive dashboards using Streamlit or Power BI and prepares documentation and final presentation.

Objective

To analyze customer sentiments from social media product reviews using Natural Language Processing (NLP) and Machine Learning techniques. The system helps brands and companies understand user opinions, identify product strengths and weaknesses, and adjust their marketing or R&D strategies accordingly.

Functional Requirements

- **Data Collection and Preprocessing**
  - Scrape or collect review data from Twitter, Reddit, YouTube comments, or product forums.
  - Clean text data: remove emojis, links, hashtags, special characters.

- **Sentiment Labeling and Annotation**
  - Use rule-based or manual techniques to label a training dataset as Positive, Neutral, or Negative.
  - Optionally integrate external datasets like IMDB or Amazon Reviews for transfer learning.

- **Model Development**
  - • Train sentiment classification models using BERT, Logistic Regression, or LSTM.
  - • Compare model performance and select the best one based on F1-score and accuracy.

- **Real-Time Sentiment Analysis**
  - • Support on-the-fly analysis of new review inputs via web interface.
  - • Show prediction result with confidence score.

- **Keyword and Emotion Extraction**
  - • Use topic modeling (LDA) or word frequency to extract key product-related themes.
  - • Optionally detect emotions like anger, joy, trust using emotion lexicons.

- **Dashboard and Visualization**
  - • Show pie/bar charts for sentiment distribution, line charts for time-based trend analysis.
  - • Word cloud for most frequent positive/negative terms.

- **Reports and Exporting**
  - • Export analysis results in PDF/Excel formats.
  - • Provide summarized insights (e.g., "60% of reviews for Product A are positive in Q1 2025").

- **Access and User Roles**
  - • Allow access for marketing team, product managers, and analysts.
  - • Implement optional authentication for internal tools.

Non-Functional Requirements

- **Performance**
  - • System should classify a batch of 1000 reviews within 5 seconds.
  - • Support concurrent requests for real-time prediction.

- **Scalability**
  - • Scalable backend to support multiple product review pipelines.
  - • Ability to scale sentiment analysis to multiple product categories or regions.

- **Usability**
  - • Dashboard must be clean, interactive, and usable without technical expertise.
  - • Include tooltips and explanations for visual components.

- **Reliability**
  - • Results must be consistent across sessions and users.
  - • Version control for model versions.

- **Maintainability**
  - • Modular and clean codebase.
  - • Documented API and model interfaces for upgrades.

- **Portability**
  - • Solution should run on both local and cloud environments.
  - • Compatible with Windows, Linux, and macOS.

- **Security**
  - Secure API and data pipelines.
  - Anonymize user data if collecting from social accounts.

- **Availability**
  - Uptime of 99% for hosted dashboards.
  - Auto-restart mechanisms for background sentiment services.

- **Documentation**
  - Include user manual, technical implementation guide, and visual explanation of outputs.
  - Support documentation for training new team members.

Technical Stack

Frontend:
- Streamlit or Power BI for interactive UI
- Plotly / Seaborn for visualizations
- WordCloud for keyword visuals

Backend:
- Python (Flask / FastAPI for API setup)
- NLTK / SpaCy / Hugging Face Transformers (for NLP)
- Scikit-learn / TensorFlow / PyTorch (for model building)

Data Collection:
- Tweepy (Twitter API), PRAW (Reddit API), YouTube Data API
- Pandas / NumPy for preprocessing
- Regex, emoji, stopwords libraries

Database (Optional):
- SQLite / PostgreSQL for storing cleaned data
- MongoDB for unstructured or JSON data

Deployment:
- Streamlit Cloud / Heroku / AWS
- GitHub for collaboration and version control

Implementation Steps

1. **Project Planning and Requirements**
   - Define scope, objectives, target products/platforms
   - Finalize data sources and assign team roles

2. **Data Collection and Cleaning**
   - Extract review data using APIs
   - Preprocess: remove noise, normalize text

3. **Labeling and Model Training**
   • Prepare or annotate dataset
   • Train baseline models and BERT-based classifier
   • Evaluate accuracy, precision, recall

4. **Exploratory Data Analysis (EDA)**
   • Visualize sentiment distribution
   • Extract keywords and phrase frequencies
   • Use LDA or TextRank for topic extraction

5. **Dashboard Development**
   • Build UI with filters (date, product, platform)
   • Add charts, sentiment summaries, and review snippets

6. **API Integration (Optional)**
   • Create RESTful APIs to classify live inputs
   • Enable third-party or internal tool integration

7. **Testing and Validation**
   • Validate with sample reviews
   • Cross-check visualizations and predictions

8. **Insight Generation and Reporting**
   • Highlight product perception and emotional sentiment
   • Generate final reports and business recommendations

9. **Deployment and Handover**
   • Deploy dashboard and models
   • Deliver documentation and code repository