Tweet Sentiment Analysis Using NLP: A CRISP-DM Approach

Business Understanding

Overview

In today's digital-first world, platforms like Twitter have become powerful spaces where individuals express their emotions, opinions, and feedback in real time. For businesses, governments, and organizations, understanding this unstructured, fast-moving data is critical to tracking public sentiment, spotting trends, managing crises, and making informed decisions. Manually analyzing thousands of tweets is not feasible, hence the need for an automated solution.

Challenges

- Tweets are short, informal, and often filled with slang, abbreviations, emojis, hashtags, and noise (e.g., @mentions, URLs).
- Public sentiment is dynamic and varies greatly across topics, brands, and user communities.
- The sentiment labels in available data are often imbalanced, with more neutral or positive expressions than negative ones.
- Businesses need to make sense of this high-volume, unstructured data quickly and accurately to act on it effectively.

Proposed Solution

This project proposes the development of a Natural Language Processing (NLP) and machine learning pipeline to classify the sentiment of tweets into one of four categories: **Positive**, **Negative**, **No Emotion**, or **I Can't Tell**. The solution involves preprocessing the tweet text, transforming it into numerical vectors using techniques like TF-IDF, and training a robust classifier (e.g., Random Forest) to detect sentiment patterns. This automated system can be used by stakeholders to track brand perception, respond to customer feedback, and monitor public discourse in real time.

Business Impact

Social media sentiment plays a critical role in shaping brand image, political opinion, and customer loyalty. Automated sentiment analysis allows organizations to:

- Monitor public opinion at scale.
- Respond quickly to crises or negative feedback.
- Understand how customers feel about products or services.
- Inform marketing, customer service, and product decisions.
- Save time and improve decision-making
- Monitor brand health
- Detect early sign of crises

Problem Statement

Organizations are overwhelmed by the volume and complexity of tweet data, making it difficult to understand public sentiment accurately and in real time. Without an automated solution, valuable insights remain hidden in unstructured social media content.

Objectives

i. To collect and explore a dataset of real-world tweets with labeled sentiment.
ii. To clean and preprocess the tweet text using NLP techniques.
iii. To engineer features using TF-IDF vectorization.
iv. To build and evaluate machine learning models that classify tweet sentiment.
v. To select the best-performing model based on accuracy and weighted F1-score.
vi. To provide recommendations for applying the model in real-world social listening applications.

Data Understanding

Goal:
To gather and explore the tweet dataset to understand its structure, assess data quality, and uncover initial patterns that will guide preprocessing and modeling.

Dataset Description:

- `tweet_text`: Raw content of the tweet (main input for NLP).
- `emotion_in_tweet_is_directed_at`: Brand or item referenced in the tweet (e.g., "iPhone").
- `is_there_an_emotion_directed_at_a_brand_or_product`: Target variable — sentiment expressed in the tweet. Categories include:
    - Positive emotion
    - Negative emotion
    - No emotion toward brand or product
    - I can’t tell

Initial Checks Performed:

- Checked for null values and missing data in each column.
- Verified uniformity and consistent formatting in the sentiment and product columns.
- Checked for duplicate tweets.
- Explored class imbalance in the sentiment distribution.
- Performed basic visualizations: bar charts and pie charts to show sentiment proportions.
- Analyzed frequently used words and sentiment-specific terms.

Data Preparation

Goal:
To clean and transform the raw tweet data into a structured format suitable for machine learning models, while preserving information relevant to sentiment.

Data Cleaning & Transformation Steps:

- Removed noise: URLs, @mentions, hashtags, punctuation.
- Standardized text: Converted all tweet text to lowercase.
- Tokenization: Split tweets into words.
- Stopword removal: Removed common, non-informative words.

- Stemming: Reduced words to their root form.
- Handled duplicates: Removed repeated tweets.

Feature Engineering:

- Created a new column processed_tweet with cleaned and transformed text.
- Vectorized using TF-IDF to extract numerical features from text.

Data Quality Checks:

- Ensured no nulls in the final features.
- Plotted sentiment label distribution to confirm class imbalance.

Modeling

Goal:
To select, train, and optimize machine learning models that can accurately classify tweet sentiment.

Models Used and Justification:

- Logistic Regression: Simple, interpretable baseline for text classification.
- Random Forest: Robust with non-linear relationships and good for imbalanced data.
- Naive Bayes: Fast, commonly used for text.
- KNN: Simple but slower and less scalable.
- XGBoost: High-performing gradient boosting model.

Model Pipeline:

- TF-IDF Vectorizer → Classifier

Optimization:

- Used GridSearchCV to fine-tune hyperparameters (e.g., C, max_depth, n_estimators, learning_rate).

Evaluation

Goal:
To test the model performance and validate its effectiveness using suitable evaluation metrics.

Metrics Used:

- Accuracy: Overall correctness.
- Precision, Recall, F1-score: Per-class and macro-averaged.
- Weighted F1-score: Chosen as the main metric due to class imbalance.

Statistical Validation:

- Used classification report and confusion matrix.
- Visuals: Heatmaps, ROC curves, learning curves.
- The best model (Random Forest) achieved ~68% accuracy and ~66% weighted F1-score.

Deployment

Goal:
To make the sentiment classification model accessible and usable in real-world applications.

Deployment Plan

- The model is intended to be deployed
- Could be integrated into dashboards for live sentiment tracking
- Further improvements could allow integration into customer feedback systems

Steps for Deployment:

- Saved the model using joblib or pickle.
- Integrated with a Streamlit UI to allow users to input tweets and receive sentiment predictions in real time.

Reflection and Future Work

What Went Well

- TF-IDF performed well for sparse text
- Random forest handled class imbalance well
- Pipeline was modular, allowing easy swapping of models or vectorizers

Challenges

- Class imbalance slightly affected minority class performance
- Short tweets made some sentiment hard to infer, especially for “No Emotion” and “I Can’t Tell”

Future Improvements

- Incorporate pretrained embeddings (e.g., Word2Vec, BERT)
- Try deep learning approaches (e.g., LSTM, transformers)
- Build a real-time sentiment dashboard
- Expand dataset to multilingual tweets or specific industries

Conclusion

This project demonstrates how Natural Language Processing (NLP) and machine learning can convert unstructured tweet data into meaningful sentiment insights.
The best-performing model (Random Forest + TF-IDF) can support real-world applications in brand monitoring, public opinion tracking, and automated feedback analysis.