Binary Classification - Sentiment Analysis Tweets

Ángela Franco
March, 2025

Project Overview

- The objective of this project is to develop a classification model to classify Tweets as positive or negative.

- Three ML models will be trained and evaluated: Logistic Regression, Random Forest and LightGBM.

- Sentiment analysis is an important part of NLP.

Business Problem

- Understand trends and opinions on various topics and classify them as positive or negative.

- The key requirements for the model are:
  - Good performance on both classes (positive and negative)
  - Efficiency in handling large datasets
  - Quick inference times

Dataset and Exploratory Data Analysis (EDA)

-Dataset**:** Sentiment140 from Kaggle, 1.6M tweets; subset of 50K tweets used, balanced dataset (25K each class).

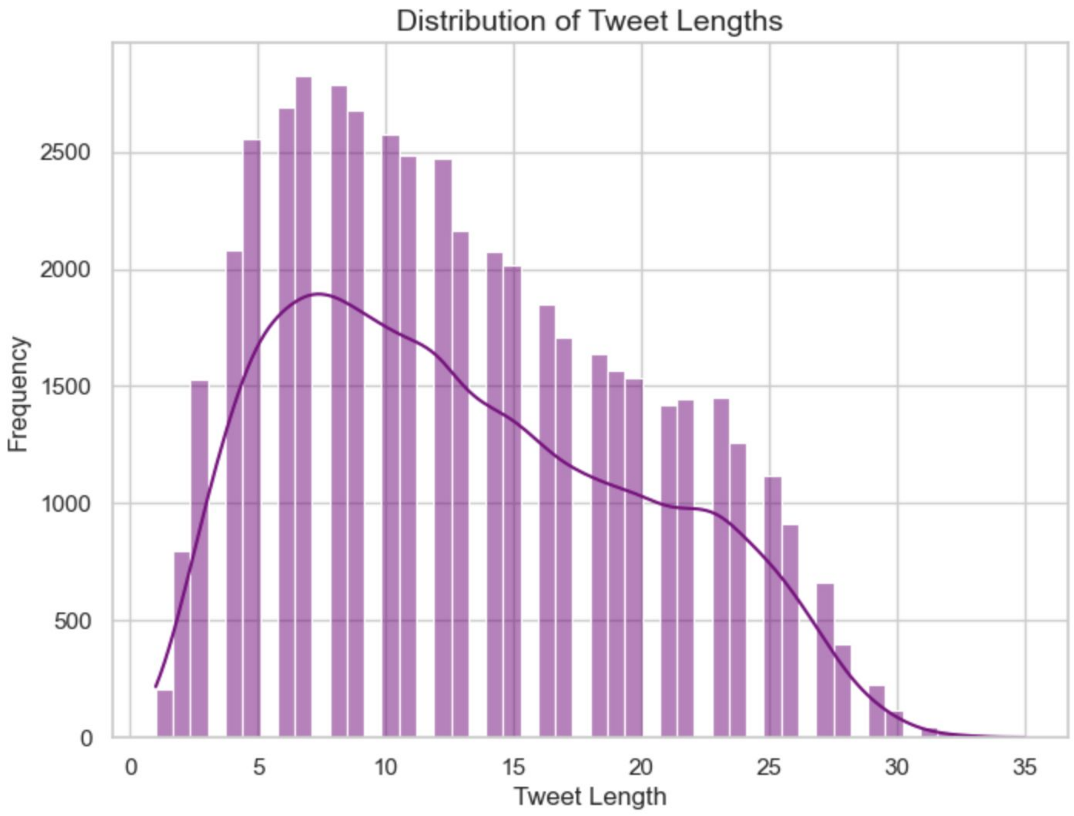
-The Tweets were sourced in 2009.

-The dataset was processed to delete duplicates, it did not include any missing value.

-The labels were normalised (0 = negative; 1 = positive.

Exploratory Data Analysis (EDA)

Tweet Length: the Tweets' length follow a normal distribution, they are adjusted to the character limitations set in Twitter.

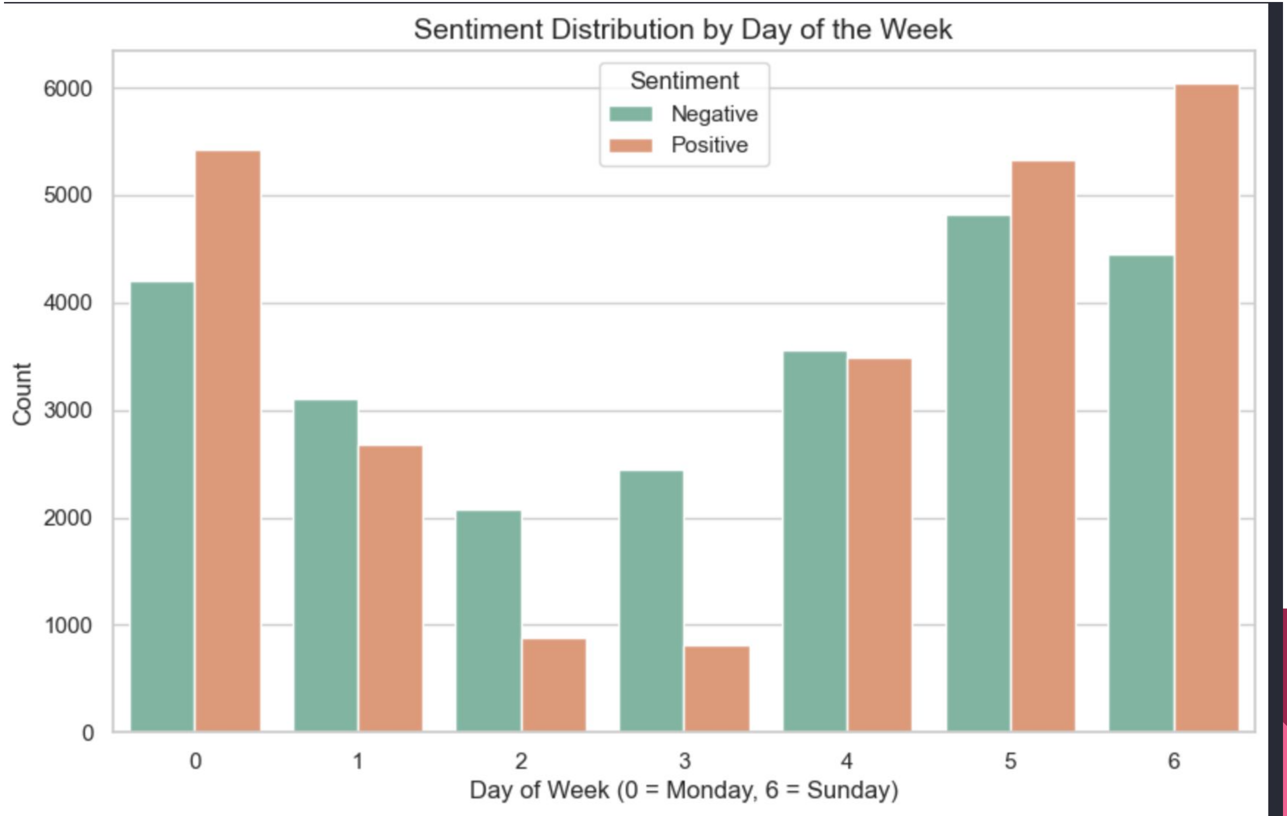


Exploratory Data Analysis (EDA)

Sentiment distribution by day of the week: most of the Tweets were posted at the beginning and end of the week.

Positive Tweets tend to appear on Monday and weekends.

Negative Tweets tend to appear during weekdays.

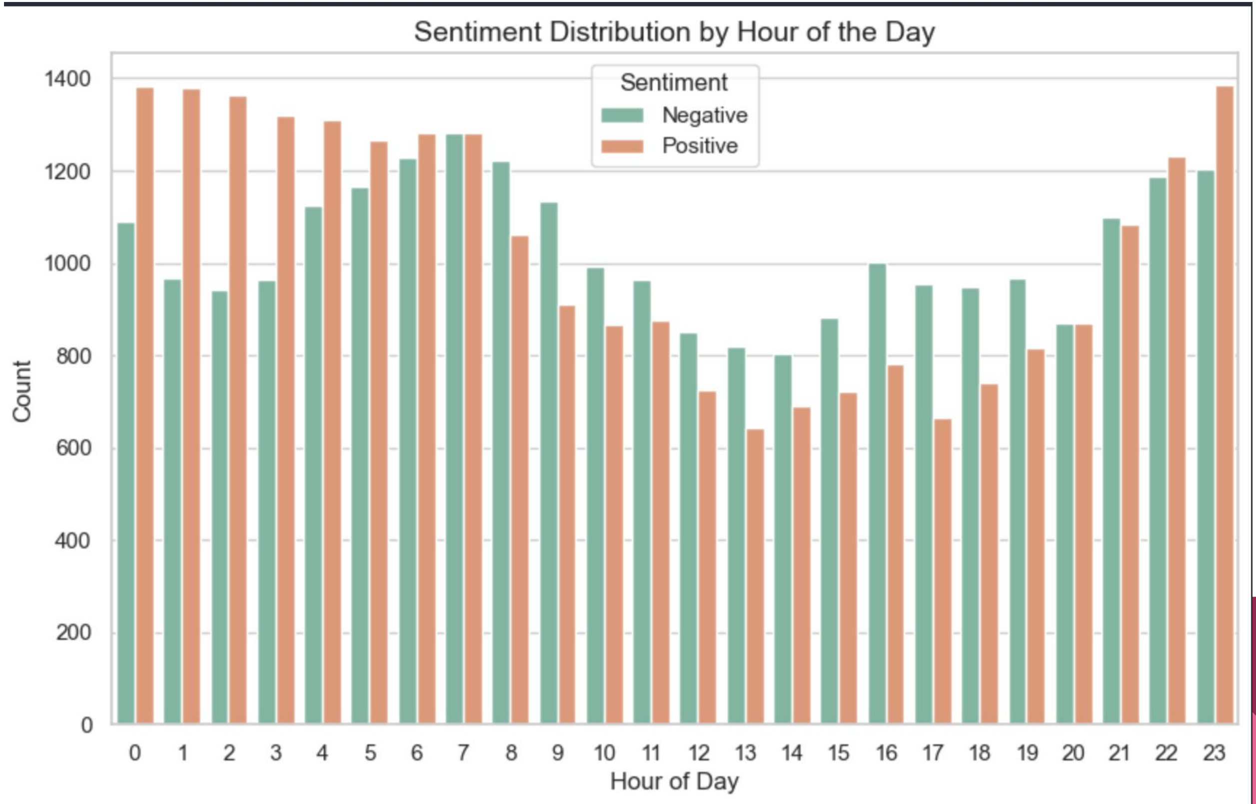


Exploratory Data Analysis (EDA)

Sentiment distribution by hour: most of the Tweets were posted outside typical working/ school hours.

Positive ones were posted mainly during the night, from 0 to 7 and between 22 and 23

Negative ones are gathered during working hours, between 8 and late afternoon, 21.



Feature Engineering

1. Text Preprocessing

-Lowercasing: Standardized all text by converting it to lowercase.

-Punctuation & Special Character Removal: Removed unnecessary symbols, hashtags, and special characters that do not contribute to sentiment analysis.

-Stopword Removal: Eliminated common words (e.g., "the", "is", "and") that do not add significant meaning.

-Lemmatization: Reduced words to their base forms to maintain semantic meaning (e.g., "running" → "run").

-Handling URLs & Mentions: Replaced or removed Twitter handles (@user) and links to focus on the core text.

Feature Engineering

2. Feature Extraction & Vectorization

-TF-IDF (Term Frequency-Inverse Document Frequency): Converted text into numerical representations while weighing important words higher.

-TF-IDF Vectorization: Used for better capturing word importance and improving the representation of the text for model training.

Model Training

- Train/ Test split (80-20)
- Function to train and evaluate the models
- The 3 models were trained and evaluated, Logistic Regression had the better base performance with 0,75 accuracy.

Model Training

- **Logistic Regression (LR):** highest accuracy (**0.7543**); similar precision, recall, and F1-score for both classes; longest training time (**146.35 seconds**)
- **Random Forest (RF):** lowest accuracy (**0.6976**); precision for class 0 (**0.77**) is higher, but recall is significantly lower (**0.58**); recall for class 1 is the highest (**0.82**); training time (**3.74 seconds**)
- **LightGBM:** accuracy (**0.7271**); good balance between precision and recall; tends to misclassify more samples compared to Logistic Regression; fastest training time (**1.91 seconds**)

Hyperparameter Tuning

- The 3 models were tuned used HalvingRandomSearchCV to optimise the computational costs and times.

- During the hyperparameter tuning I modified several hyperparameters to try different combinations.

- Best two models: Logistic Regression and LightGBM.

Hyperparameter Tuning

- **Tuned Logistic Regression:**accuracy: **76.27%** (**+0.84%**); training Time: **478.74s** (↑ **longer**); improved recall for both classes; fewer misclassifications in class 0.
- **Tuned Random Forest: a**ccuracy: **71.25%** (**+1.49%**); training Time: **9.71s** (↑ **but still fast**); improved precision for class 0, better recall for class 1; still struggles with false positives.
- **Tuned LightGBM (Best Performing Model):** accuracy: **75.37%** (**+2.66%**); inference Time: **0.42s** (**fastest**); small but consistent improvements in precision/recall; fewer misclassifications across both classes.

Results and Model selection

- Given the key requirements for our business problem: best Choices: **Tuned LightGBM** → Fast & high accuracy (**0.7537**) and T**uned Logistic Regression** → Good accuracy but longer training time

- Less Favorable: T**uned Random Forest** → Less improvement, slower

Conclusions and Final Considerations

- Social media posts contain **subjectivity & nuance** (e.g., irony, metaphors)

- Traditional ML models **struggle** with these complexities

- **Future Research:** Explore **Large Language Models (LLMs)** for better NLP performance