# Tweet Sentiment Classifier

## 1. Objective

Classify tweets as positive or negative using classical NLP methods. The pipeline uses TF-IDF vectorization (unigrams + bigrams) and a Logistic Regression classifier. Evaluation uses 3-fold cross-validation and a 15% hold-out test set.

## 2. Dataset and Preprocessing

**Dataset:** Sentiment140.  
**Size:** Approximately 1.6 million tweets. A subset was used to speed processing.  
Labels: 0 = negative, 4 = positive. Neutral tweets were not used in this binary setup.

Preprocessing steps applied:

* Lowercasing
* Removal of URLs and www links
* Removal of user mentions (@username)
* Removal of hashtag symbols, punctuation and non-alphanumeric characters
* Normalization of whitespace

## 3. Methodology

Feature extraction and modeling details:

|  |  |
| --- | --- |
| Component | Configuration |
| Vectorizer | TF-IDF with ngram\_range=(1,2), max\_features=10000 |
| Classifier | LogisticRegression, solver=lbfgs, max\_iter=1000, C=1.0 |
| Hyperparameter Search | GridSearchCV over C and tfidf params, cv=3 |
| Train/Test Split | 15% hold-out test, stratified by label |

## 4. Results

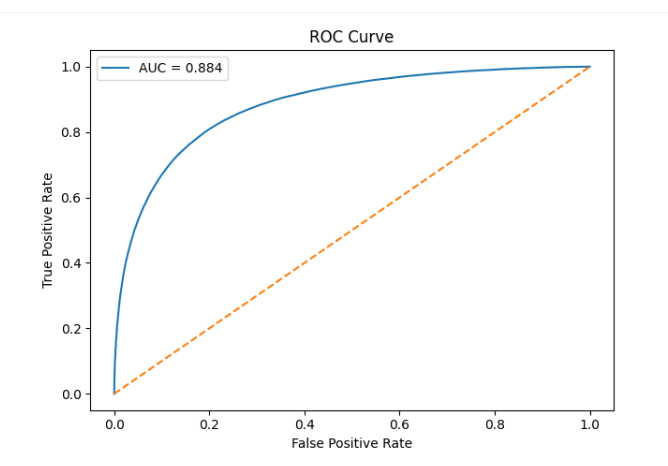
Evaluation metrics computed on the hold-out test set:

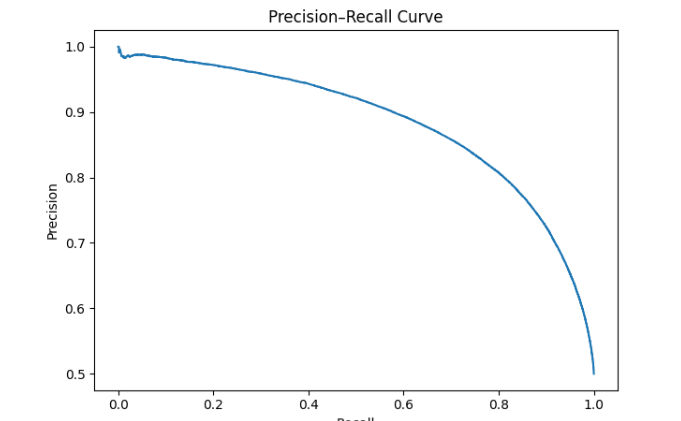
|  |  |
| --- | --- |
| Metric | Value |
| Accuracy | 0.804 |
| Precision | 0.797 |
| Recall | 0.815 |
| F1-score | 0.806 |
| ROC-AUC | 0.884 |
| Matthews Corr. Coef. | 0.608 |
| Cohen's Kappa | 0.608 |

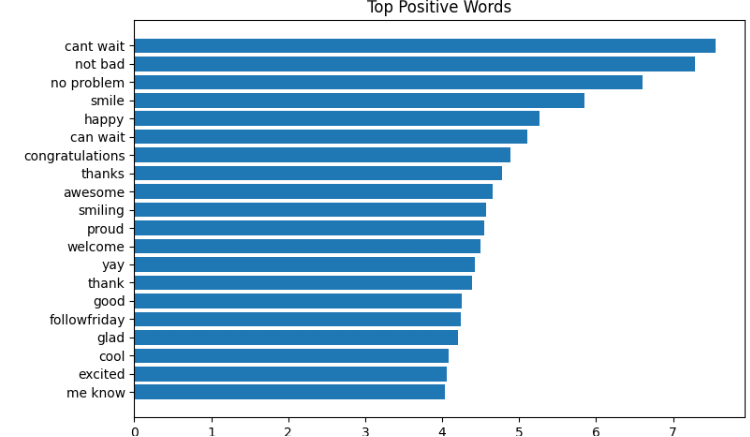
Summary: The model achieves about 80.4% accuracy and an AUC of 0.884. Precision and recall are balanced, indicating stable performance across classes. Matthews correlation and Cohen's kappa around 0.61 show substantial agreement beyond chance.

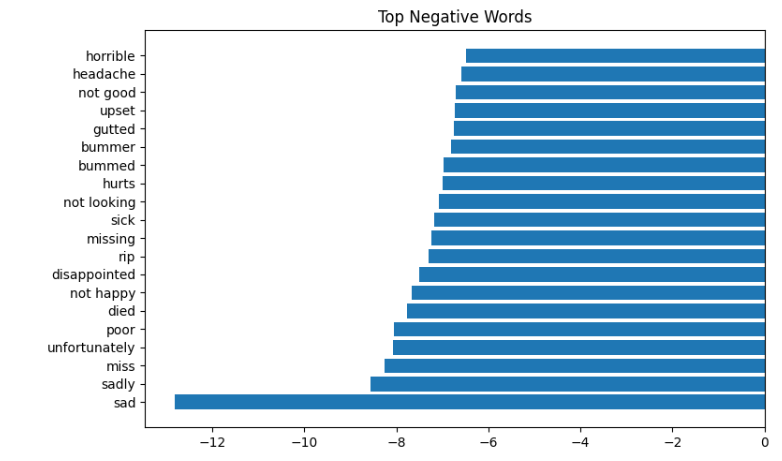
## 5. Visualizations

Included below are key evaluation plots generated during analysis.









## 6. Error Analysis

Confusion matrix (from evaluation):  
 - True Negatives: 94,949  
 - False Positives: 24,802  
 - False Negatives: 22,114  
 - True Positives: 97,596

Observations-

* The model’s errors are largely balanced between positive and negative predictions, suggesting it does not favor one sentiment over the other.
* Misclassifications often arise from tweets with **ambiguous or mixed tones**, where the language conveys both positive and negative emotion (e.g., “not bad but could be better”).
* **Sarcasm and irony** remain difficult for the model to interpret, leading to false positives and negatives.
* Incorporating **bigrams** in the TF-IDF representation notably improved handling of negations and short expressions (e.g., distinguishing “not good” from “good”).

## 7. Feature Interpretation

The most influential tokens provide insight into how the model identifies sentiment patterns in text:

* **Positive indicators:** cant wait, not bad, no problem, smile, happy, congratulations, awesome, glad, excited
* **Negative indicators:** sad, sadly, miss, unfortunately, disappointed, sick, horrible, headache, upset

These features demonstrate that the classifier effectively captures **emotionally charged words** and **common sentiment expressions**, showing that lexical polarity plays a central role in tweet-level sentiment detection.

## 8. Key NLP Topics Demonstrated

1. **Text Pre-processing:** Cleaning and normalization using regular expressions to remove noise and standardize text.
2. **Vectorization:** Representing text through TF-IDF, capturing both single words and short phrases (n-grams).
3. **Modeling:** Using a logistic regression classifier as a strong, interpretable baseline for sentiment tasks.
4. **Evaluation:** Employing diverse metrics (Precision, Recall, F1, ROC-AUC, MCC, Cohen’s Kappa) and visual diagnostics (ROC and PR curves).
5. **Interpretability:** Analyzing model coefficients to understand which words drive sentiment decisions.
6. **Deployment Readiness:** Saving the trained pipeline for easy integration into applications or dashboards.

## 9. Conclusions

Classical NLP techniques continue to perform effectively for large-scale sentiment classification on social media data.  
This pipeline achieved an AUC of 0.884, demonstrating strong separation between positive and negative sentiments while remaining computationally efficient.  
While modern transformer-based models (like BERT) could further improve performance particularly on nuanced language such as sarcasm they require significantly greater resources..