Sentiment Analysis on Twitter Data

Using ML & Deep Learning on the Sentiment140 Dataset

Presented by the data science team • May 2025

Business Problem: Understanding Twitter Sentiment

In today's digital landscape, businesses need to automatically and accurately analyze customer sentiment from millions of tweets to gain real-time insights into customer satisfaction, and market trends—turning unstructured social media data into actionable intelligence.

Technical & Business Objectives

Technical Objectives

- Clean and prepare Sentiment140 dataset
- Build and compare 3 models: Logistic Regression (TF-IDF), LSTM (Word2Vec), DistilBERT
- Achieve at least 80% accuracy
- Deploy the best model on HuggingFace

Business Objectives

- Understand customer feelings about products or services
- Help track brand reputation effectively
- Save time by automating sentiment analysis
- Provide insights to improve decision-making

Data Overview & Preprocessing

Sentiment140 Dataset

- 1.6 million labeled tweets
- Binary sentiment: negative & positive
- Fields: text, user, date, sentiment

Preprocessing Steps

- Lowercase normalization
- Remove URLs, emojis, mentions
- Tokenization & punctuation cleanup

Model Comparison & Performance

Model	Features	Accuracy	Precision	Recall
Logistic Regression	TF-IDF	81%	0.81	0.81
LSTM	Word embeddings	83%	0.83	0.83
DistilBERT	Pretrained Transformer	**86%**	**0.86**	**0.86**

For our use case, **Recall is more important than Precision**. We want to detect as many relevant sentiments as possible, especially negative ones. Missing a negative tweet could mean overlooking customer frustration, which is risky for businesses.

Data Preparation Pipeline

Load Data

Acquire Sentiment140 dataset from Kaggle

Clean Text

Normalize, remove noise, tokenize

Train-Test Split

80% training, 20% testing

Tokenization

Methods vary by model (TF-IDF, Word2Vec, BERT)

Deployment & Resources

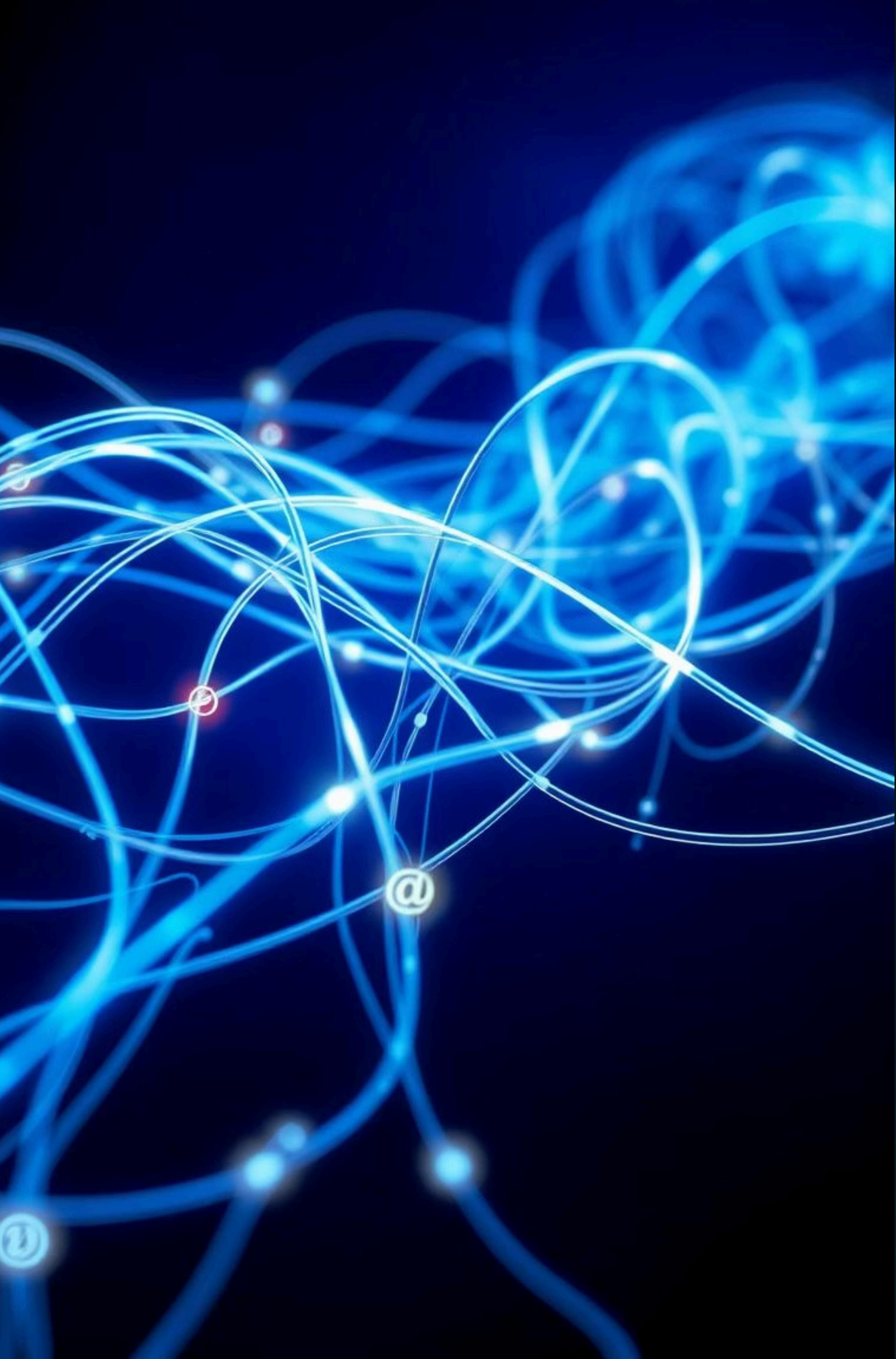
Deployment Platform

HuggingFace public model repo

Deployment Platform Link

Resources

- Google Colab Free Tier
- Open Source Tools: TensorFlow, PyTorch, scikit-learn



Challenges Faced

Messy Tweets

Tweets had emojis, slang, and symbols that were hard to clean.

Slow Training

Training big models like DistilBERT was slow and sometimes crashed.

Overfitting

Some models learned too much from training data and didn’t perform well on new data.

Practical Applications

1. **Customer Service Enhancement:** Flag negative tweets for quick response and spot common complaints.
2. **Marketing Optimization:** Measure sentiment shifts and identify effective messaging.
3. **Product Development:** Collect feedback to improve features and fix pain points.
4. **Competitive Analysis:** Compare brand sentiment with competitors and find opportunities.

Conclusion

Machine learning enables deep insights from social media sentiment data.

TF-IDF + Logistic Regression offers fast, resource-light sentiment detection with 81% accuracy.

BERT models boost accuracy to 87%, requiring more computing power and time.

Business needs and resources should dictate model selection, or use a hybrid approach for best results.

Thank You

- Salma Anwer Anwer
- Mahmoud Elbahy
- Ahmed Mohamed
- Mohamed Ehab
- Zeyad Tamer
- Micheal George