



VIT[®]
BHOPAL

VITYARTHI PROJECT REPORT

NAME – SAKSHAM SRIVASTAVA

REGISTRATION NUMBER – 25BCY10196

TOPIC – SENTIMENT ANALYSIS OF SOCIAL
MEDIA POSTS USING MACHINE LEARNING

INTRODUCTION

Many people express their thoughts and feelings through social media platforms such as Twitter, Instagram, and YouTube to show whether they are happy, sad, angry, or neutral.

This project uses AI and ML to automatically detect the sentiment behind a social media post.

PROBLEM STATEMENT

- Reads a text post
- Processes the words
- Understands the meaning
- Classifies the sentiment into Positive, Negative, or Neutral

OBJECTIVE

- Data preprocessing
- Feature extraction
- Classification algorithms

- Model evaluation techniques

EXPECTED RESULT

The model should correctly detect emotions in text with an accuracy between 85%–95%, depending on the dataset.

The system can be used by:

- Businesses to understand customer feedback
- Students to learn NLP
- Companies to analyze brand sentiment
- Users to understand trending emotions on social media

APPLICATIONS

- Customer review analysis
- Product rating improvement

- Political opinion study
- Mental health monitoring
- Hate speech detection
- Chatbot emotion understanding

CONCLUSION

This project successfully demonstrates how Artificial Intelligence and Machine Learning help machines understand human emotions.

Using NLP techniques and classification algorithms, the model can automatically analyze the sentiment of social media posts. This showcases the powerful connection between AI and real-world communication.

REFRENCES

Books

1. Liu, B. (2015). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press.
2. Jurafsky, D., & Martin, J. H. (2023). *Speech and Language Processing* (3rd ed.). Draft. Stanford University.

Research Papers

3. Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2), 1–135.

4. Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. Stanford University Technical Report.
5. Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. Proceedings of LREC, 1320–1326.

Web Articles / Documentation

6. IBM. (2023). What is sentiment analysis? IBM Cloud Learning Hub.
<https://www.ibm.com>
7. Google Developers. (2023). Natural Language Processing Overview.
<https://developers.google.com>
8. Kaggle. (2023). Sentiment Analysis Datasets. <https://www.kaggle.com>

Datasets

9. Sentiment140 Dataset. (2009). Available at:
<https://www.kaggle.com/kazanova/sentiment140>
10. Twitter US Airline Sentiment Dataset. (2015). Kaggle.
<https://www.kaggle.com/crowdflower/twitter-airline-sentiment>

Online Courses

11. Andrew Ng. (2022). Natural Language Processing Specialization. Coursera.
<https://www.coursera.org>
12. Stanford CS224N. (2021). Natural Language Processing with Deep Learning. Stanford University.

CODES

```
▶ # Sentiment Analysis of Social Media Posts using Machine Learning
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

# Sample social media posts (you can replace with dataset)
posts = [
    "I love this product! It's amazing.",
    "Worst experience ever, I hate it.",
    "Feeling happy today!",
    "This is so bad, very disappointed.",
    "Great service, I am satisfied.",
    "Terrible quality, not worth it."
]

# Corresponding labels (1 = Positive, 0 = Negative)
labels = [1, 0, 1, 0, 1, 0]

# Step 1: Convert text to numerical vectors
tfidf = TfidfVectorizer()
X = tfidf.fit_transform(posts)

# Step 2: Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, labels, test_size=0.3, random_state=42)

# Step 3: Train ML model
model = LogisticRegression()
model.fit(X_train, y_train)

# Step 4: Predict
pred = model.predict(X_test)

# Step 5: Accuracy
print("Accuracy:", accuracy_score(y_test, pred))

# Test on a new social media post
new_post = ["I really enjoyed using this app!"]
new_vec = tfidf.transform(new_post)
prediction = model.predict(new_vec)

if prediction[0] == 1:
    print("Sentiment: Positive")
else:
    print("Sentiment: Negative")
```

- Accuracy: 0.5
Sentiment: Negative

GIT HUB LINK:-
<https://github.com/srivastavasaksham1502-collab>
