The Role of VC-Dimension in Overfitting Prevention for Sentiment Analysis(Develop an optimized sentiment analysis model using VC- dimension concepts)

Machine Learning
(CSL7620)

Submitted by:
Anurag Samota (M25CSE007)
Shubham Jindal (M25CSE031)
Satyam Pal (M25CSA026)

Contributions:

Satyam Pal - Data collection, preprocessing and cleaning, generated learning curve plots, prepared README
Data_utils.py, hf_utils.py, plot_utils.py

Shubham Jindal - TF-IDF, performance metrics, Implemented Logistic Regression, SVM, and Decision Tree models
train_utlis.py, vc_utils.py, plot_utils.py , vc.ipynb

Anurag Samota - Developed Streamlit frontend (app.py), integrated model pipeline, compared results, created report
app.py, infer_utils.py, requirements.txt, plot_utils.py

ABSTRACT

This project aims to develop accurate models in the field of sentiment analysis using Machine Learning techniques and theoretical insights from the Vapnik–Chervonenkis (VC) Dimension. It focuses on building models capable of classifying text sentiments—positive, negative, or neutral—based on

movie reviews from the IMDB dataset. The implementation includes classical ML algorithms such as Logistic Regression, Support Vector Machine (SVM), and Decision Tree, enabling a comparative study of their predictive performance.

The evaluation of these models is carried out using standard performance metrics such as Accuracy, Precision, Recall, and F1-score, along with an analytical assessment of model complexity using VC Dimension theory. The dataset used in this project is open-source and undergoes preprocessing steps such as text cleaning and feature extraction using TF-IDF vectorization.

Introduction to Sentiment Analysis

Sentiment Analysis, also known as Opinion Mining, is a branch of and Machine Learning (ML) that deals with identifying and classifying emotions expressed in textual data. Its primary objective is to determine whether a piece of text conveys a positive, negative, or neutral sentiment.

With the rise of digital communication, people share their thoughts and opinions online through reviews, tweets, blogs, and comments. Analyzing this enormous volume of unstructured text manually is nearly impossible. Hence, automated sentiment analysis systems are developed to process and classify text efficiently.

Technologies and Tools Used

1 Python Programming Language
2 Streamlit for User Interface
3 Hugging Face Datasets (IMDB and Twitter)
4 Matplotlib and Seaborn for Visualization

Datasets and Preprocessing

1 IMDB Movie Review Dataset
2 Twitter Sentiment Dataset
3 Data Cleaning and Normalization
4 TF-IDF Vectorization 5 Dataset Splitting (Training and Testing Sets)

Machine Learning Models Used

1 Logistic Regression
2 Support Vector Machine (SVM)
3 Decision Tree

Model Evaluation and Comparison

1 Evaluation Metrics
2 Accuracy 3 Precision
4 Recall
5 F1-Score

VC Dimension Analysis

Understanding VC Dimension in Machine Learning

The Vapnik–Chervonenkis (VC) Dimension is a fundamental concept in statistical learning theory that measures the capacity or complexity of a model — i.e., its ability to fit a variety of data patterns.

Effect of TF-IDF Feature Count on Model Complexity

In text classification tasks, the number of features (terms extracted from TF-IDF vectorization) acts as a proxy for VC Dimension.
As the number of features increases, the model's hypothesis space expands, allowing it to learn more intricate relationships.

Results and Discussion

Both Linear SVM and Logistic Regression produce strong, comparable results on the local sentiment data. The SVM shows a small but consistent edge in test accuracy in most configurations, while Logistic Regression offers similar mean accuracy with somewhat lower variance across runs and hyperparameters.

Logistic Regression tends to be marginally faster to train and more stable with default solver settings; Linear SVMs (LinearSVC) can be slightly slower when sweeping many feature sizes but often yield the top test scores.

The VC-dimension proxy (TF-IDF Max Features) is an effective, interpretable control knob: choose a moderate vocabulary size rather than the largest possible to avoid overfitting—especially on smaller datasets.

Results depend on the dataset size and preprocessing. The HF comparison is affected by label normalization (dropping neutrals) and domain mismatch. The current experiments use linear models and TF-IDF only; non-linear or contextual models (fine-tuned transformers) were not evaluated here and may change relative ordering.

Model Performance Summary

Dataset	Model	Accuracy	Observations
IMDB	Logistic Regression	90.03%	Stable across feature sizes, minimal overfitting.
IMDB	SVM	89.83%	Best accuracy but required a High VC dimension.
IMDB	Decision Tree	70.4%	Overfitting is observed at higher dimensions.
Twitter	Logistic Regression	84.14%	Performed well with fewer features.
Twitter	SVM	85.36%	Balanced accuracy and generalization.
Twitter	Decision Tree	48.61%	Sensitive to noise and small sample variations.