Twitter Sentiment Analysis

Depression Detection

Aris Tsilifonis – mtn2323

Agenda

- Abstract
- Implemented methods(Theory)
- Approach
- Data Preprocessing
- Data Analysis
- Feature extraction
- Evaluation-metrics
- Hyperparameter optimization
- Conclusions

Abstract

- Sentiment analysis is a machine learning technique that detects polarity(a positive or negative opinion) in a paragraph, phrase or a whole document
- In today's age, people interact through social media at a high rate. Twitter is being one of the most popular platforms where people exchange opinions about various topics. However, an individual tweet can reveal many aspects of the user's psychology.
- Detecting if an individual tweet contains **depression or not** can be proved very useful since a lot of safety measures can be taken thereafter.

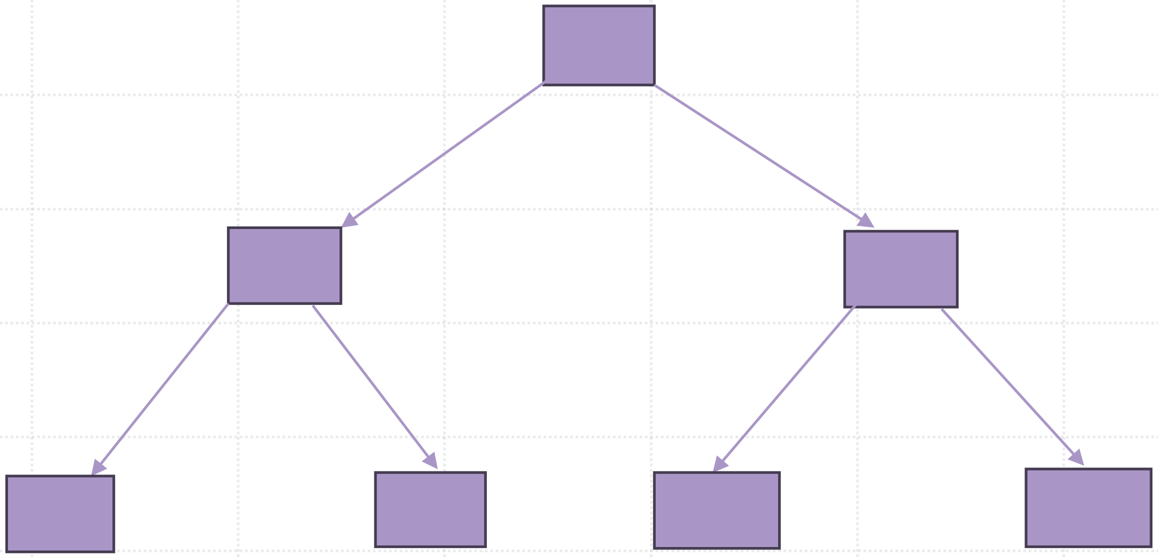
Implemented Methods
Decision Trees

- The decision tree stands as a widely recognized and highly effective method for both classification and prediction tasks. It is depicted as a tree-like model in flowchart form. In this structure, each internal node signifies a test on a feature, every branch indicates the possible result of this test, and the leaf nodes, or terminal nodes, contain the classification labels.

- Decision trees offer the advantage of executing classifications with minimal computational demand.

- Furthermore, they are versatile, capable of processing variables that are either continuous or categorical in nature.

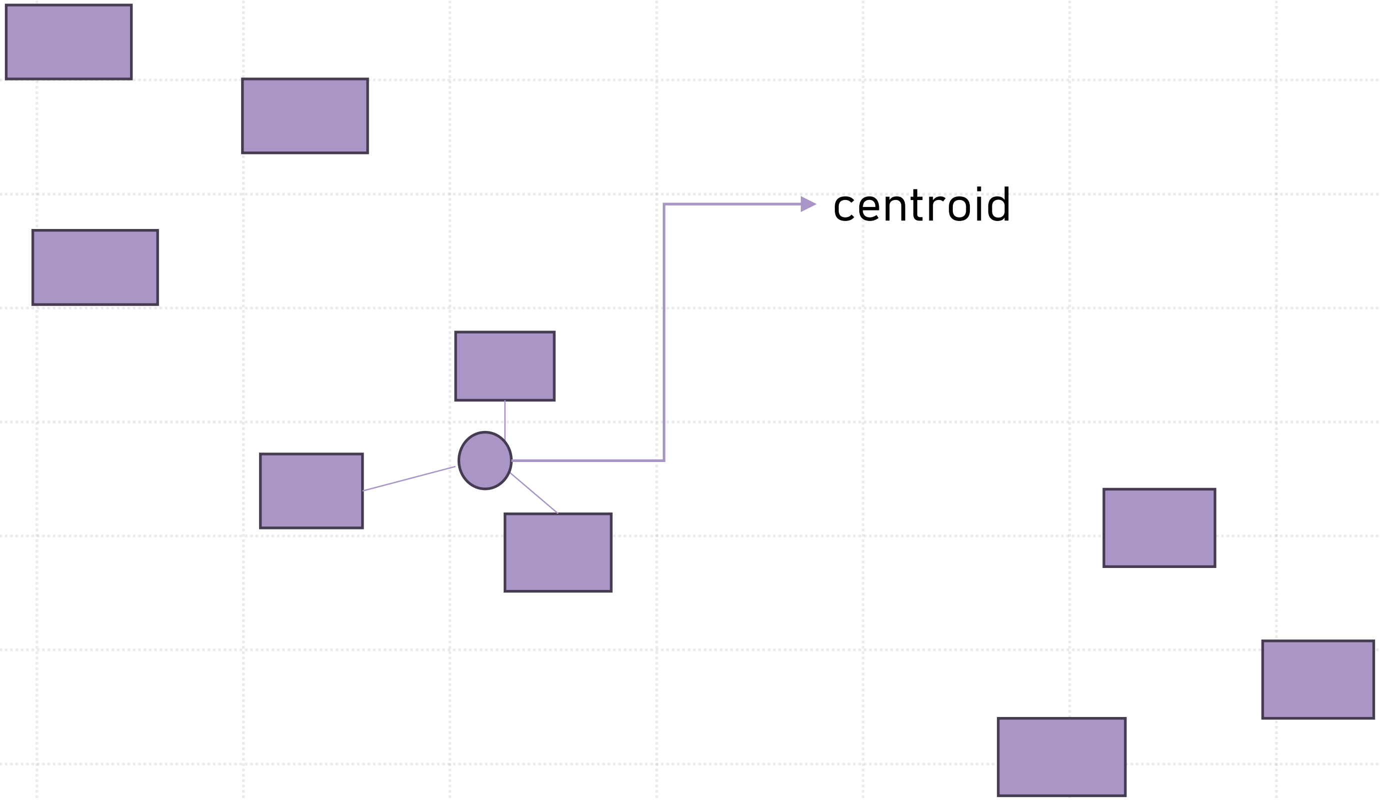
- Decision Trees



K-nearest neighbors

- The K-nearest neighbors (KNN) algorithm is a supervised machine learning technique applicable to both classification and regression predictive tasks.

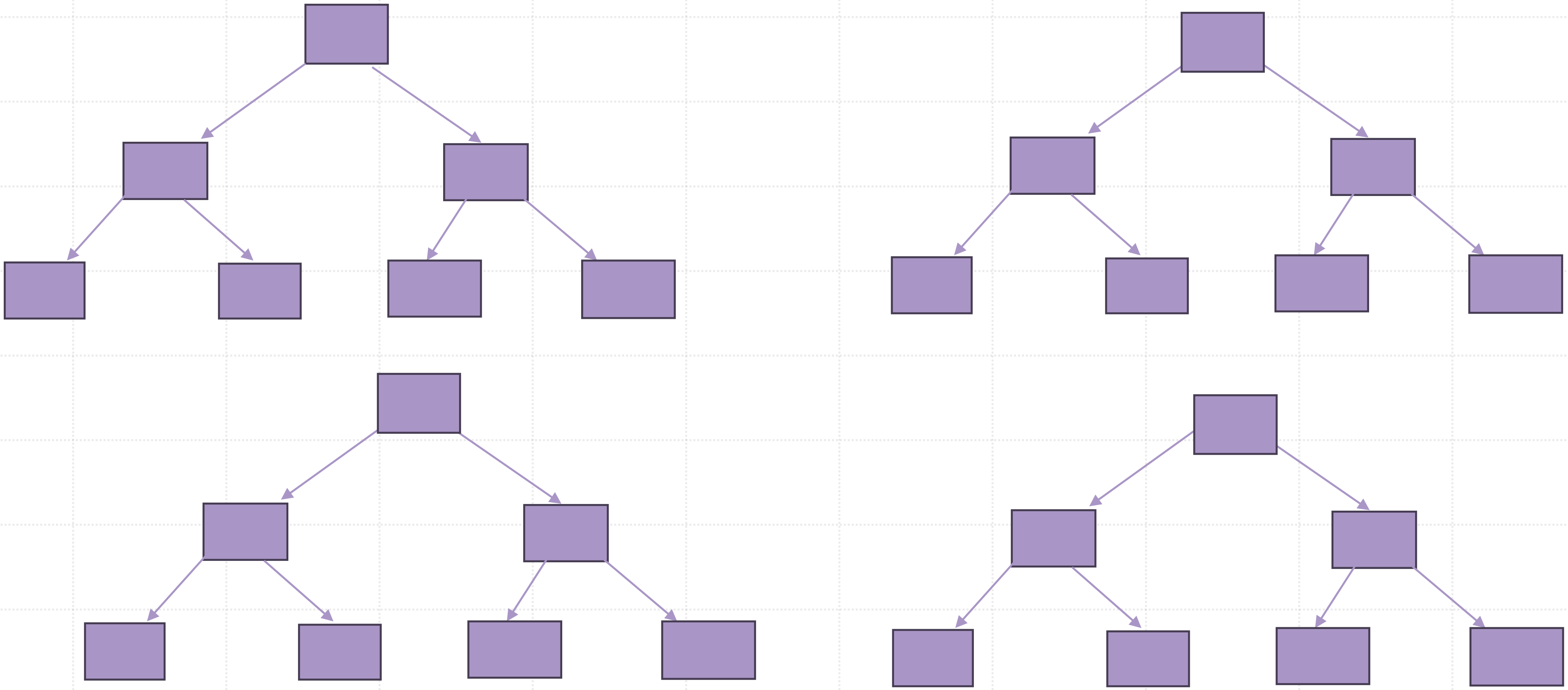
- The essence of the KNN algorithm is to predict the value of new data points by analyzing their 'feature similarity' with existing points in the training set. This implies that a new data point is given a value according to its proximity to the points already present in the training dataset.


centroid

Random Forest

- Random forest is an ensemble method that combines a multitude of decision trees to make predictions.
- In this approach, each tree within the random forest contributes its prediction, with the most frequently predicted class becoming the final output of the model.
- The underlying principle of the random forest is straightforward yet effective: By pooling the predictions of several independent models (trees), the collective decision is typically more accurate than that of any single tree within the ensemble.

- Random Forest

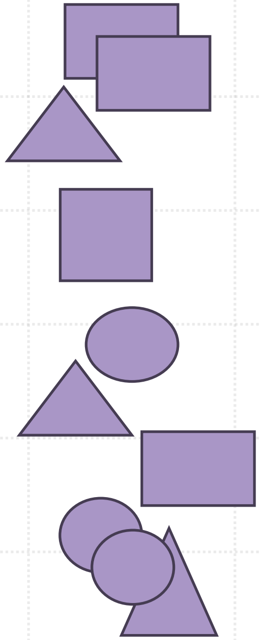


Multinomial Naïve Bayes

- Multinomial Naive Bayes (MultinomialNB) employs the frequencies of words in a document as its features or predictors for classification. This approach is predominantly applied to text classification challenges.
- It assesses the probability of each category for a given text and selects the category with the highest probability for its output. The presence or absence of one feature is considered independent of any other feature's presence or absence. This "naive" assumption of feature independence simplifies probability calculations, enhancing the algorithm's computational efficiency.

Bayes
Theorem

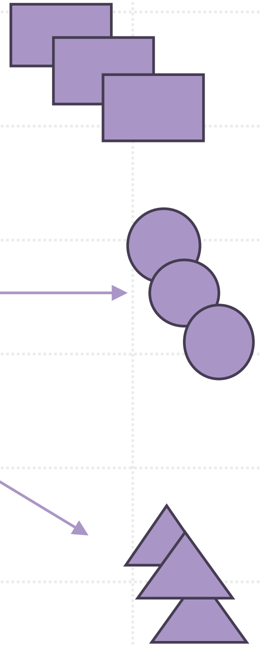
Semi-
supervised

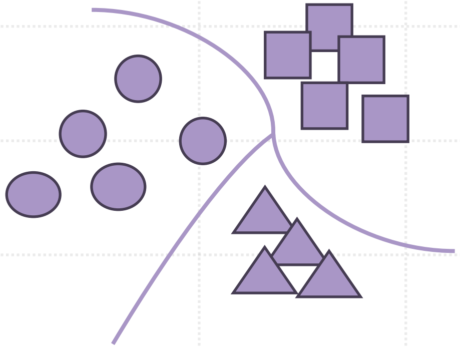


$$P(C_k|W) = logP(C_k) + \sum_{i=1}^{n}(w_i * \log p(w_i|C_k)$$

prior

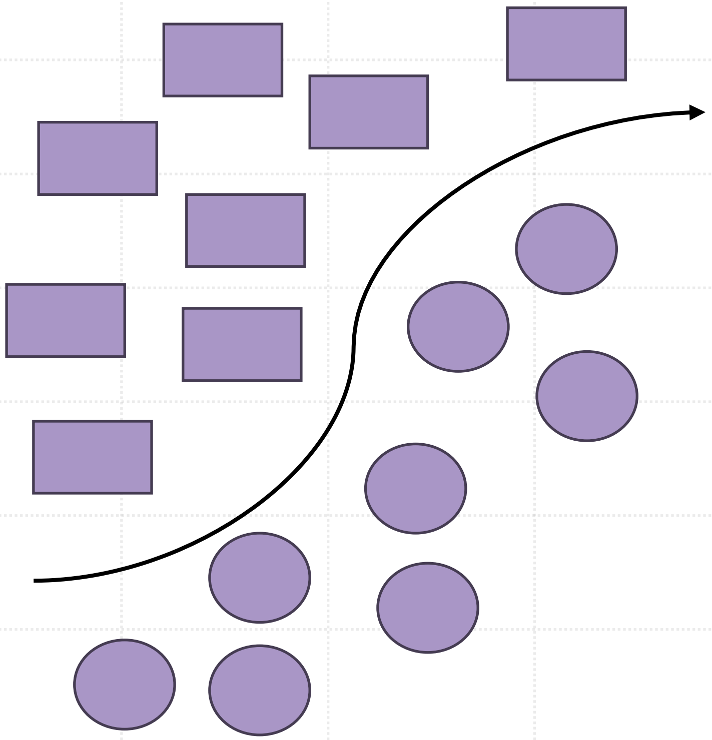
likelihood





Logistic Regression

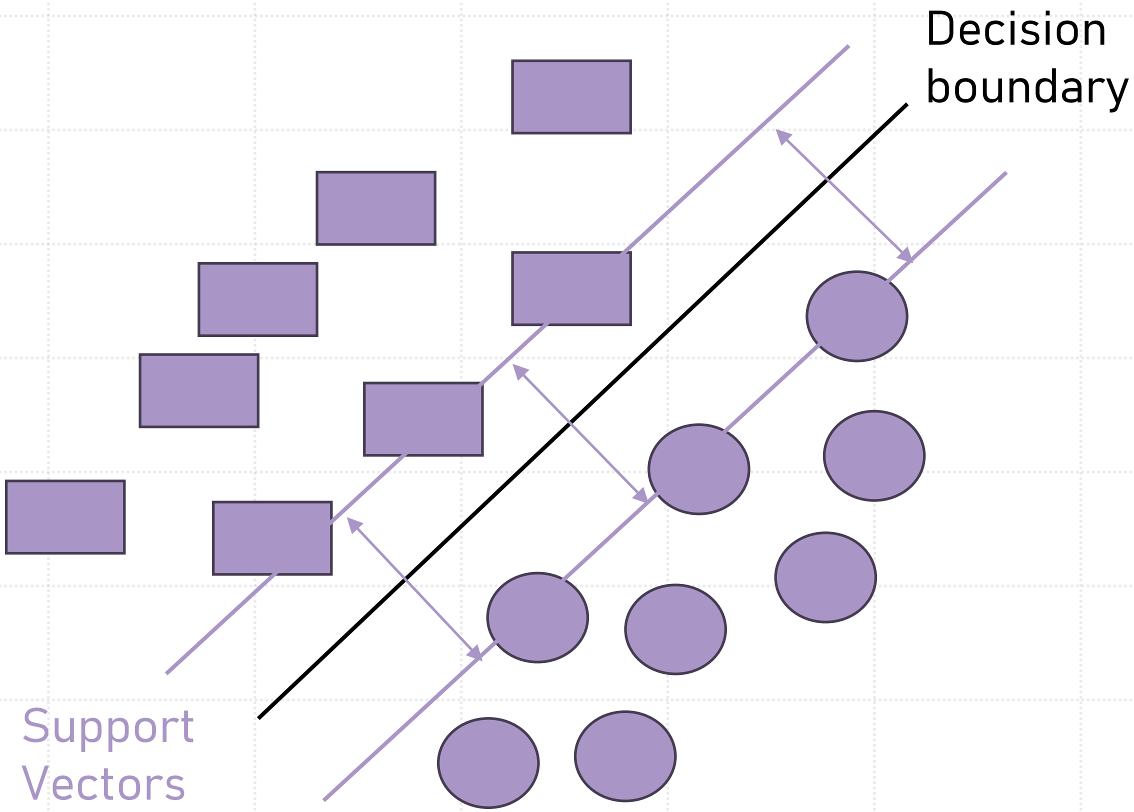
- Logistic Regression is a statistical method used for binary classification. It models the probability that a given input belongs to a particular category.
- Logistic Regression works by fitting a logistic curve to the data and using the sigmoid function to estimate probabilities, which are then mapped to the closest class.
- This technique is widely used for predictive analysis to determine outcomes that have two possible states like yes/no, win/lose, alive/dead.



$$p = \frac{1}{1 + e^{-(b_0 + b_1 x)}}$$

Supported Vector Machines

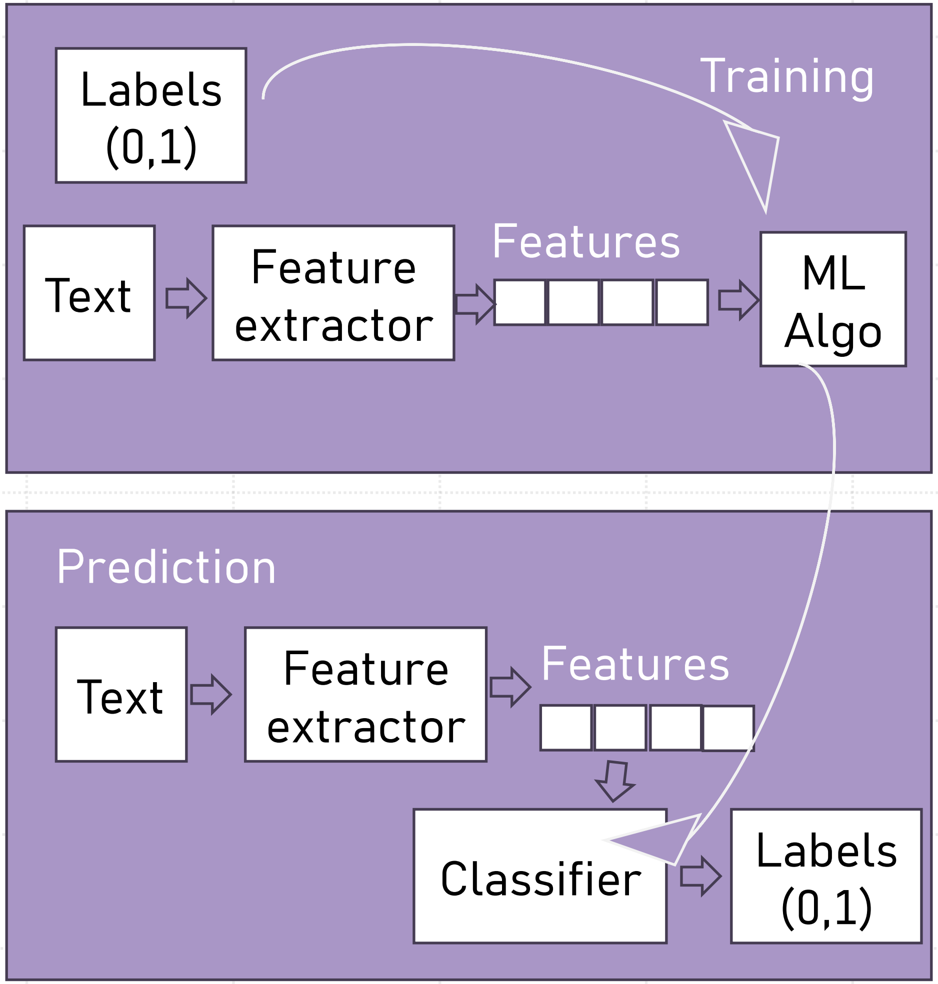
- Support Vector Machines (SVM) are a set of supervised learning methods used for classification, regression, and outliers detection. The core principle of SVM is to find the hyperplane that best divides a dataset into classes.
- The strength of SVM lies in its use of kernels, which allow it to efficiently perform a non-linear classification, thereby transforming the input space into a higher dimensional space.
- SVM is particularly well-suited for complex but small- or medium-sized datasets, offering high accuracy and robustness against overfitting, especially in cases where the dimensionality of the data exceeds the number of samples.


Decision boundary
Support Vectors

Approach

- During the training phase, the model is taught to link text with the appropriate outcome using the training samples provided. The **feature extractor** converts the text input into a feature vector, which is then inputted into the machine learning algorithm to create a model.

- In the prediction phase, **the same** feature extractor converts new, unseen text inputs into feature vectors. These vectors are inputted into the trained model, which then produces the predicted labels.

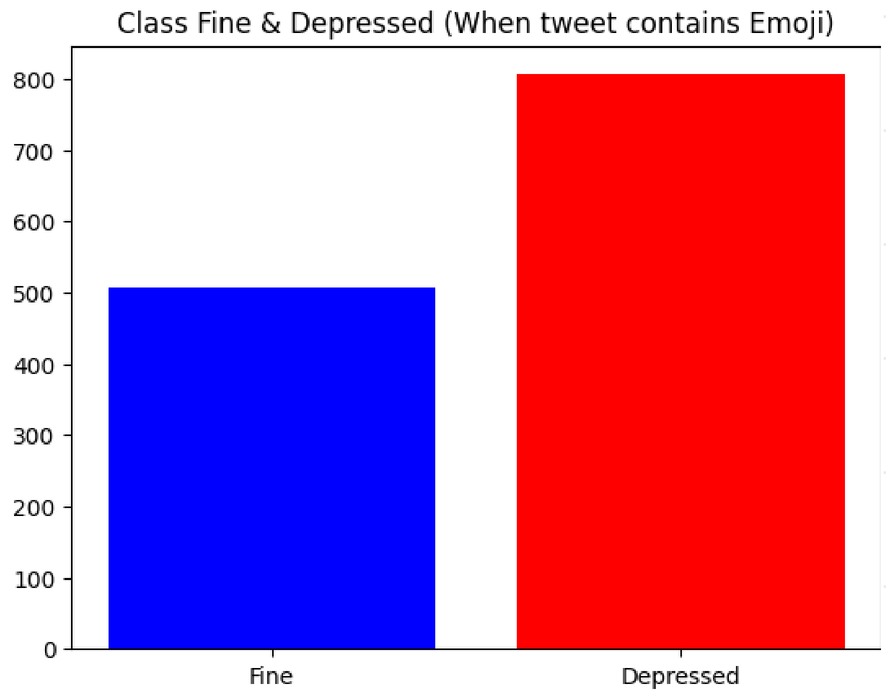


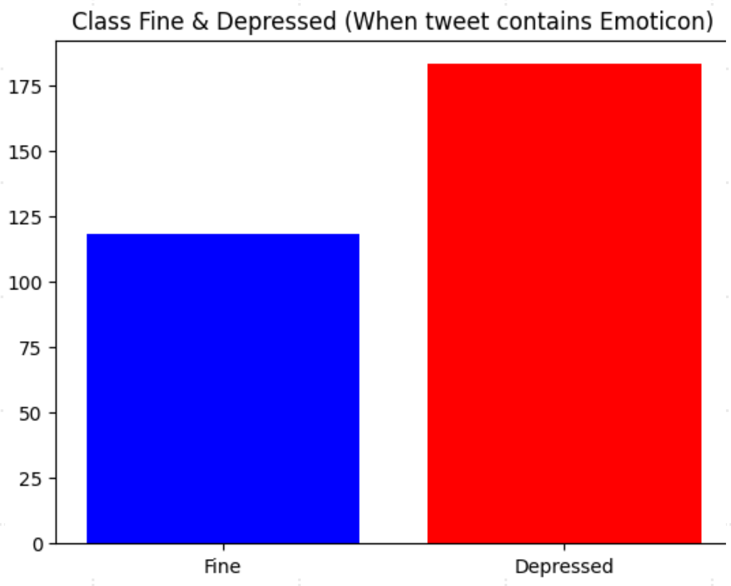
Data preprocessing

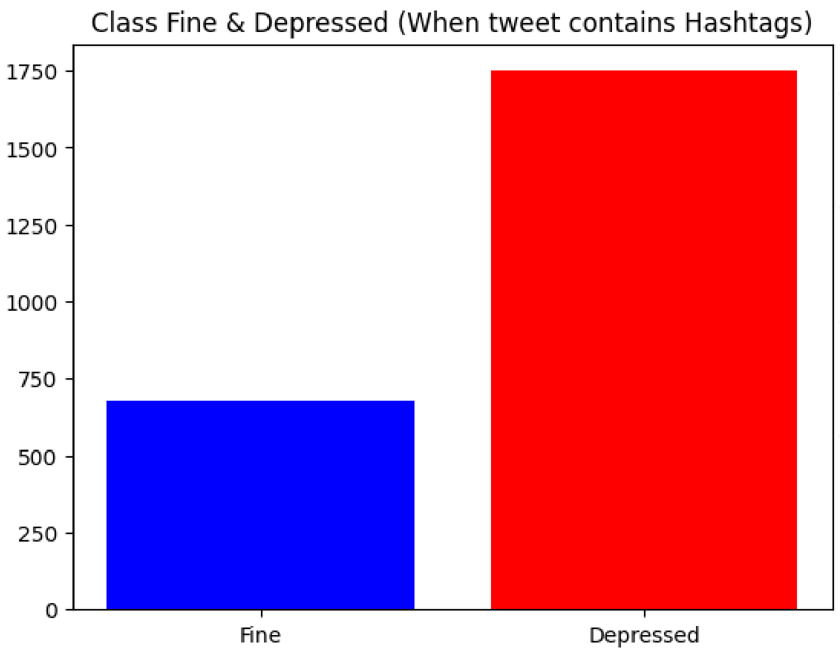
1. Hold links/mentions/hashtags(optional)
2. Hold emojis/emoticons(optional)
3. Remove non-alphanumeric characters
4. Apply Blob Techniques
5. Lemmatization & Remove Stopwords

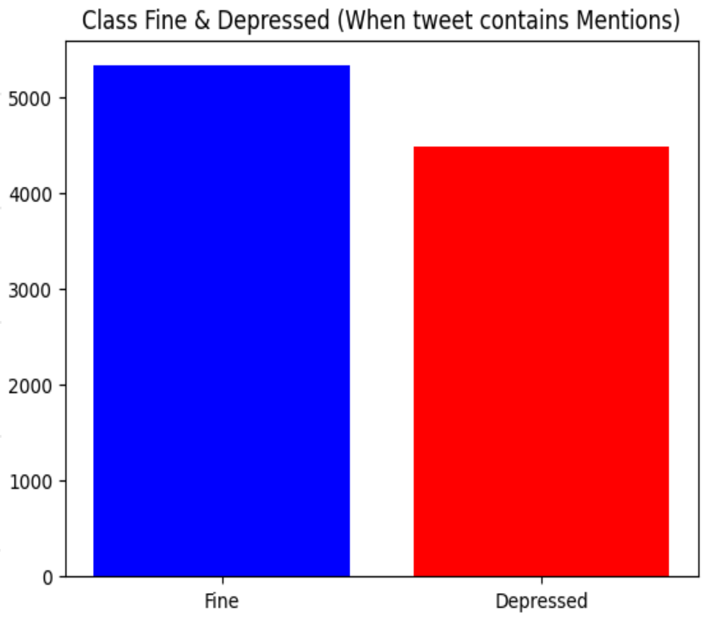
Data Analysis

Percentage of rows containing Depression
50.0 %
Percentage of rows not containing Depression
50.0 %   [Balanced dataset]









Data Analysis

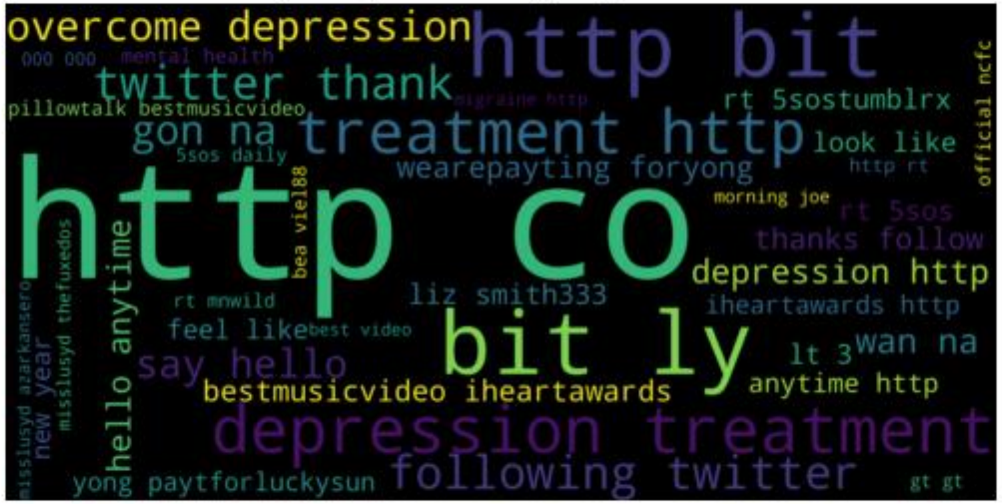
Depression dataframe's
Most popular emojis

1. 😂
2. ❤️
3. 😭
4. RU
5. 😍

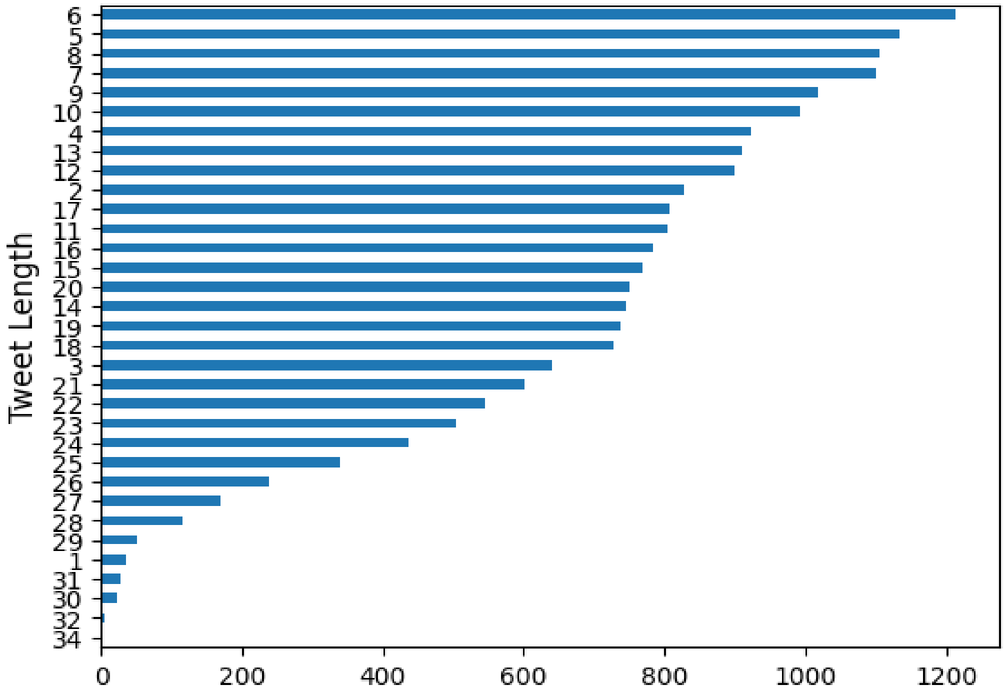
Non-Depression dataframe's
Most popular emojis

1. 😂
2. 😭
3. 👀
4. 🎉
5. 🔥

Most frequently occurring bigrams for tweets



Number of tweets(Both classes)



Feature Extraction

- **Count Vectorizer** : It works by breaking down text into words (or tokens) and counting how many times each word occurs.
- **TF-IDF vectorizer** : Stands for Term Frequency-Inverse Document Frequency, is a numerical statistic used to indicate how important a word is to a document in a collection or corpus
    - Term Frequency (TF): This measures how frequently a term occurs in a document
    - Inverse Document Frequency (IDF): This measures the importance of the term across a set of documents. It's calculated by taking the logarithm of the number of documents in the corpus divided by the number of documents where the specific term appears.

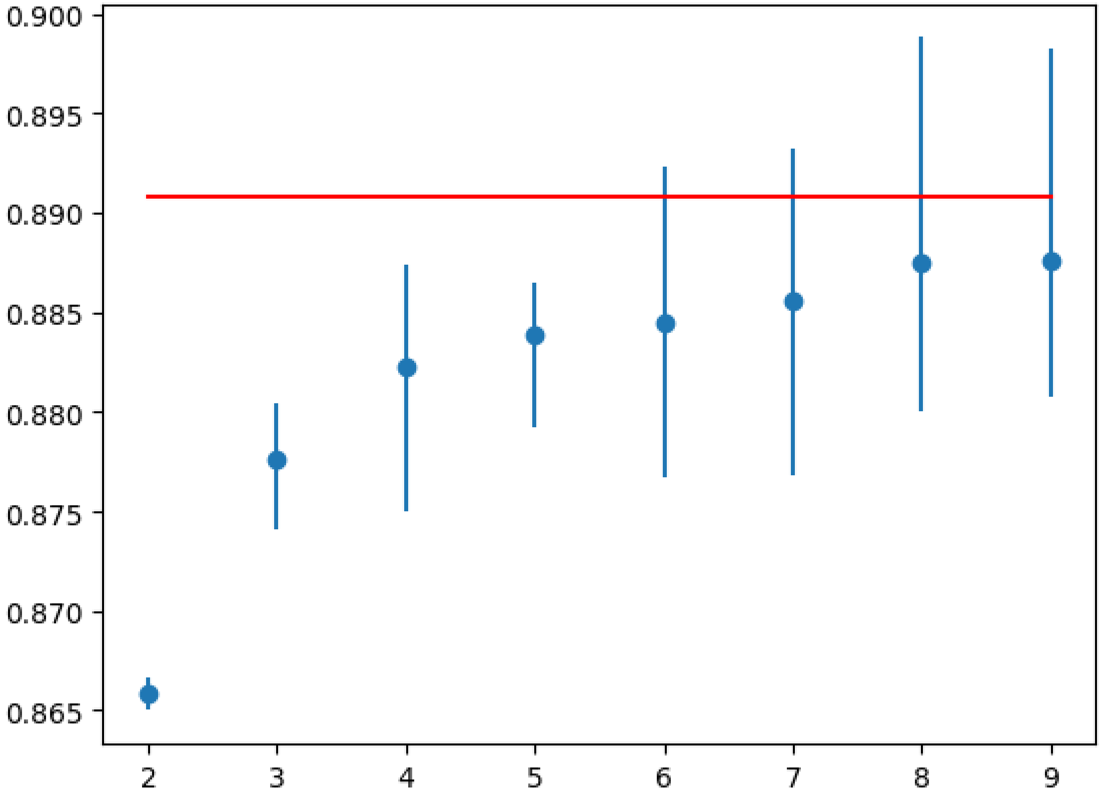
Performance overview

	(without emojis, links, mentions) preprocessed text	unchanged	emoji textual replacement	(with emojis, links, mentions) preprocessed text
MNB	0.868	0.8615	0.82	0.8765
KNN	0.762	0.786	0.746	0.7725
DT	0.7885	0.7135	0.731	0.8165
RF	0.7955	0.7525	0.7395	0.827
SVM	Not Tested	Not Tested	Not Tested	0.877

Evaluate machine Learning Models

Multinomial Naïve Bayes

- Stratified K-Fold cross-validation(Maintain the same ratio in the sample as in original dataset)



Ideal: 0.891
> folds=2, accuracy=0.866 (0.865,0.867)
> folds=3, accuracy=0.878 (0.874,0.880)
> folds=4, accuracy=0.882 (0.875,0.887)
> folds=5, accuracy=0.884 (0.879,0.886)
> folds=6, accuracy=0.884 (0.877,0.892)
> folds=7, accuracy=0.886 (0.877,0.893)
> folds=8, accuracy=0.887 (0.880,0.899)
> folds=9, accuracy=0.888 (0.881,0.898)

Calculate the ideal test condition using Leave-One-Out

Confusion Matrix

Multinomial NB

	Fine (predicted)	Depression (Predicted)	Sum
Fine (true)	819	155	974
Depression (true)	91	935	1026
Sum	910	1090	2000

Logistic Regression

	Fine (predicted)	Depression (Predicted)	Sum
Fine (true)	829	145	974
Depression (true)	155	871	1026
Sum	984	1016	2000

Classification Report
Multinomial Naïve Bayes

- Accuracy Score: 0.877

	precision	recall	f1-score	support
0	0.90	0.84	0.87	974
1	0.86	0.91	0.88	1026
accuracy			0.88	2000
macro avg	0.88	0.88	0.88	2000
weighted avg	0.88	0.88	0.88	2000

Classification Report

Logistic Regression

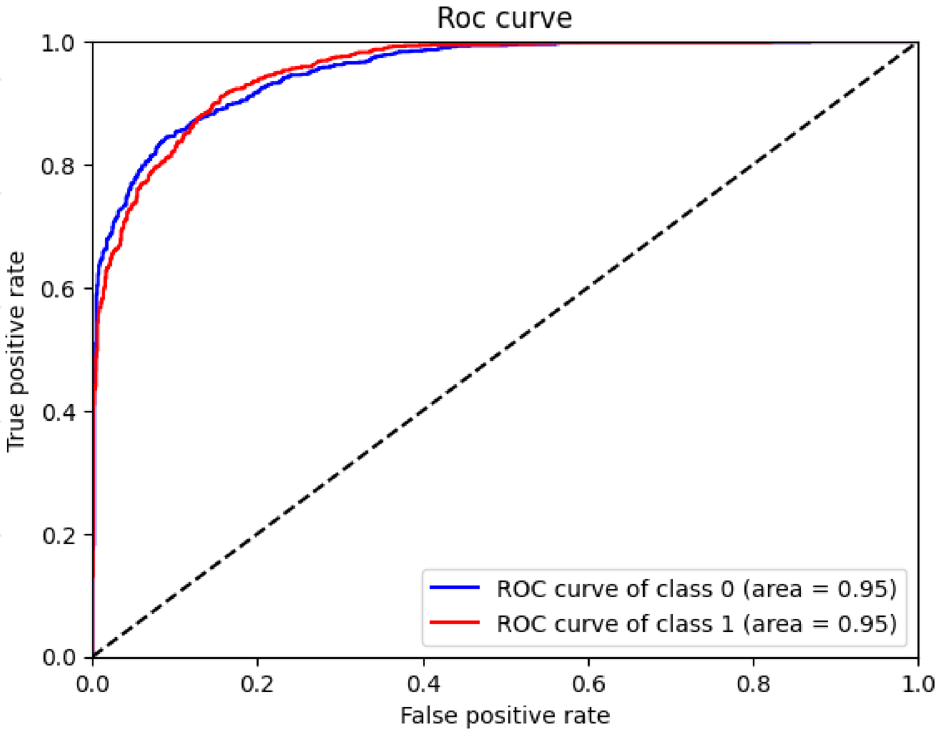
- Accuracy Score: 0.85

	precision	recall	f1-score	support
0	0.84	0.85	0.85	974
1	0.86	0.85	0.85	1026
accuracy			0.85	2000
macro avg	0.85	0.85	0.85	2000
weighted avg	0.85	0.85	0.85	2000

ROC-Curves

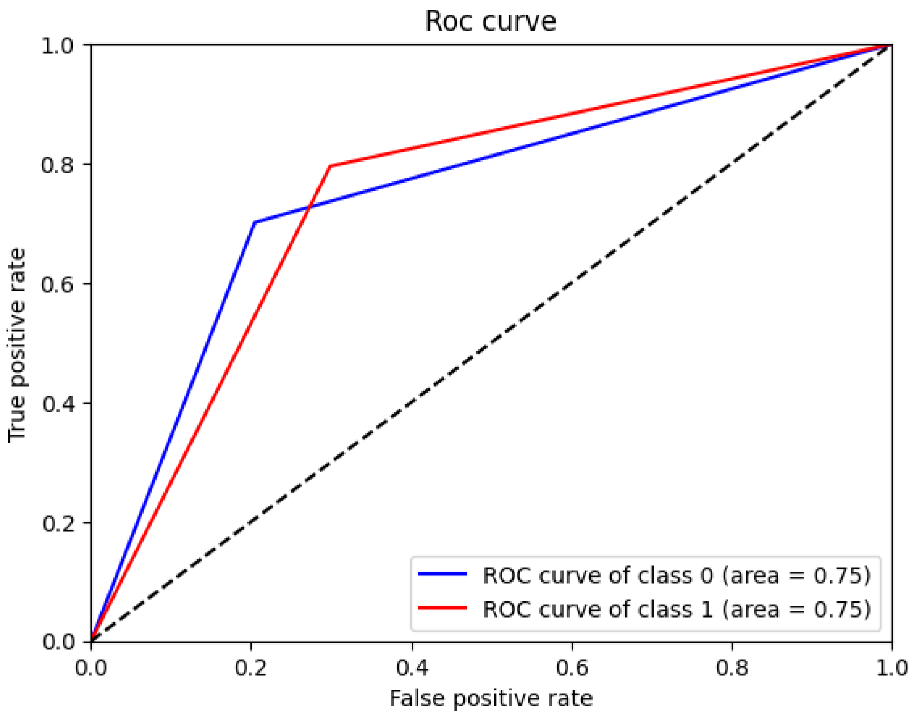
Multinomial Naive Bayes

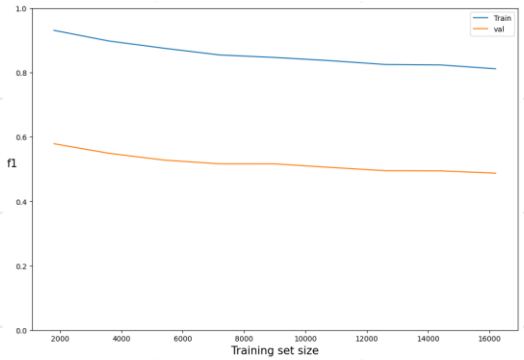
auc for the class 0 0.953
auc for the class 1 0.953



Decision Tree

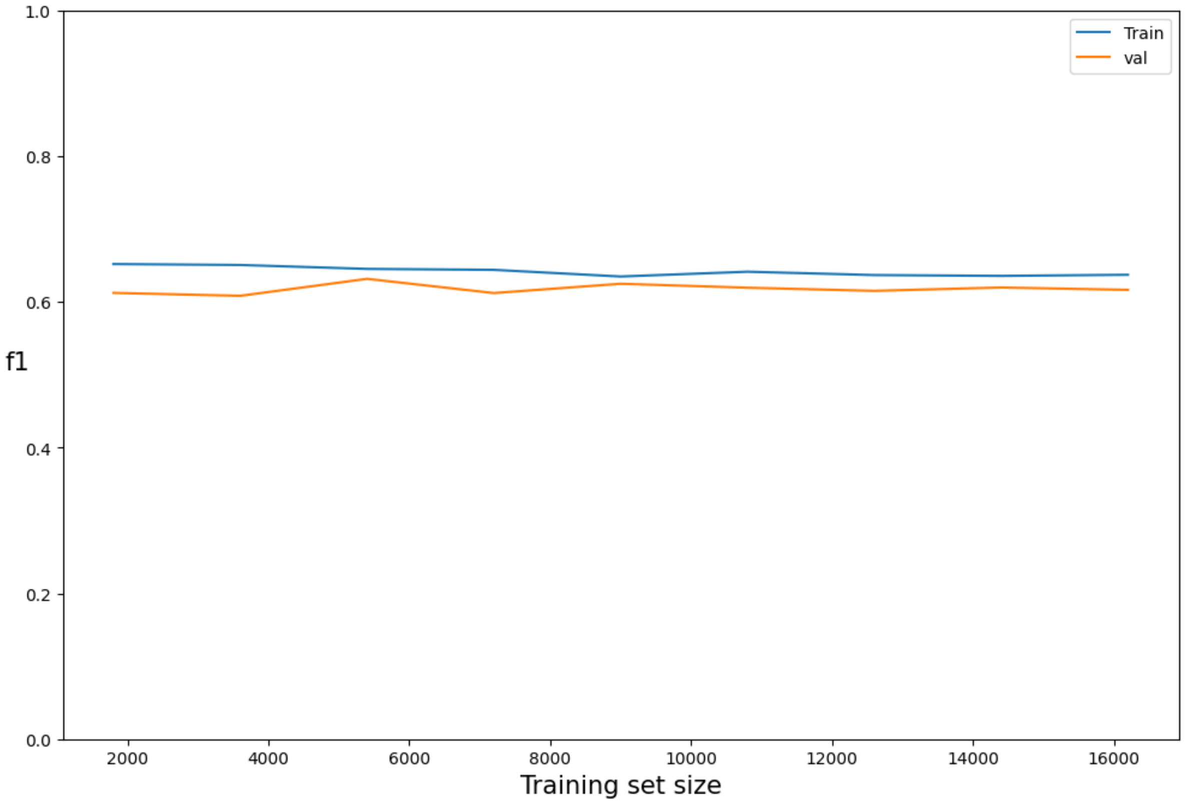
auc for the first class 0.748
auc for the second class 0.748



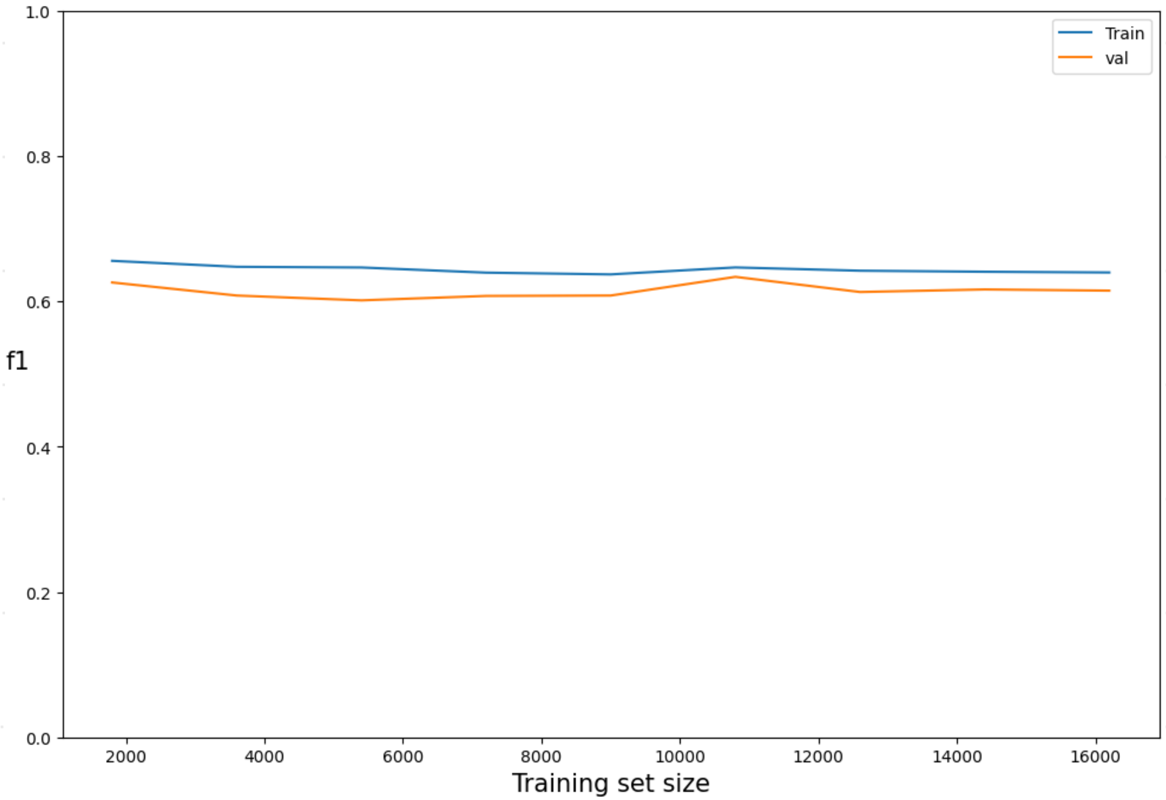


Learning Curves

MultinomialNB learning curve (f1-training size)



Logistic Regression learning curve (f1-training_size)



Hyperparameter Optimization
Multinomial Naïve Bayes

- Create Stratified 10-Fold
- Tune hyperparameters
- gs_clf = GridSearchCV(mnb_classifier, parameters, cv=kf)

```
kf = StratifiedKFold(n_splits=10,shuffle=True,random_state=10)
parameters = {
        "alpha":  [0.00001 ,0.1,0.1180,0.11899,0.12,0.1200009,0.125,0.13,0.15,0.2, 1, 10],
        "fit_prior": [True, False],
        'force_alpha':[True, False]}

gs_clf = GridSearchCV(mnb_classifier, parameters, cv=kf)
```

MultinomialNB best parameters:
{'alpha': 0.12, 'fit_prior': False, 'force_alpha': True}

Hyperparameter Optimization

Multinomial Naïve Bayes

Accuracy Score: 0.891

	precision	recall	f1-score	support
0	0.90	0.87	0.89	974
1	0.88	0.91	0.90	1026
accuracy			0.89	2000
macro avg	0.89	0.89	0.89	2000
weighted avg	0.89	0.89	0.89	2000

1% improvement

Conclusions

- Robust models, high accuracy [89.1%, dataset 20.000 labeled tweets]
- Data analysis proved effective in understanding the dataset
- No overfitting or underfit(learning curve- training, validation loss)
- Cross-validation verifies performance(underestimated accuracy)
- Confusion matrix shows low false positive and false negative rate
- ROC-curves depict the accuracy for each specific class(determined by the covered area)
- Tagging the sentiment can be highly subjective, influenced by personal experience, irony
- Stacking classifiers did not provide any improvements in predictions
- Further improve performance with feature engineering(subjectivity, polarity) ?