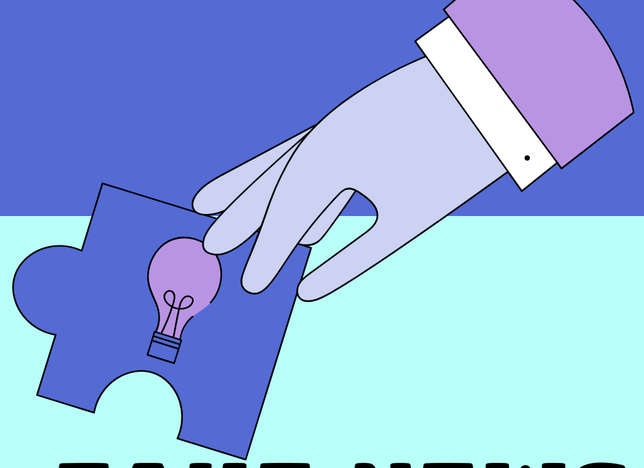
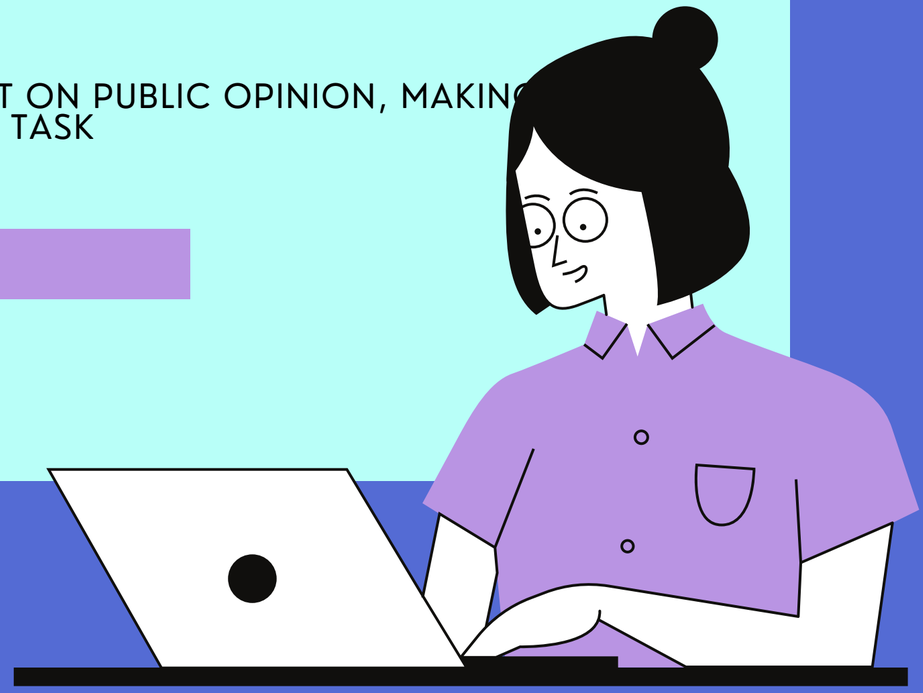


FAKE NEWS USING NATURAL LANGUAGE PROCESSING (NLP) TECHNIQUES

FAKE NEWS HAS BECOME A SIGNIFICANT ISSUE DUE TO ITS IMPACT ON PUBLIC OPINION, MAKING AUTOMATIC DETECTION AN IMPORTANT TASK

HELENA GOMEZ



INTRODUCTION

The objective of this project is to build a machine learning model capable of distinguishing real news from fake news using Natural Language Processing (NLP) techniques. Fake news has become a significant issue due to its impact on public opinion, making automatic detection an important task.

This project explores classical NLP approaches combined with supervised machine learning models to address this problem

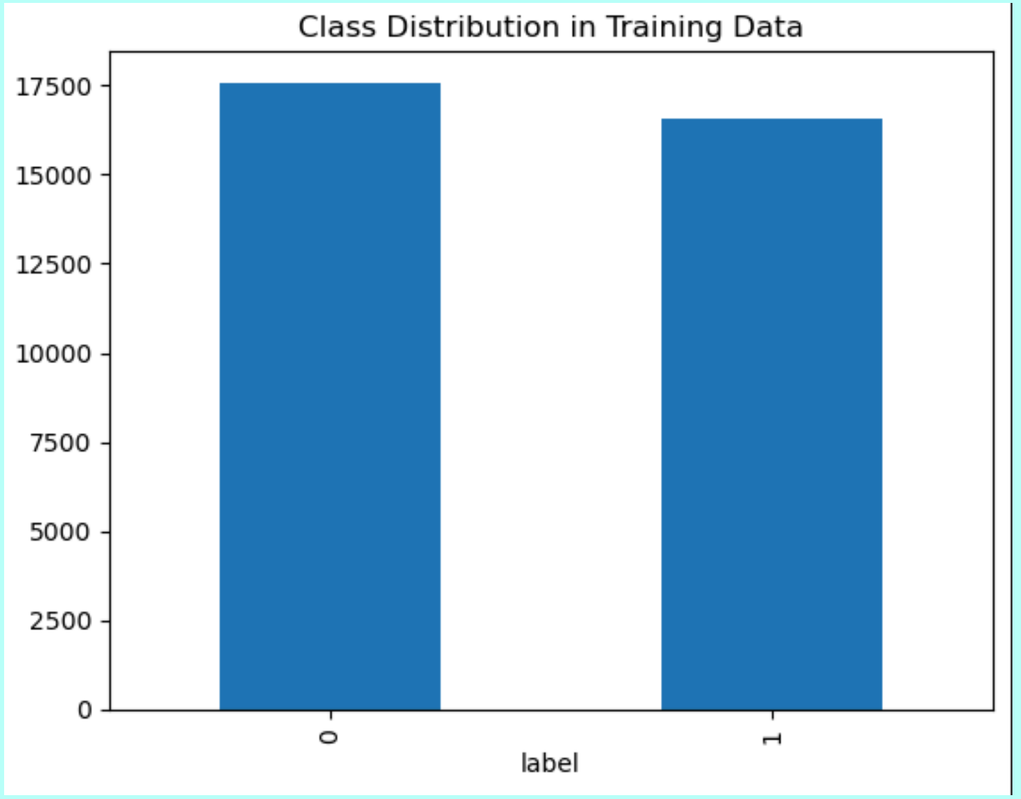
DATASET OVERVIEW

The dataset consists of news articles labeled as real (0) or fake (1).

Exploratory Data Analysis

- 34,152 news articles
- Clean dataset (no missing values)
- Two columns: text and label
- Balanced class distribution
  - Real: 17,572
  - Fake: 16,580

Conclusion:
The dataset is clean, well structured, and suitable for text classification.



	Feature	Value
0	Number of rows	34152
1	Number of columns	2
2	Columns	label, text
3	Missing values	0
4	Real news (label = 0)	17572
5	Fake news (label = 1)	16580

DATA PROCESSING

Before modeling, text data was preprocessed to reduce noise and standardize the input.
The following steps were applied:

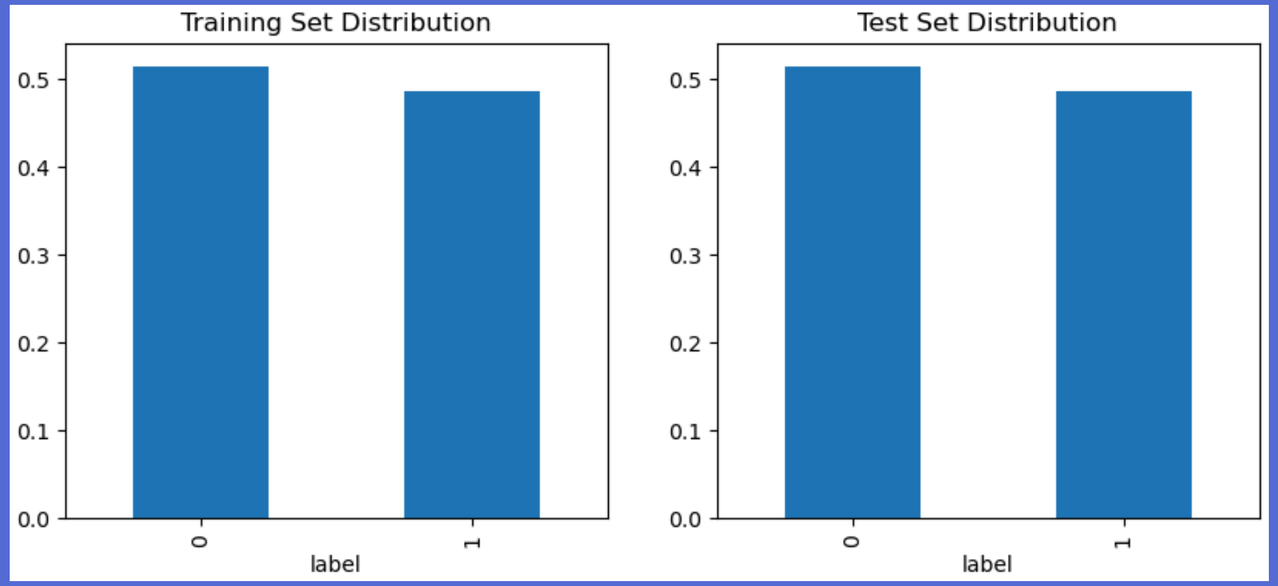
- Conversion to lowercase
- Removal of punctuation and numerical characters
- Tokenization
- Removal of stopwords using NLTK

This preprocessing step ensures that the models focus on meaningful textual patterns rather than irrelevant variations.

TRAINING DATASET

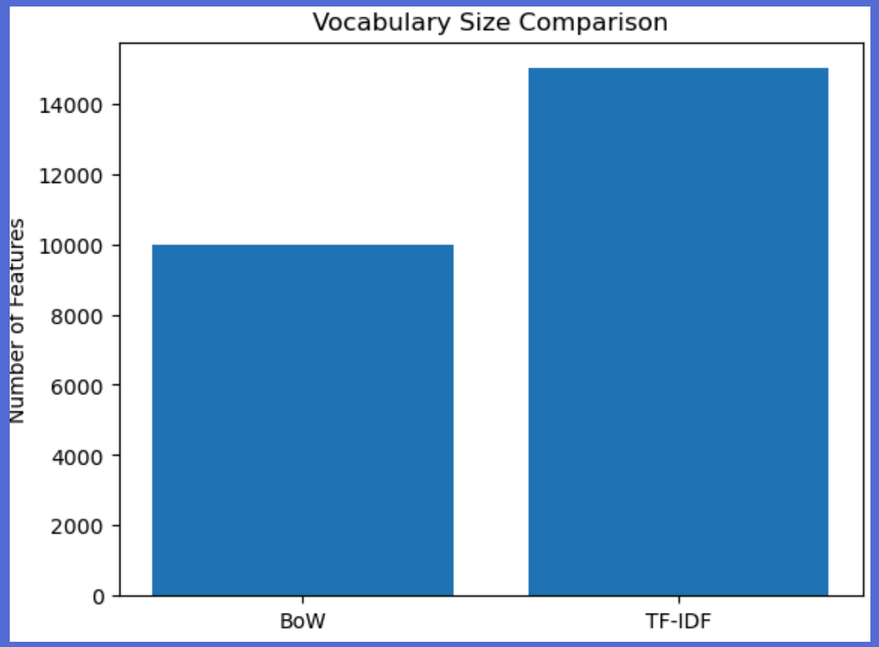
DIVIDING TRAINING DATASET INTO TRAIN AND TEST

- 80% training set / 20% test set
- Training samples: 27,321
- Test samples: 6,831
- Stratified split applied



TEXT VECTORIZATION

Text data was transformed into numerical vectors using TF-IDF and Bag of Words.
This representation allows machine learning models to process textual data.

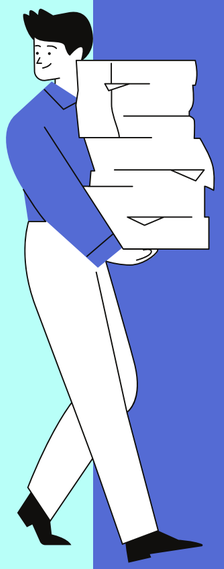


MODELS TESTED

All classifiers were evaluated using two different text representations:
Bag of Words (BoW) and TF-IDF.
The same training and test splits were used to ensure a fair comparison.

Model Training: Accuracy Comparison

Representation	Model	Accuracy
TF-IDF	Linear SVM	**0.9448**
BoW	Logistic Regression	**0.9403**
TF-IDF	Logistic Regression	**0.9376**
TF-IDF	Multinomial Naive Bayes	**0.9362**
BoW	Multinomial Naive Bayes	**0.9338**
BoW	Linear SVM	**0.9332**
TF-IDF	Random Forest	**0.9157**
BoW	Random Forest	**0.9123**



Models evaluated:

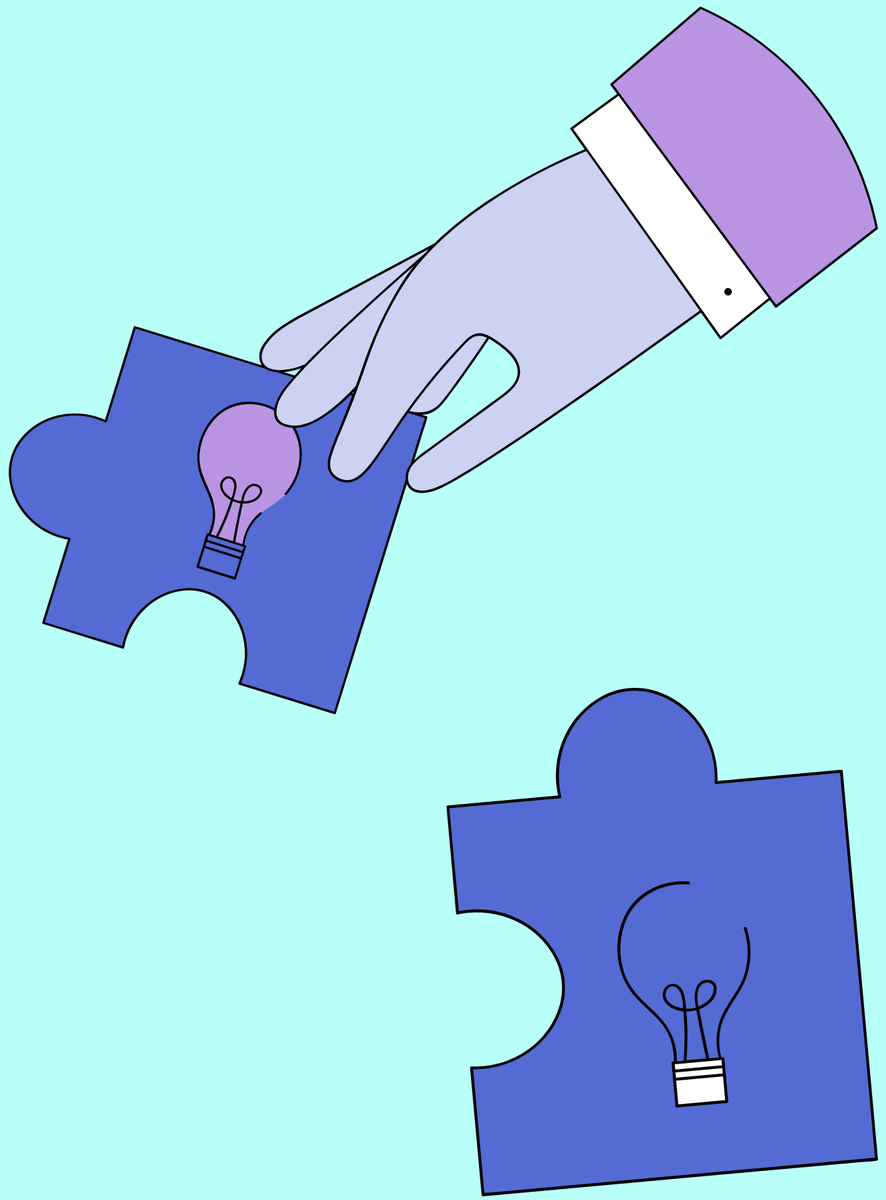
- Logistic Regression
- Linear SVM
- Multinomial Naive Bayes
- Random Forest
- XGBoost (0.8872785829307569)

MODEL COMPARISON

- Linear SVM performs better with TF-IDF than with BoW
- Logistic Regression achieves competitive performance with both representations

- Random Forest shows lower performance compared to linear models

- The worst model was XGBoost



HYPERPARAMETER TUNING

Hyperparameter tuning was applied only to the best-performing models identified during the baseline comparison.

Linear models achieved the highest accuracy and were therefore selected for further optimization

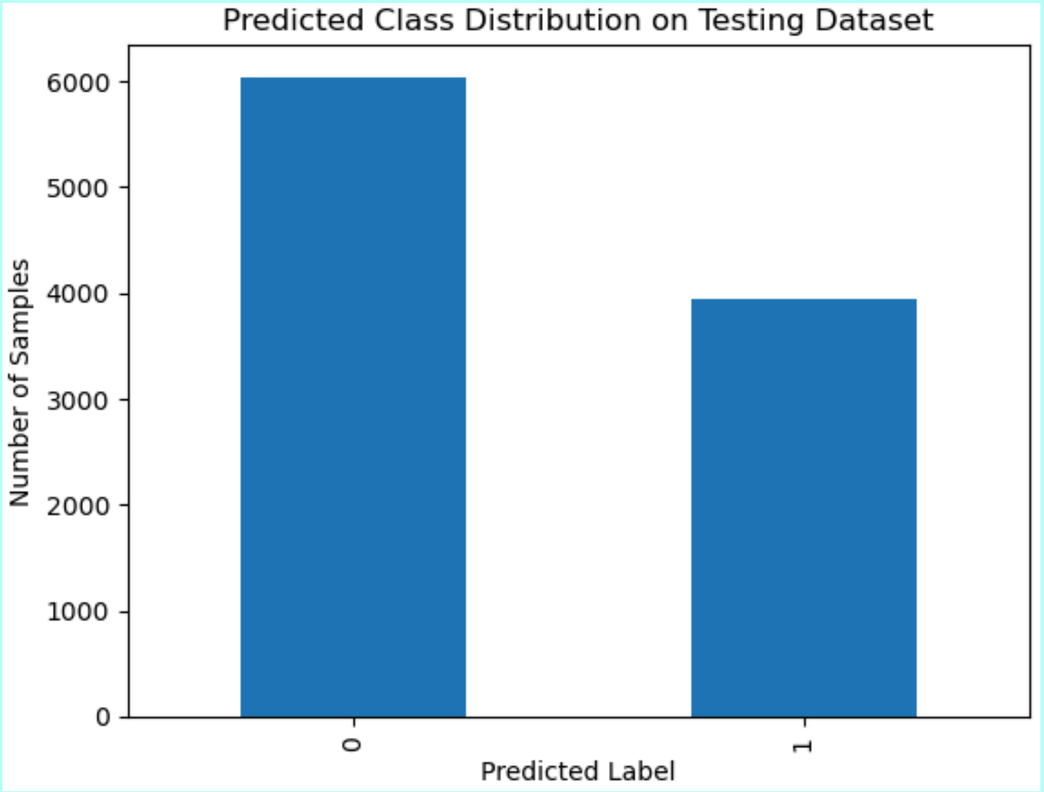
Hyperparameter Tuning Summary

Model	Hyperparameter	Values Tested	Reason
Linear SVM	C	0.01, 0.1, 1, 10	Controls regularization strength
Logistic Regression	C	0.01, 0.1, 1, 10	Balances bias vs variance
Multinomial NB	alpha	0.01, 0.1, 0.5, 1.0	Controls smoothing
Linear SVM	loss	hinge, squared_hinge	Different margin formulations

FINAL EVALUATION

- After hyperparameter tuning, all models achieved strong performance.
- Linear SVM obtained the highest cross-validation accuracy, slightly outperforming
- Logistic Regression and Naive Bayes.

Model	Best Hyperparameters	CV Accuracy
Logistic Regression	C = 10	0.9389
Linear SVM	**C = 1, loss = squared_hinge**	**0.9392**
Multinomial NB	alpha = 0.1	0.9331



TESTING DATASET & PREDICTIONS

The final tuned Linear SVM model was used to generate predictions for the
testing dataset, which contains unseen news articles without labels.

The placeholder value (2) was replaced with the predicted classes (0 or 1), producing the final submission file testing_with_predictions.csv.

	label	text
0	2	copycat muslim terrorist arrested with assault...
1	2	wow! chicago protester caught on camera admits...
2	2	germany's fdp look to fill schaeuble's big shoes
3	2	mi school sends welcome back packet warning ki...
4	2	u.n. seeks 'massive' aid boost amid rohingya '...

	label	text
0	0	copycat muslim terrorist arrested with assault...
1	0	wow! chicago protester caught on camera admits...
2	1	germany's fdp look to fill schaeuble's big shoes
3	0	mi school sends welcome back packet warning ki...
4	1	u.n. seeks 'massive' aid boost amid rohingya '...

CONCLUSION

- The dataset contained 34,152 news articles with a balanced class distribution
- (≈51% real news and ≈49% fake news), allowing fair model evaluation.
- TF-IDF consistently outperformed Bag of Words across most classifiers.
- The best-performing model was TF-IDF combined with Linear SVM, achieving a
- cross-validation accuracy of 93.92%.
- After hyperparameter tuning, the Linear SVM achieved a test accuracy of 94%,
- with balanced precision, recall, and F1-score (≈0.94–0.95) for both classes.
- The final tuned model was successfully applied to the testing dataset,
- generating the required predictions for submission.

REFINEMENT

Focus on sentence-level details like grammar, punctuation, and word choice.

FINAL DRAFT

Prepare the final draft by incorporating all revisions and refinements.

THANK YOUUU!!!!