Project Report on MBTI Personality Prediction

Introduction to the Problem

The Myers-Briggs Type Indicator (MBTI) is a widely-used personality assessment tool that categorizes individuals into one of 16 distinct personality types based on four dichotomies. Understanding an individual's MBTI type can be beneficial in various domains such as career counseling, team building, personal development, and interpersonal relationships. Automating the prediction of MBTI types from textual data, such as social media posts or personal writings, can offer scalable insights into user personalities without the need for extensive surveys.

Our Approach

Data Preprocessing

To prepare the text data for modeling, we employed several preprocessing steps to clean and normalize the data. The `clear_text` function implemented these steps:

Lowercasing: Standardizing the text to lowercase to ensure uniformity.
URL Removal: Stripping away hyperlinks that do not contribute to personality type inference.
Symbol Removal: Cleaning non-alphanumeric characters to focus on textual content.

Handling Data Imbalance

Given the potential class imbalance in the dataset, we opted to use the `class_weight='balanced'` parameter in our models. This approach helps to ensure that each class is given appropriate importance during training, despite differences in their frequency. Additionally, during the train-test split, we utilized `stratify=y` to maintain a consistent distribution of MBTI types in both training and testing datasets, enhancing the reliability of our model evaluations.

Machine Learning Models

We experimented with several machine learning models known for their effectiveness in text classification tasks:

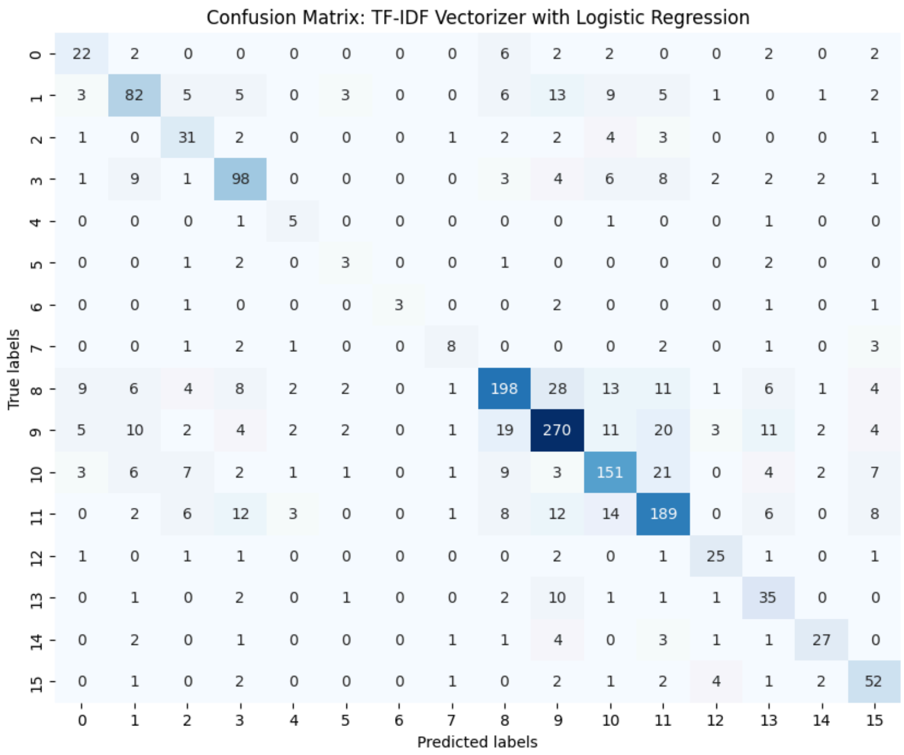
- **Logistic Regression**
- **Support Vector Machine (SVM)**
- **Multinomial Naive Bayes**
- **Random Forest**
- **BERT**

These models were integrated into pipelines with two types of text vectorizers: TF-IDF and Count Vectorizer. This approach allowed us to convert raw text into a format suitable for model training while evaluating the impact of different text representation techniques.

Model Evaluation and Selection

We conducted 5-fold cross-validation for each model and pipeline combination to robustly assess their performance. The evaluation metrics included average accuracy, F1-score, recall, precision, and a confusion matrix. The best-performing combinations were:

- **TF-IDF Vectorizer with Logistic Regression**:
    - ☐ Average Accuracy: 0.691
    - ☐ Average Precision: 0.697
    - ☐ Average Recall: 0.691
    - ☐ Average F1 Score: 0.691



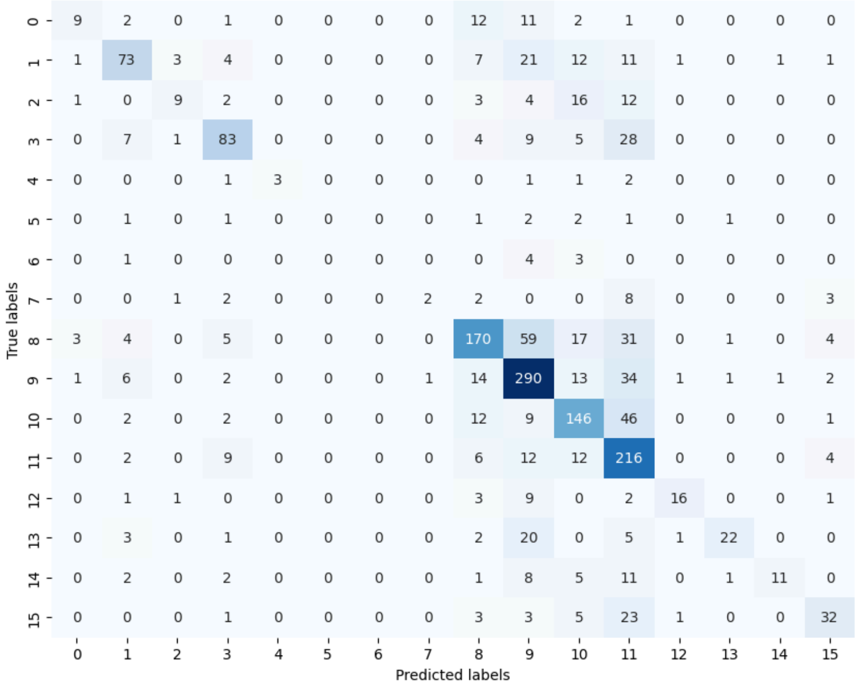
Confusion Matrix: TF-IDF Vectorizer with Logistic Regression

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	22	2	0	0	0	0	0	0	6	2	2	0	0	2	0	2
1	3	82	5	5	0	3	0	0	6	13	9	5	1	0	1	2
2	1	0	31	2	0	0	0	1	2	2	4	3	0	0	0	1
3	1	9	1	98	0	0	0	0	3	4	6	8	2	2	2	1
4	0	0	0	1	5	0	0	0	0	0	1	0	0	1	0	0
5	0	0	1	2	0	3	0	0	1	0	0	0	0	2	0	0
6	0	0	1	0	0	0	3	0	0	2	0	0	0	1	0	1
7	0	0	1	2	1	0	0	8	0	0	0	2	0	1	0	3
8	9	6	4	8	2	2	0	1	198	28	13	11	1	6	1	4
9	5	10	2	4	2	2	0	1	19	270	11	20	3	11	2	4
10	3	6	7	2	1	1	0	1	9	3	151	21	0	4	2	7
11	0	2	6	12	3	0	0	1	8	12	14	189	0	6	0	8
12	1	0	1	1	0	0	0	0	0	2	0	1	25	1	0	1
13	0	1	0	2	0	1	0	0	2	10	1	1	1	35	0	0
14	0	2	0	1	0	0	0	1	1	4	0	3	1	1	27	0
15	0	1	0	2	0	0	0	1	0	2	1	2	4	1	2	52

- **TF-IDF Vectorizer with SVM**:
    - ☐ Average Accuracy: 0.6280115273775215
    - ☐ Average Precision: 0.6463181833007485
    - ☐ Average Recall: 0.6280115273775215

- ☐ Average F1 Score: 0.6138485255336215

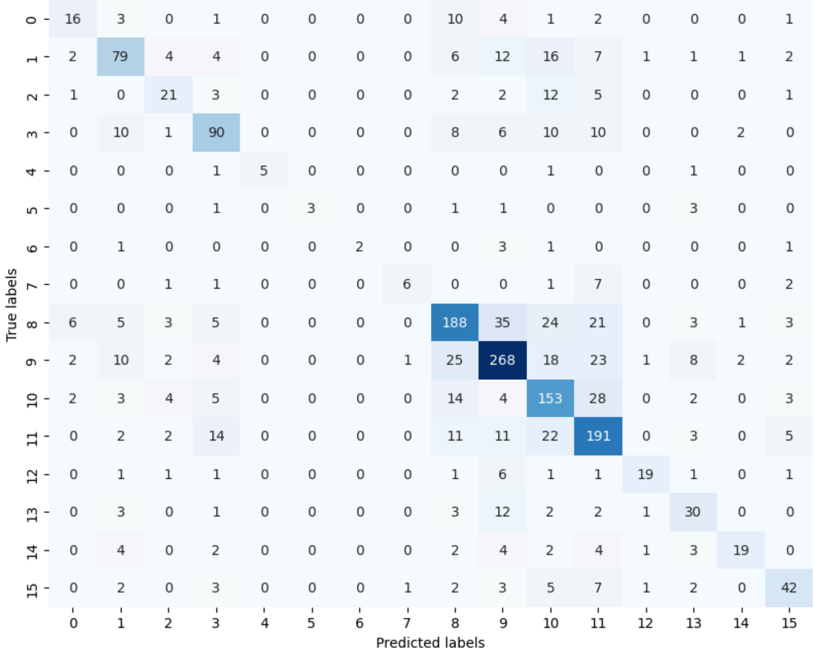
Confusion Matrix: TF-IDF Vectorizer with Support Vector Machine



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	9	2	0	1	0	0	0	0	12	11	2	1	0	0	0	0
1	1	73	3	4	0	0	0	0	7	21	12	11	1	0	1	1
2	1	0	9	2	0	0	0	0	3	4	16	12	0	0	0	0
3	0	7	1	83	0	0	0	0	4	9	5	28	0	0	0	0
4	0	0	0	1	3	0	0	0	0	1	1	2	0	0	0	0
5	0	1	0	1	0	0	0	0	1	2	2	1	0	1	0	0
6	0	1	0	0	0	0	0	0	0	4	3	0	0	0	0	0
7	0	0	1	2	0	0	0	2	2	0	0	8	0	0	0	3
8	3	4	0	5	0	0	0	0	170	59	17	31	0	1	0	4
9	1	6	0	2	0	0	0	1	14	290	13	34	1	1	1	2
10	0	2	0	2	0	0	0	0	12	9	146	46	0	0	0	1
11	0	2	0	9	0	0	0	0	6	12	12	216	0	0	0	4
12	0	1	1	0	0	0	0	0	3	9	0	2	16	0	0	1
13	0	3	0	1	0	0	0	0	2	20	0	5	1	22	0	0
14	0	2	0	2	0	0	0	0	1	8	5	11	0	1	11	0
15	0	0	0	1	0	0	0	0	3	3	5	23	1	0	0	32

- **Count Vectorizer with SVM**:
  - ☐ Average Accuracy: 0.657521613832853
  - ☐ Average Precision: 0.6608670500584127
  - ☐ Average Recall: 0.657521613832853
  - ☐ Average F1 Score: 0.652151263402881

Confusion Matrix: Count Vectorizer with Support Vector Machine



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	16	3	0	1	0	0	0	0	10	4	1	2	0	0	0	1
1	2	79	4	4	0	0	0	0	6	12	16	7	1	1	1	2
2	1	0	21	3	0	0	0	0	2	2	12	5	0	0	0	1
3	0	10	1	90	0	0	0	0	8	6	10	10	0	0	2	0
4	0	0	0	1	5	0	0	0	0	0	1	0	0	1	0	0
5	0	0	0	1	0	3	0	0	1	1	0	0	0	3	0	0
6	0	1	0	0	0	0	2	0	0	3	1	0	0	0	0	1
7	0	0	1	1	0	0	0	6	0	0	1	7	0	0	0	2
8	6	5	3	5	0	0	0	0	188	35	24	21	0	3	1	3
9	2	10	2	4	0	0	0	1	25	268	18	23	1	8	2	2
10	2	3	4	5	0	0	0	0	14	4	153	28	0	2	0	3
11	0	2	2	14	0	0	0	0	11	11	22	191	0	3	0	5
12	0	1	1	1	0	0	0	0	1	6	1	1	19	1	0	1
13	0	3	0	1	0	0	0	0	3	12	2	2	1	30	0	0
14	0	4	0	2	0	0	0	0	2	4	2	4	1	3	19	0
15	0	2	0	3	0	0	0	1	2	3	5	7	1	2	0	42

Encoding of MBTI Personality Types

To facilitate the machine learning processes, we have converted the Myers-Briggs Type Indicator (MBTI) personality types from textual representations to numerical encodings using LabelEncoder. This transformation is crucial for the effective application of algorithms that require numerical input. Below is the mapping used in our analysis:

- **0**: ENFJ
- **1**: ENFP
- **2**: ENTJ
- **3**: ENTP
- **4**: ESFJ
- **5**: ESFP
- **6**: ESTJ
- **7**: ESTP
- **8**: INFJ
- **9**: INFP
- **10**: INTJ
- **11**: INTP
- **12**: ISFJ
- **13**: ISFP
- **14**: ISTJ
- **15**: ISTP

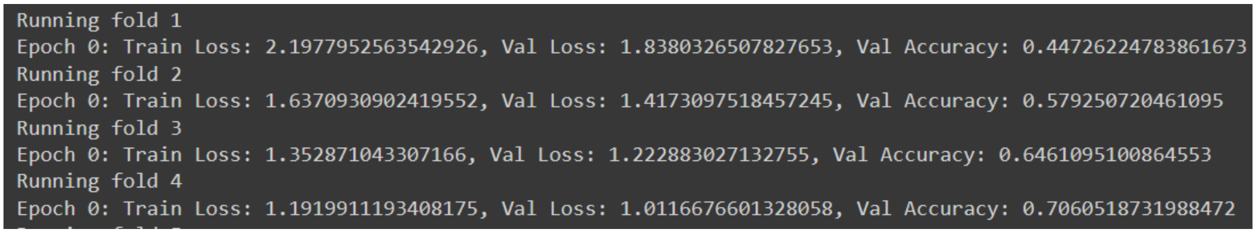
Hyperparameter Tuning

Based on the initial results, we selected the top three model configurations for further optimization through hyperparameter tuning. The tuning process also considered different parameters for the vectorizers to maximize performance. This step is crucial for refining our models to achieve the best possible predictions and is currently ongoing, with results pending at the time of this report’s deadline.

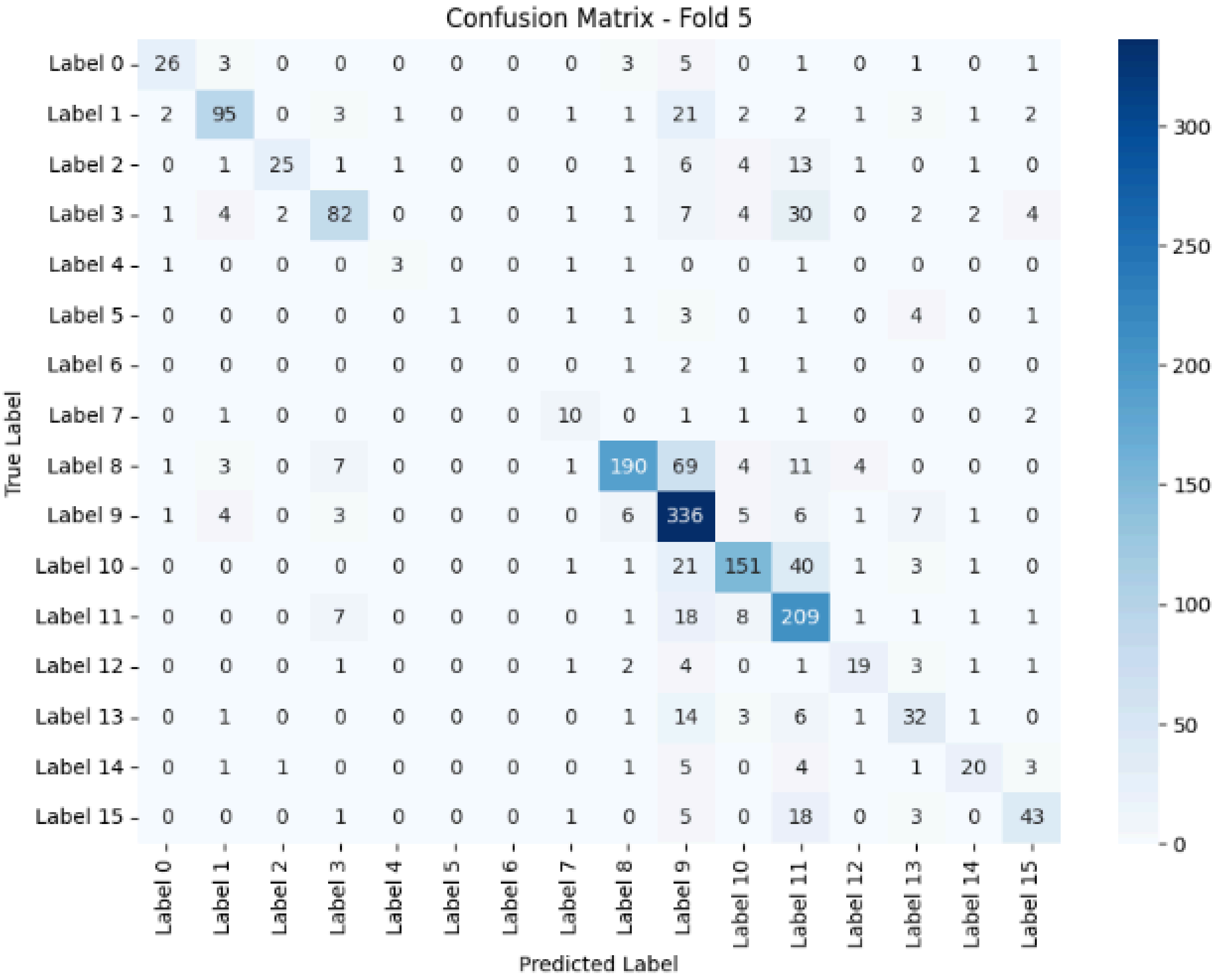
Bert

With BERT, we can arguably see the best accuracy of up to 71% with only 5 epochs thus this offers significantly better performance. However, we observe that labels with very less data are hard to predict.

Final accuracy average:
{'accuracy': [0.715850144092219],
 'precision': [0.7441454649692727],
 'recall': [0.715850144092219],
 'f1': [0.7114172551038348]}



```
Running fold 1
Epoch 0: Train Loss: 2.1977952563542926, Val Loss: 1.8380326507827653, Val Accuracy: 0.44726224783861673
Running fold 2
Epoch 0: Train Loss: 1.6370930902419552, Val Loss: 1.4173097518457245, Val Accuracy: 0.579250720461095
Running fold 3
Epoch 0: Train Loss: 1.352871043307166, Val Loss: 1.222883027132755, Val Accuracy: 0.6461095100864553
Running fold 4
Epoch 0: Train Loss: 1.1919911193408175, Val Loss: 1.0116676601328058, Val Accuracy: 0.7060518731988472
```


Confusion Matrix - Fold 5

	Label 0	Label 1	Label 2	Label 3	Label 4	Label 5	Label 6	Label 7	Label 8	Label 9	Label 10	Label 11	Label 12	Label 13	Label 14	Label 15
Label 0	26	3	0	0	0	0	0	0	3	5	0	1	0	1	0	1
Label 1	2	95	0	3	1	0	0	1	1	21	2	2	1	3	1	2
Label 2	0	1	25	1	1	0	0	0	1	6	4	13	1	0	1	0
Label 3	1	4	2	82	0	0	0	1	1	7	4	30	0	2	2	4
Label 4	1	0	0	0	3	0	0	1	1	0	0	1	0	0	0	0
Label 5	0	0	0	0	0	1	0	1	1	3	0	1	0	4	0	1
Label 6	0	0	0	0	0	0	0	0	1	2	1	1	0	0	0	0
Label 7	0	1	0	0	0	0	0	10	0	1	1	1	0	0	0	2
Label 8	1	3	0	7	0	0	0	1	190	69	4	11	4	0	0	0
Label 9	1	4	0	3	0	0	0	0	6	336	5	6	1	7	1	0
Label 10	0	0	0	0	0	0	0	1	1	21	151	40	1	3	1	0
Label 11	0	0	0	7	0	0	0	0	1	18	8	209	1	1	1	1
Label 12	0	0	0	1	0	0	0	1	2	4	0	1	19	3	1	1
Label 13	0	1	0	0	0	0	0	0	1	14	3	6	1	32	1	0
Label 14	0	1	1	0	0	0	0	0	1	5	0	4	1	1	20	3
Label 15	0	0	0	1	0	0	0	1	0	5	0	18	0	3	0	43

Label Mappings:

```
Label mappings:
0: ENFJ
1: ENFP
2: ENTJ
3: ENTP
4: ESFJ
5: ESFP
6: ESTJ
7: ESTP
```

```
8: INFJ
9: INFP
10: INTJ
11: INTP
12: ISFJ
13: ISFP
14: ISTJ
15: ISTP
```

Conclusion

This project aims to demonstrate the feasibility of predicting MBTI personality types using machine learning techniques applied to text data. Through systematic preprocessing, model evaluation, and hyperparameter tuning, we strive to develop a robust predictive system. Future work will focus on completing the tuning process and potentially deploying the model in a real-world application to provide automated personality assessments.