Understanding its power and influence

Categorizing Severity in Large-Scale Customer Complaints with NLP

Jui-Li Chen, Yu-Jou Chu, Szu-Yu Chen

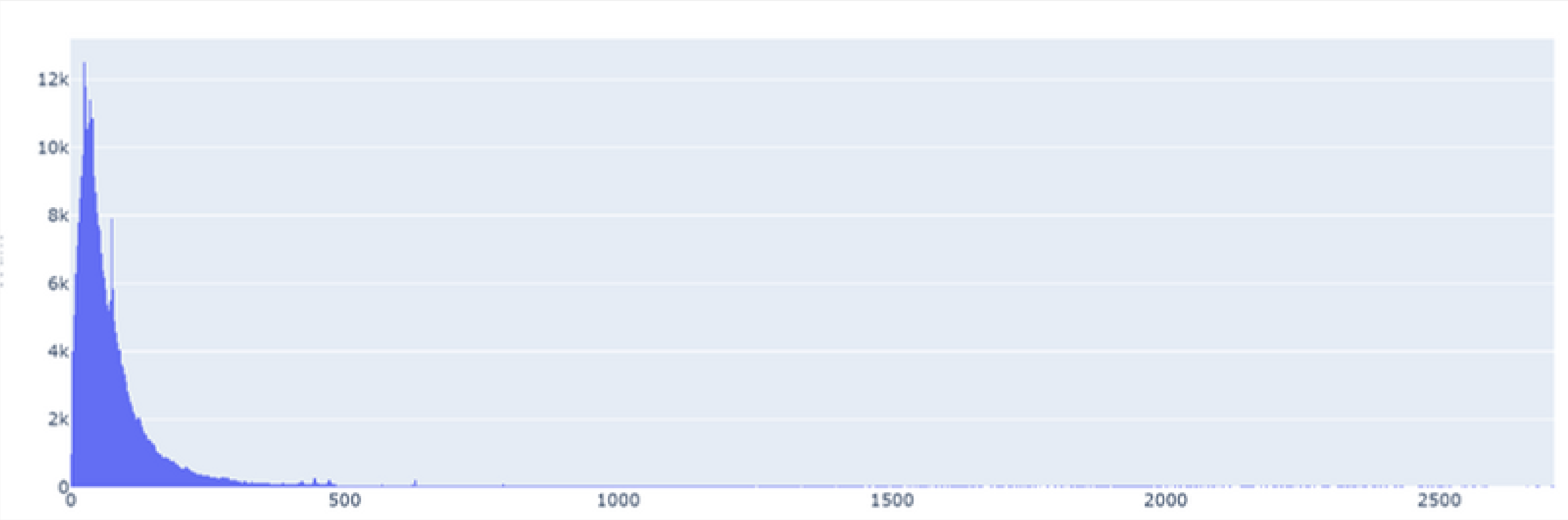
Customer Complaints Severity Prediction

- Data Exploration
- Data Preparation
- Model Creation
- Model Evaluation and Modification
- Conclusion

Complaint dataset contains information aimed at improving CFPB

we start to explore this dataset with univariate analysis

- Complaints on financial product and service made by consumers
- Complied by the Consumer Financial Protection Bureau (CFPB)
- There are 17 variables
- The shortest complaint is under 10 words and the longest one is much over than 2000




```
df_novec.nunique()
✓ 4.1s
```

17 variables

Date received	1782
Product	14
Sub-product	56
Issue	88
Sub-issue	198
Consumer complaint narrative	863443
Company public response	10
Company	4767
State	61
ZIP code	6931
Tags	3
Consumer consent provided?	1
Submitted via	1
Date sent to company	1787
Company response to consumer	5
Timely response?	2
Consumer disputed?	0
Complaint ID	863443

Discerning complaints severity provides advantage for CFPB

Project goal: Ensure consumer access to fair, transparent, and competitive financial products and services

Proactive Problem Solving → **Prioritized Response** → **Enhanced the Reputation**

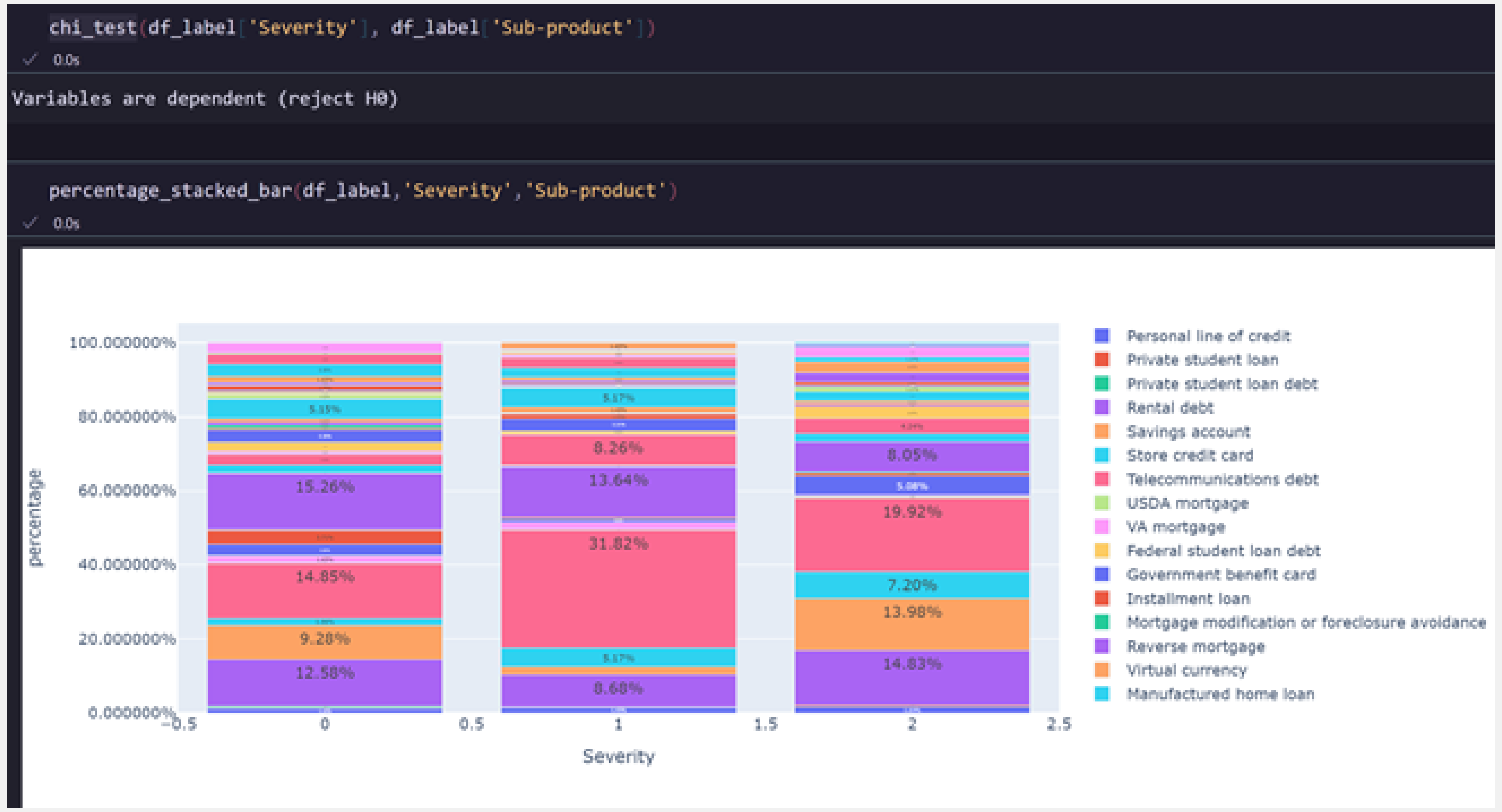
Severity labels are determined by our assessment of what the company considers important.

Level 1	Level 2	Level 3
Financial issue	Fraud issue, Information leakage	Life-changing

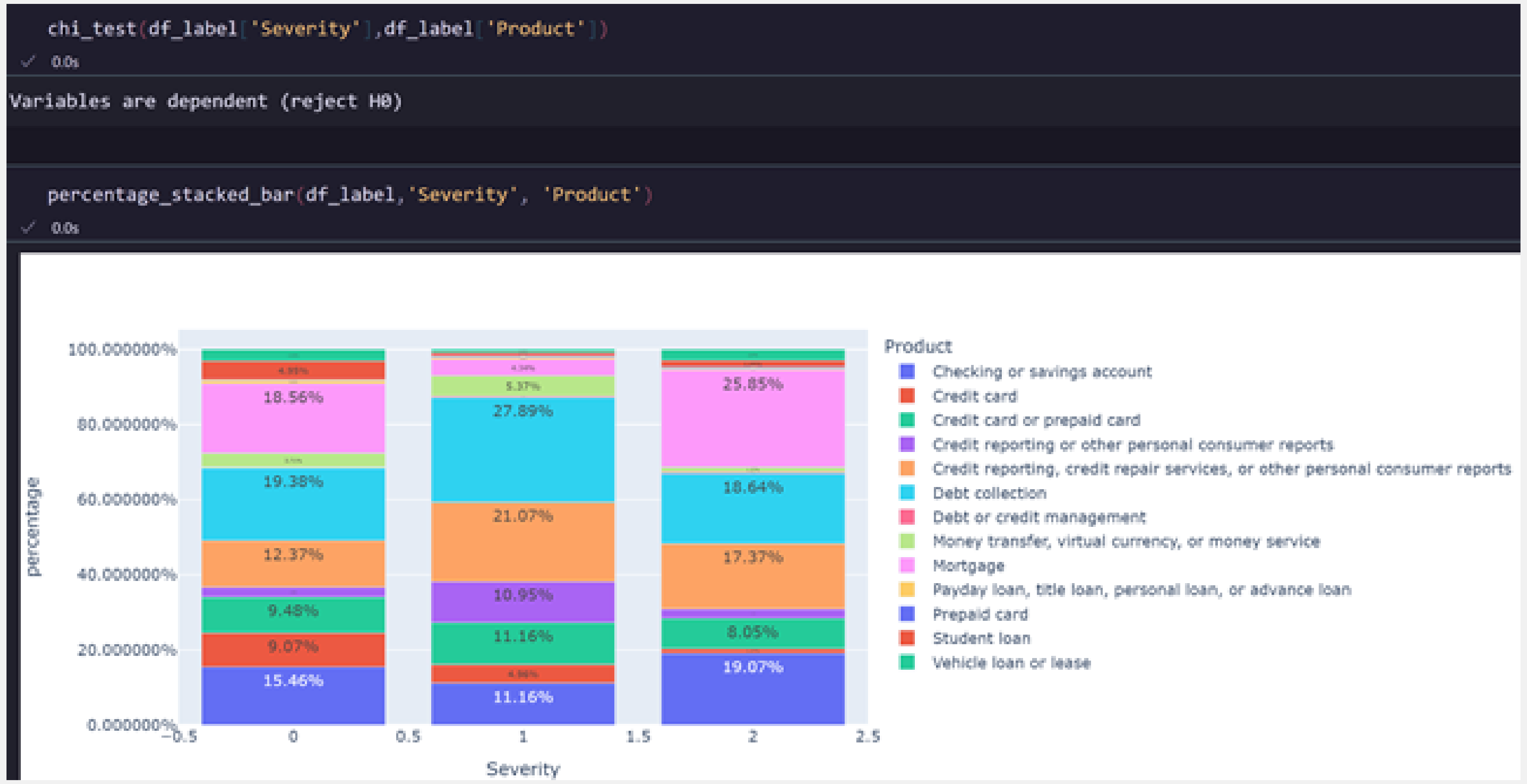
Certain variables influence the complaint severity

Doing **bivariate analysis** to investigate the relationship between severity and other variables

sub-product vs severity



product vs severity



Issue vs severity

Certain variables influence the complaint severity

Doing bivariate analysis to investigate the relationship between severity and other variables

sub-issue vs severity

```
chi_test(df_label['Severity'], df_label['Sub-issue'])
```

✓ 0.0s

```
Variables are dependent (reject H0)
```

```
percentage_stacked_bar(df_label,'Severity','Sub-issue')
```

✓ 0.2s

company vs severity

```
chi_test(df_label['Severity'], df_label['Company'])
```

✓ 0.0s

```
Variables are dependent (reject H0)
```

```
percentage_stacked_bar(df_label,'Severity','Company')
```

✓ 0.3s

Company responses to consumer vs severity

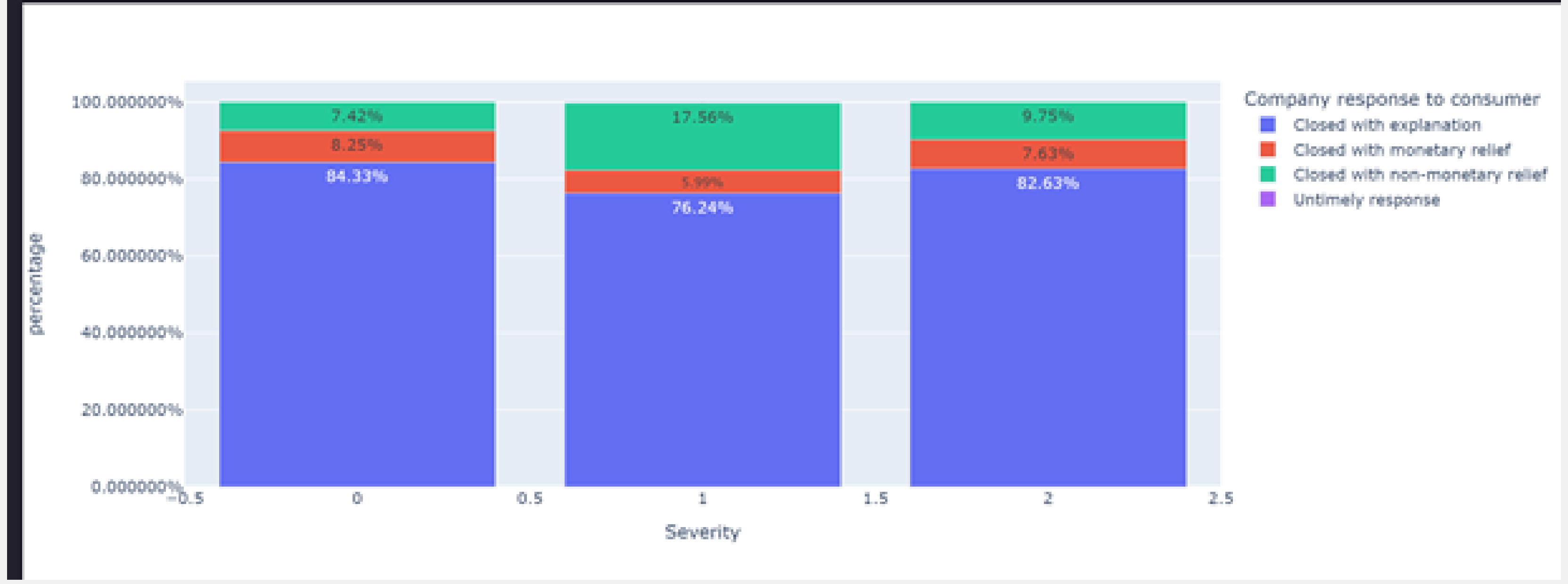
```
chi_test(df_label['Severity'], df_label['Company response to consumer'])
```

✓ 0.0s

```
Variables are dependent (reject H0)
```

```
percentage_stacked_bar(df_label,'Severity','Company response to consumer')
```

✓ 0.0s



Get Clean data for modeling
Data Preparation

We went through techniques to decrease the noise for models

- Remove punctuations
- Remove numbers
- Tokenize
- Set lowercase
- Remove < 10 words
- Remove ‘XXXX’

It is privacy information in the complaints

Try different models

In this project, we examine 5 models to find the best one for our project. Then, we apply these models to the two Feature Engineering methods.

Experiment on Word2Vec

Five tables to show the results of our experiment on Word2Vec with five different models

LogisticRegression	
Mean cross-validation score(cv = 10)	0.61
Standard deviation of cross-validation scores	0.03
Validation Set Accuracy	0.63
Test Set Accuracy	0.63

RandomForest	
Mean cross-validation score(cv = 10)	0.55
Standard deviation of cross-validation scores	0.05
Validation Set Accuracy	0.65
Test Set Accuracy	0.49

GradientBoosting	
Mean cross-validation score(cv = 10)	0.57
Standard deviation of cross-validation scores	0.04
Validation Set Accuracy	0.65
Test Set Accuracy	0.57

NeuralNetwork	
Mean cross-validation score(cv = 10)	0.61
Standard deviation of cross-validation scores	0.03
Validation Set Accuracy	0.62
Test Set Accuracy	0.58

XGBoost	
Mean cross-validation score(cv = 10)	0.56
Standard deviation of cross-validation scores	0.04
Validation Set Accuracy	0.64
Test Set Accuracy	0.53

Try different models

In this project, we examine 5 models to find the best one for our project. Then, we apply these models to the two Feature Engineering methods.

Experiment on TF-IDF

Five tables to show the results of our experiment on TF-IDF with five different models

LogisticRegression	
Mean cross-validation score(cv = 10)	0.67
Standard deviation of cross-validation scores	0.04
Validation Set Accuracy	0.63
Test Set Accuracy	0.7

RandomForest	
Mean cross-validation score(cv = 10)	0.67
Standard deviation of cross-validation scores	0.05
Validation Set Accuracy	0.7
Test Set Accuracy	0.73

GradientBoosting	
Mean cross-validation score(cv = 10)	0.69
Standard deviation of cross-validation scores	0.04
Validation Set Accuracy	0.73
Test Set Accuracy	0.74

XGBoost	
Mean cross-validation score(cv = 10)	0.7
Standard deviation of cross-validation scores	0.04
Validation Set Accuracy	0.74
Test Set Accuracy	0.76

NeuralNetwork	
Mean cross-validation score(cv = 10)	0.64
Standard deviation of cross-validation scores	0.03
Validation Set Accuracy	0.65
Test Set Accuracy	0.71

Try different models

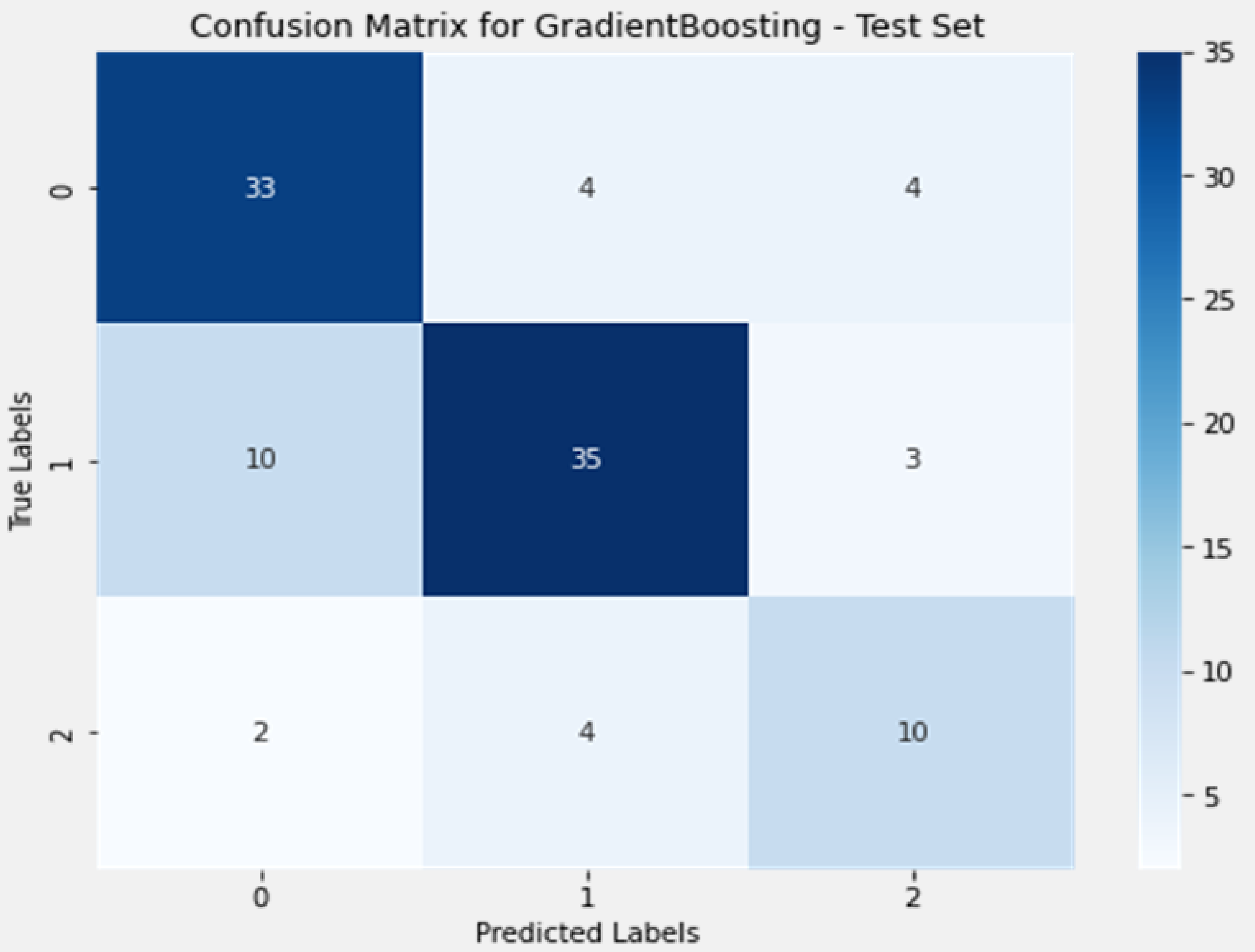
In this project, we examine 5 models to find the best one for our project. Then, we apply these models to the two Feature Engineering methods.

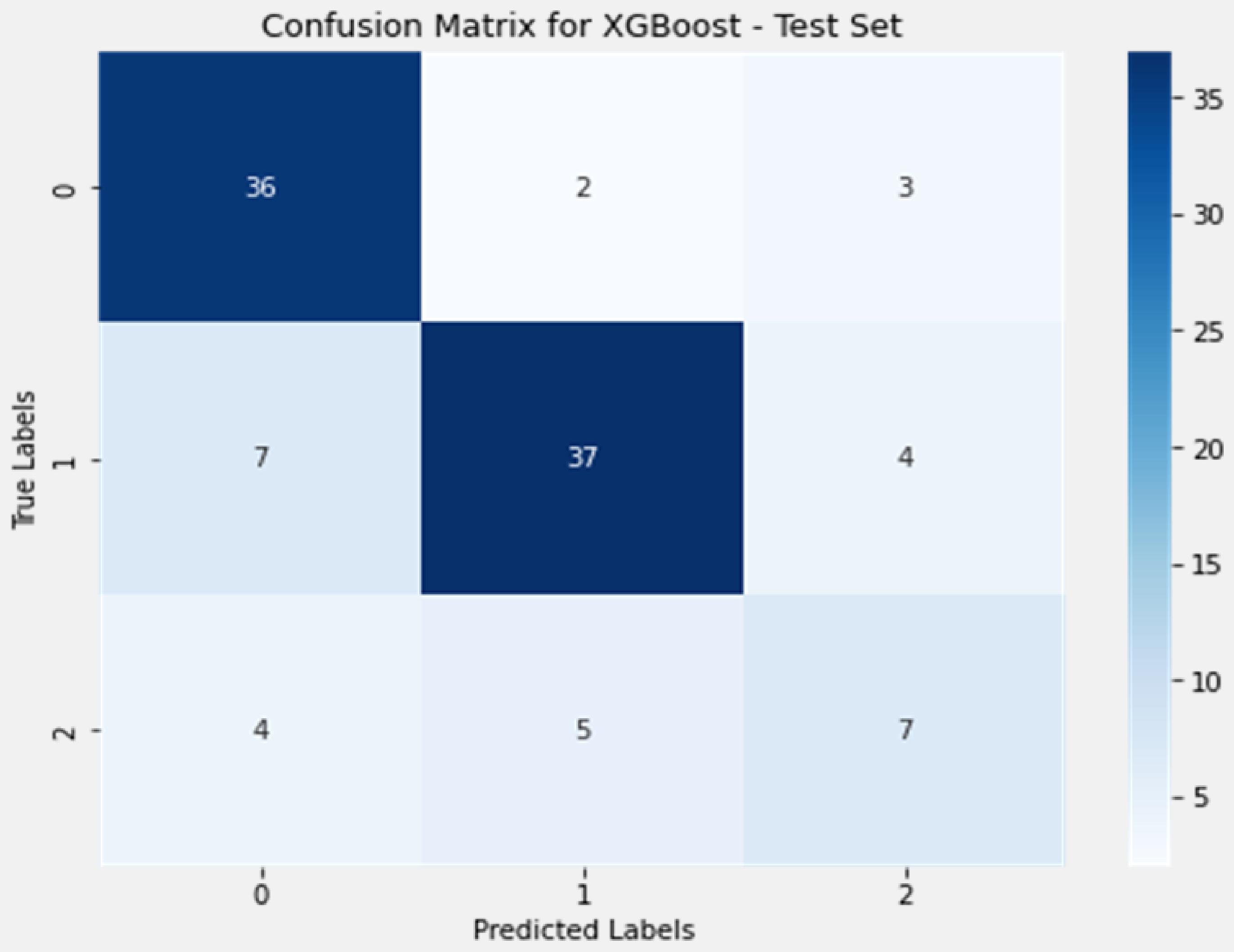
Experiment on TF-IDF

Model selection: why we choose GradientBoosting over XGBoost

GradientBoosting	
Mean cross-validation score(cv = 10)	0.69
Standard deviation of cross-validation scores	0.04
Validation Set Accuracy	0.73
Test Set Accuracy	0.74

XGBoost	
Mean cross-validation score(cv = 10)	0.7
Standard deviation of cross-validation scores	0.04
Validation Set Accuracy	0.74
Test Set Accuracy	0.76





Modify the model to achieve a higher accuracy rate

Before error analysis

- Too many labels and too many noise in the lowest level
- Insufficient sample problem

Accuracy improves with alterations

- Only consider 3 labels rather than 4 labels
- Increase additional 480 labels after modeling

Power Consumption

What will be the next?

- Keep increase labels, increasing accuracy

Thank you!