Malignant_Comments_Classifier

Submitted by:

Shivam Namdev Gadekar

ACKNOWLEDGMENT

For this project, I deep studied models required, and sample projects from Kaggle. Then websites explaining more concepts and revised data trained lectures.

INTRODUCTION

- ## Business Problem Framing

  The proliferation of social media enables people to express their opinions widely online. However, at the same time, this has resulted in the emergence of conflict and hate, making online environments uninviting for users. Although researchers have found that hate is a problem across multiple platforms, there is a lack of models for online hate detection.

  ### Conceptual Background of the Domain Problem

  Online hate, described as abusive language, aggression, cyberbullying, hatefulness and many others has been identified as a major threat on online social media platforms. Social media platforms are the most prominent grounds for such toxic behaviour.

  There has been a remarkable increase in the cases of cyberbullying and trolls on various social media platforms. Many celebrities and influences are facing backlashes from people and have to come across hateful and offensive comments. This can take a toll on anyone and affect them mentally leading to depression, mental illness, self-hatred and suicidal thoughts.

- ## Review of Literature

  Internet comments are bastions of hatred and vitriol. While online anonymity has provided a new outlet for aggression and hate speech, machine learning can be used to fight it. The problem we sought to solve was the tagging of internet comments that are aggressive towards other users. This means that insults to third parties such as celebrities will be tagged as unoffensive, but “u are an idiot” is clearly offensive.

- ## Motivation for the Problem Undertaken

  Social media has ingrained itself into our lives, and it has the same effects on our minds and daily lives as the actual world does. As a

result, just as in the actual world, we must be vigilant in this one as well.

world. Similar to how traditional bullying affects a person's mental health, cyberbullying like as trolling and stalking do as well. Therefore, hateful or unfavourable remarks on social media platforms can have an effect on a person's psychology and even cause sadness or the development of suicidal thoughts in that individual.
Even in the context of enterprises, these hateful remarks can harm the reputation of the brand and sway customers' opinions. Therefore, it is crucial to identify these hateful remarks by categorising.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  Describe the mathematical, statistical and analytics modelling done during this project along with the proper justification.

- Data Sources and their formats

  We have data in CSV format, and later converted into pandas data frame.

- Data Preprocessing Done

  1) Converting comments to lower case

  2) Removing punctuation

  3) Replacing white spaces between space

  4) Removing leading and trailing whitespace

  5) Replacing email address, URLs, numbers, money symbols & 10 digits phone room with a blank space.

  6) Removing stopwords

  7) Applying Lemmatization

- Data Inputs- Logic- Output Relationships
  1) Correlation - Malignant comment is highly correlated with rude comments and abuse comments
- State the set of assumptions (if any) related to the problem under consideration

  Here, you can describe any presumptions taken by you.

- Hardware and Software Requirements and Tools Used

  pandas, numpy, matplotlib.pyplot, seaborn,

  scikit_multilearn and scikit_learn. The laptop used is with Intel I3

  9th generation, 8GB RAM

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

  cleaned the data by removing punctuation, whitespace,

  numbers, emails, URLs, phone number and stop words. Then changed the comments into vector form using TF DIF vectorizer. transformed the target variables from multi-label to multi class target

- Testing of Identified Approaches (Algorithms)

  The algorithms used for testing are as follows:-

  1) Term Frequency Inverse Document Frequency Vectorizer(TF-IDF)

  2) Multinomial Naïve Bayes

  3) Gaussian Naïve Bayes

  4) Decision Tree classifier

  5) Random Forest classifier

  6) Ada Boost Classifier

  7) Binary Relevance

  8) Classifier Chain

- Run and Evaluate selected models

In [63]:
```
B_r_mnb
```

Out[63]:
```
{'accuracy': 0.9131016042780749, 'log_loss': 1.3429786840782607}
```

In [64]:
```
B_r_gnb=build_model(GaussianNB(),BinaryRelevance,x_train,y_train,x_test,y_test)
```

	precision	recall	f1-score	support
0	0.27	0.86	0.41	4582
1	0.03	0.84	0.07	486
2	0.17	0.89	0.29	2556
3	0.01	0.75	0.02	136
4	0.15	0.85	0.25	2389
5	0.02	0.80	0.05	432
micro avg	0.12	0.86	0.21	10581
macro avg	0.11	0.83	0.18	10581
weighted avg	0.20	0.86	0.31	10581
samples avg	0.04	0.08	0.05	10581

In [66]:
```
chain_model_Multi= build_model(MultinomialNB(),ClassifierChain,x_train,y_train,x_test,y_test)
```

	precision	recall	f1-score	support
0	0.95	0.46	0.62	4582
1	0.35	0.63	0.45	486
2	0.79	0.67	0.72	2556
3	0.07	0.45	0.12	136
4	0.69	0.62	0.65	2389
5	0.14	0.67	0.23	432
micro avg	0.57	0.56	0.57	10581
macro avg	0.50	0.58	0.47	10581
weighted avg	0.78	0.56	0.62	10581
samples avg	0.03	0.04	0.03	10581

In [68]:
```
chain_model_Gau= build_model(GaussianNB(),ClassifierChain,x_train,y_train,x_test,y_test)
```

	precision	recall	f1-score	support
0	0.27	0.86	0.41	4582
1	0.04	0.84	0.08	486
2	0.17	0.89	0.28	2556
3	0.01	0.75	0.02	136
4	0.14	0.87	0.24	2389
5	0.02	0.80	0.05	432
micro avg	0.12	0.86	0.21	10581
macro avg	0.11	0.83	0.18	10581
weighted avg	0.19	0.86	0.31	10581
samples avg	0.04	0.08	0.05	10581

In [70]:
```
chain_model_DTC= build_model(DecisionTreeClassifier(),ClassifierChain,x_train,y_train,x_test,y_test)
```

	precision	recall	f1-score	support
0	0.66	0.59	0.63	4582
1	0.25	0.15	0.19	486
2	0.73	0.67	0.70	2556
3	0.26	0.24	0.25	136
4	0.59	0.53	0.56	2389
5	0.39	0.28	0.33	432
micro avg	0.64	0.56	0.60	10581
macro avg	0.48	0.41	0.44	10581
weighted avg	0.63	0.56	0.59	10581
samples avg	0.05	0.05	0.05	10581

```
In [72]: chain_model_LR= build_model(LogisticRegression(),ClassifierChain,x_train,y_train,x_test,y_test)
```

	precision	recall	f1-score	support
0	0.90	0.57	0.69	4582
1	0.60	0.16	0.25	486
2	0.88	0.67	0.76	2556
3	0.72	0.19	0.30	136
4	0.72	0.60	0.66	2389
5	0.76	0.19	0.30	432
micro avg	0.84	0.56	0.67	10581
macro avg	0.77	0.40	0.49	10581
weighted avg	0.83	0.56	0.66	10581
samples avg	0.05	0.05	0.05	10581

```
In [76]: chain_model_RF= build_model(RandomForestClassifier(),ClassifierChain,x_train,y_train,x_test,y_test)
```

	precision	recall	f1-score	support
0	0.86	0.59	0.70	4582
1	0.46	0.08	0.14	486
2	0.85	0.70	0.77	2556
3	0.59	0.10	0.16	136
4	0.70	0.58	0.63	2389
5	0.72	0.18	0.29	432
micro avg	0.81	0.57	0.67	10581
macro avg	0.70	0.37	0.45	10581
weighted avg	0.79	0.57	0.65	10581
samples avg	0.05	0.05	0.05	10581

```
In [79]: chain_model_AB= build_model(AdaBoostClassifier(),ClassifierChain,x_train,y_train,x_test,y_test)
```

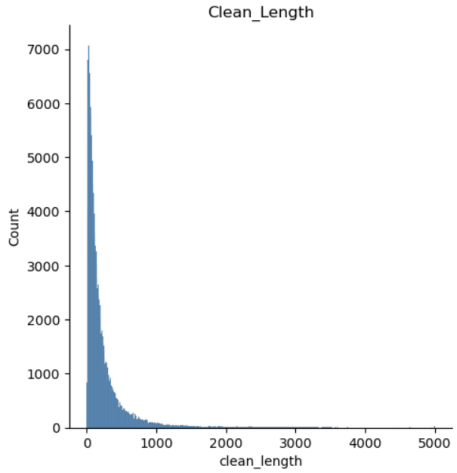
	precision	recall	f1-score	support
0	0.87	0.53	0.66	4582
1	0.53	0.22	0.31	486
2	0.85	0.68	0.76	2556
3	0.42	0.24	0.30	136
4	0.68	0.61	0.64	2389
5	0.53	0.24	0.33	432
micro avg	0.79	0.55	0.65	10581
macro avg	0.65	0.42	0.50	10581
weighted avg	0.79	0.55	0.64	10581
samples avg	0.04	0.05	0.04	10581

.

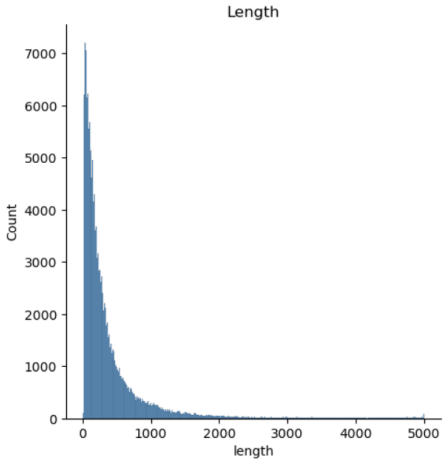
- Key Metrics for success in solving problem under consideration

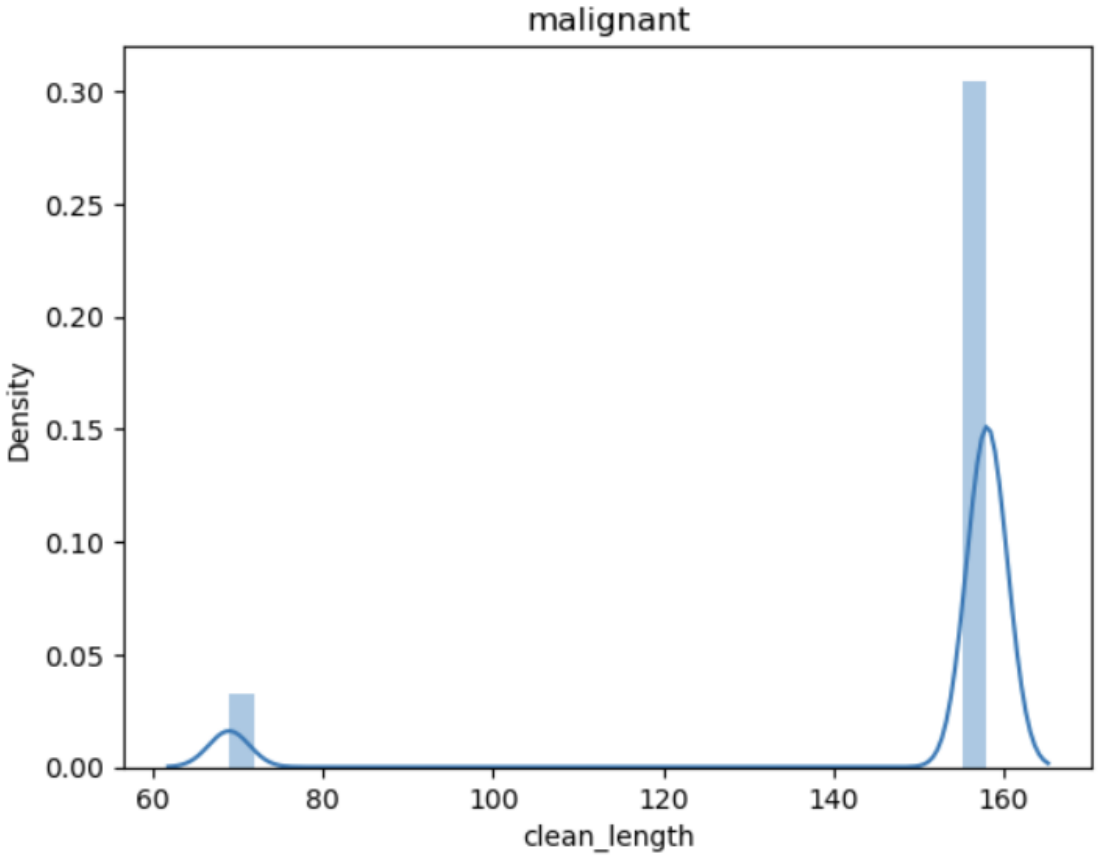
  The metrics used are accuracy_score, classification_report and Log_loss.
- Visualizations

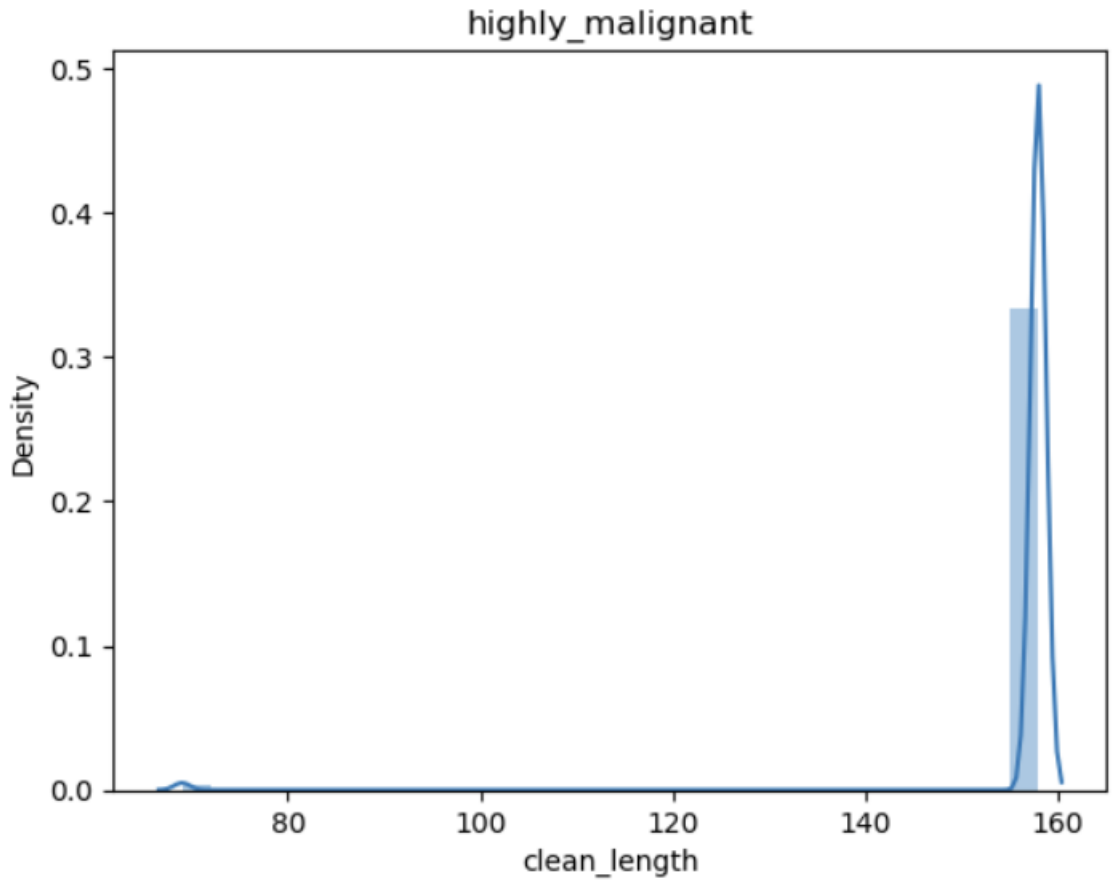
In [30]:
```
sns.displot(x='clean_length',data=df)
plt.title('Clean_Length')
plt.show()
```

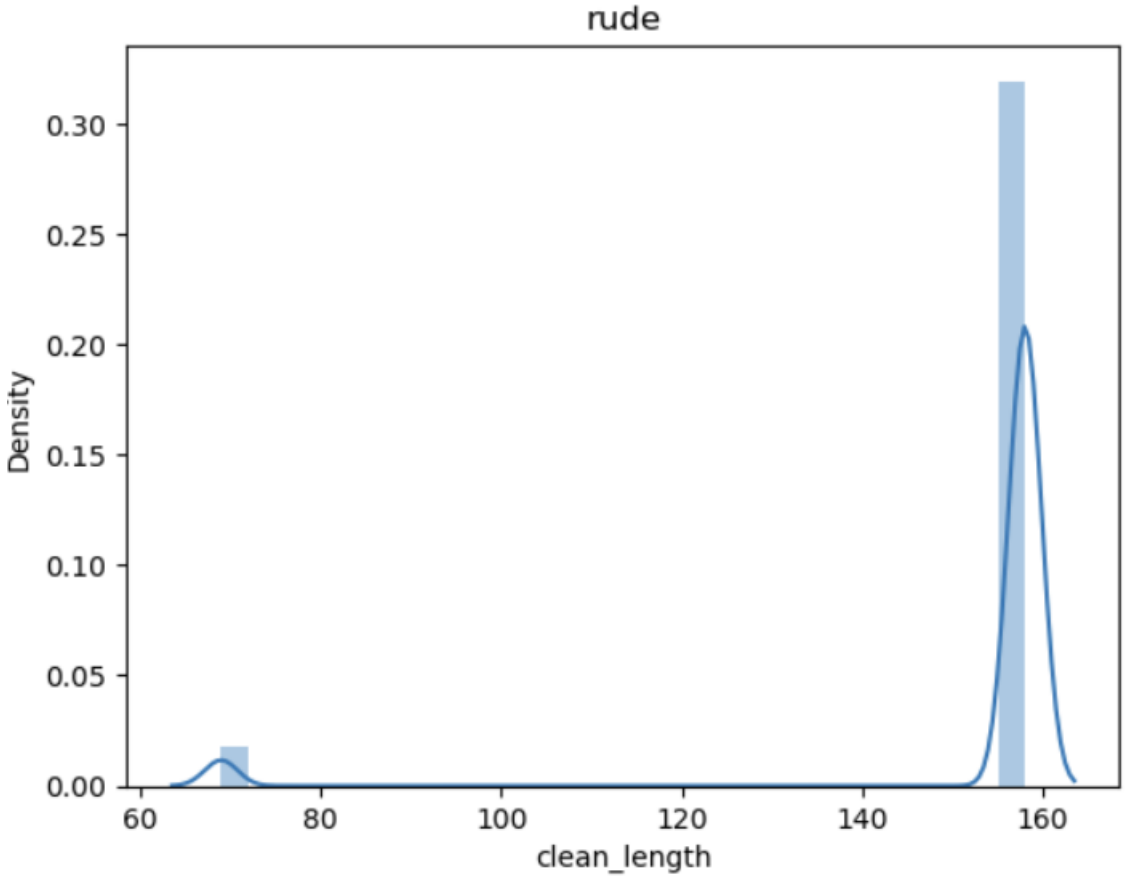


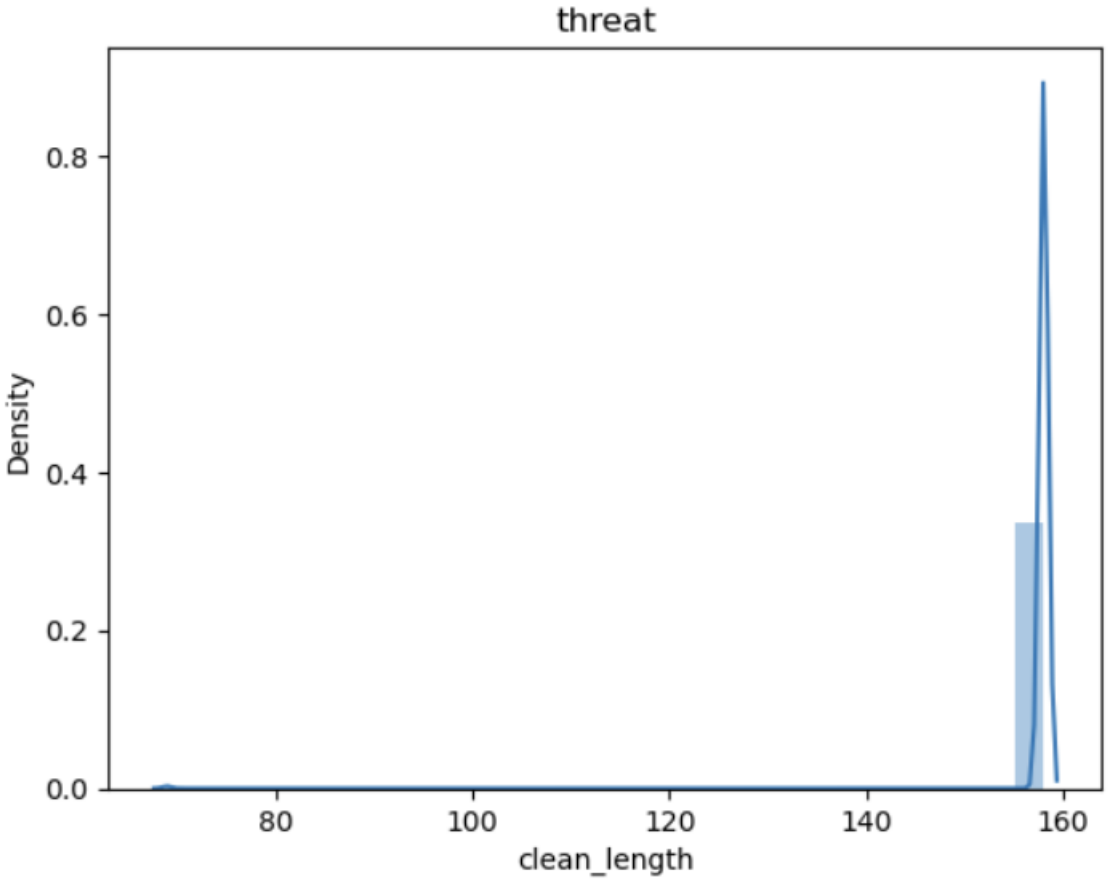
In [31]:
```
sns.displot(x='length',data=df)
plt.title('Length')
plt.show()
```

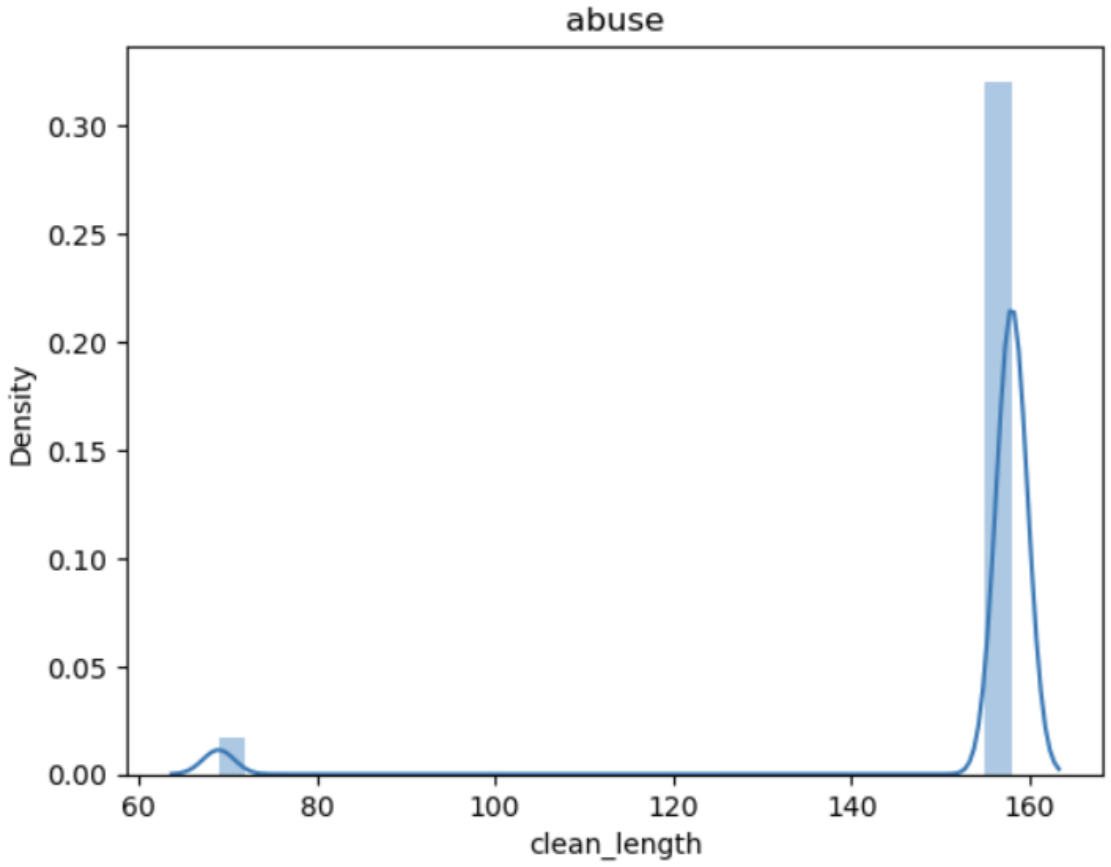


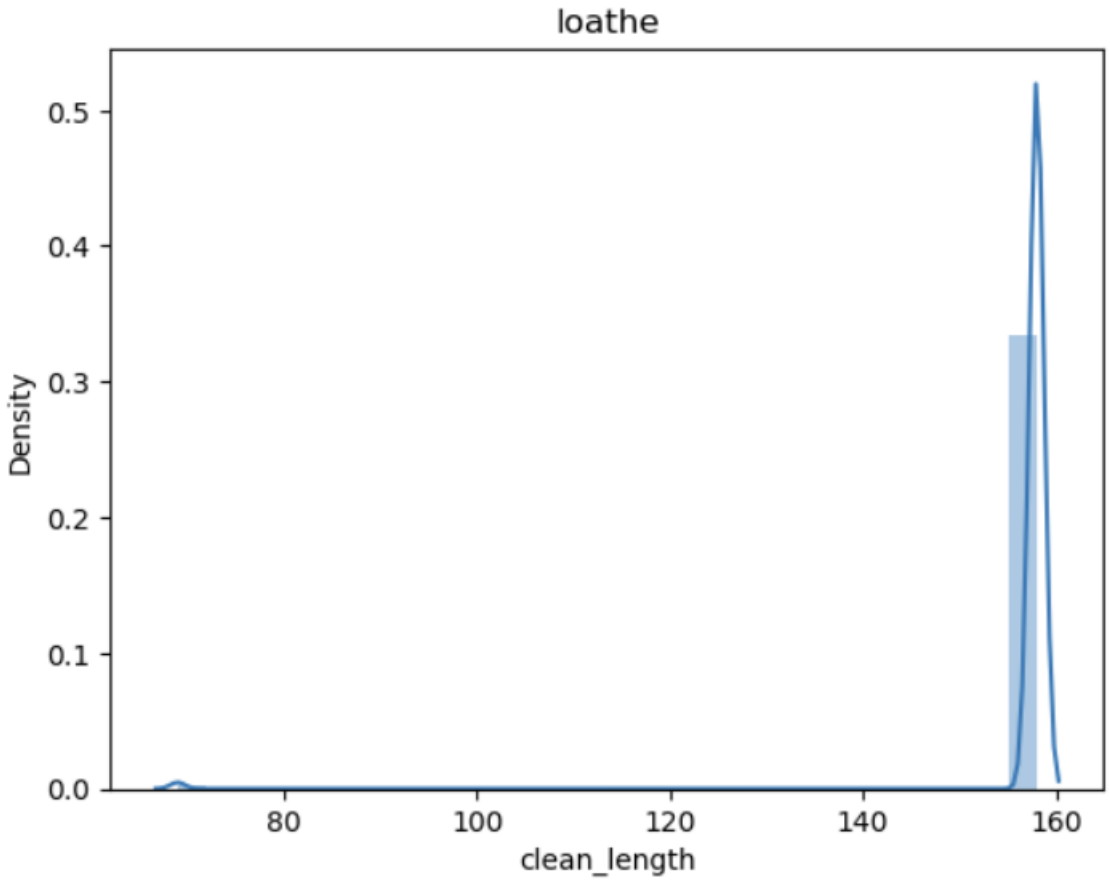



highly_malignant
Density
clean_length
0.0
0.1
0.2
0.3
0.4
0.5
80
100
120
140
160


rude
Density
clean_length
0.00
0.05
0.10
0.15
0.20
0.25
0.30
60
80
100
120
140
160


threat
Density
0.0
0.2
0.4
0.6
0.8
80
100
120
140
160
clean_length


abuse
Density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
60
80
100
120
140
160
clean_length



- Interpretation of the Results

The target variables have correlation with each other but not

with the length of text.

CONCLUSION

- Key Findings and Conclusions of the Study

1) Punctuation, blanks, numbers, URLs, phone numbers, email identifiers that have been deleted to reduce length.

2Malicious comments are closely connected to rude comments and comments about abuse.

- Learning Outcomes of the Study in respect of Data Science

  Clearing the data was critical in order to remove any potential contaminants.

  Punctuation, whitespaces, stop words, URLs, email ids and other special characters which need to be escaped for HTML

  There is no impact on the classification of a text based on its Lemmatization.

  After converting the text into a vector format, I have made it machine-readable.

  Converting a mutli-label variable into a multi-class will allow me to perform more accurate statistical analyses.

  It is important to be prepared for model making, as the variables

  Rephrase I have trained the model using an algorithm and found the best models based on various criteria the least log loss.

- Limitations of this work and Scope for Future Work

Rephrase The data in the study was imbalanced, so better results can be achieved by containing it.

Rephrase People also upload pictures and their comments with this article, thus making it a rich source of information.

Rephrase Neural networks can be used to classify these images too.