

### Data Collection and Preprocessing Phase

Date	15 October 2024
Team ID	739743
Project Title	Spooky Author Identification Using Deep Learning.
Maximum Marks	6 Marks

### Preprocessing Template

To preprocess data for spooky author identification using deep learning, start by loading and exploring the dataset to understand its structure, class distribution, and any missing or incomplete text entries. Handle missing data by dropping records with missing text, as imputing text may introduce noise. Clean the text by removing punctuation, numbers, special characters, and stop words, and convert it to lowercase for uniformity. Tokenize the text into words or subwords depending on the model. Convert the text into numeric representations using techniques like TF-IDF, Bag-of-Words, or word embeddings (e.g., GloVe, FastText). For advanced models like transformers, utilize pre-trained tokenizers to encode the text directly. Finally, split the preprocessed data into training and testing sets to develop and evaluate the deep learning model effectively.

Section	Description																		
Data Overview	<pre>data.describe()</pre> <table><thead><tr><th></th><th>author_id</th></tr></thead><tbody><tr><td>count</td><td>19579.000000</td></tr><tr><td>mean</td><td>0.905205</td></tr><tr><td>std</td><td>0.838595</td></tr><tr><td>min</td><td>0.000000</td></tr><tr><td>25%</td><td>0.000000</td></tr><tr><td>50%</td><td>1.000000</td></tr><tr><td>75%</td><td>2.000000</td></tr><tr><td>max</td><td>2.000000</td></tr></tbody></table>		author_id	count	19579.000000	mean	0.905205	std	0.838595	min	0.000000	25%	0.000000	50%	1.000000	75%	2.000000	max	2.000000
	author_id																		
count	19579.000000																		
mean	0.905205																		
std	0.838595																		
min	0.000000																		
25%	0.000000																		
50%	1.000000																		
75%	2.000000																		
max	2.000000																		

	<pre>plt.figure(figsize=(10, 6)) data.boxplot ()</pre>
Boxplot	<pre>&lt; # Display the DataFrame with the new column print(data)  import matplotlib.pyplot as plt  # Get value counts for 'author' column data = train_df['author'].value_counts()  # Create a boxplot for the distribution of counts (the values of 'data') plt.figure(figsize=(10, 6))  # Boxplot of the counts of authors plt.boxplot(data, vert=False, patch_artist=True,             boxprops=dict(facecolor='skyblue', color='blue'),             whiskerprops=dict(color='blue'),             capprops=dict(color='blue'),             flierprops=dict(markerfacecolor='red', marker='o', markersize=8))  # Add titles and labels plt.title('Boxplot of Author Distribution', fontsize=16) plt.xlabel('Counts', fontsize=14)</pre>
Outliers	<pre># Show the plot plt.tight_layout() plt.show()</pre>

	id	text \
0	id26305	This process, however, afforded me no means of...
1	id17569	It never once occurred to me that the fumbling...
2	id11008	In his left hand was a gold snuff box, from wh...
3	id27763	How lovely is spring As we looked from Windsor...
4	id12958	Finding nothing else, not even gold, the Super...
...	...	...
19574	id17718	I could have fancied, while I looked at it, th...
19575	id08973	The lids clenched themselves together as if in...
19576	id05267	Mais il faut agir that is to say, a Frenchman ...
19577	id17513	For an item of news like this, it strikes us i...
19578	id00393	He laid a gnarled claw on my shoulder, and it ...
		text_id
0		[26, 2945, 143, 1372, 22, 36, 294, 2, 7451, 1,...
1		[11, 89, 125, 723, 4, 22, 9, 1, 5924, 79, 28, ...
2		[7, 15, 144, 173, 8, 5, 714, 4929, 560, 23, 18...
3		[121, 595, 25, 779, 16, 34, 212, 23, 696, 4246...
4		[1126, 166, 680, 20, 76, 714, 1, 4930, 1794, 1...
...		...
19574		[6, 46, 31, 1130, 102, 6, 212, 21, 11, 9, 54, ...
19575		[1, 2616, 7330, 365, 396, 16, 62, 7, 5, 7596]
19576		[10440, 6083, 15918, 25943, 9, 25, 4, 128, 5, ...
19577		[17, 37, 4653, 2, 2703, 82, 26, 11, 9643, 84, ...
19578		[13, 1354, 5, 6664, 6557, 27, 10, 1670, 3, 11,...
		[19579 rows x 3 columns]

## Loading Data

	id	text	author
0	id26305	This process, however, afforded me no means of...	EAP
1	id17569	It never once occurred to me that the fumbling...	HPL
2	id11008	In his left hand was a gold snuff box, from wh...	EAP
3	id27763	How lovely is spring As we looked from Windsor...	MWS
4	id12958	Finding nothing else, not even gold, the Super...	HPL

	id	text	author	text_id	author_id
0	id26305	This process, however, afforded me no means of...	EAP	[26, 2945, 143, 1372, 22, 36, 294, 2, 7451, 1,...	0
1	id17569	It never once occurred to me that the fumbling...	HPL	[11, 89, 125, 723, 4, 22, 9, 1, 5924, 79, 28, ...	1
2	id11008	In his left hand was a gold snuff box, from wh...	EAP	[7, 15, 144, 173, 8, 5, 714, 4929, 560, 23, 18...	0
3	id27763	How lovely is spring As we looked from Windsor...	MWS	[121, 595, 25, 779, 16, 34, 212, 23, 696, 4246...	2
4	id12958	Finding nothing else, not even gold, the Super...	HPL	[1126, 166, 680, 20, 76, 714, 1, 4930, 1794, 1...	1

`data.tail()`

	id	text	author	text_id	author_id
19574	id17718	I could have fancied, while I looked at it, th...	EAP	[6, 46, 31, 1130, 102, 6, 212, 21, 11, 9, 54, ...	0
19575	id08973	The lids clenched themselves together as if in...	EAP	[1, 2616, 7330, 365, 396, 16, 62, 7, 5, 7596]	0
19576	id05267	Mais il faut agir that is to say, a Frenchman ...	EAP	[10440, 6083, 15918, 25943, 9, 25, 4, 128, 5, ...	0
19577	id17513	For an item of news like this, it strikes us i...	EAP	[17, 37, 4653, 2, 2703, 82, 26, 11, 9643, 84, ...	0
19578	id00393	He laid a gnarled claw on my shoulder, and it ...	HPL	[13, 1354, 5, 6664, 6557, 27, 10, 1670, 3, 11,...	1

```
test_df=pd.read_csv('/content/test.csv')
test_df.head()
```

	id	text
0	id02310	Still, as I urged our leaving Ireland with suc...
1	id24541	If a fire wanted fanning, it could readily be ...
2	id00134	And when they had broken down the frail door t...
3	id27757	While I was thinking how I should possibly man...
4	id04081	I am not sure to what limit his knowledge may ...



```
data.isnull()
```

	id	text	author	text_id	author_id
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...	...	...	...	...	...
19574	False	False	False	False	False
19575	False	False	False	False	False
19576	False	False	False	False	False
19577	False	False	False	False	False
19578	False	False	False	False	False

19579 rows × 5 columns

	<pre>data.info()</pre> <pre>&lt;class 'pandas.core.frame.DataFrame'&gt; RangeIndex: 19579 entries, 0 to 19578 Data columns (total 5 columns):</pre> <pre>y_pred</pre> <pre>array([[0.32830974, 0.42775834, 0.24393189],        [0.32830974, 0.42775834, 0.24393189],        [0.32830974, 0.42775834, 0.24393189],        ...,        [0.32830974, 0.42775834, 0.24393189],        [0.32830974, 0.42775834, 0.24393186],        [0.32830974, 0.42775834, 0.24393186]], dtype=float32)</pre>
Save Processed Data	<pre>import pickle import joblib joblib.dump(train_df, 'train_df.pkl') joblib.dump(test_df, 'test_df.pkl') joblib.dump(data, 'data.pkl')</pre> <pre>['data.pkl']</pre>