

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Load the uploaded CSV file to analyze its structure
file_path = '/content/drive/MyDrive/Colab Notebooks/size_number_spacing.csv'
data = pd.read_csv(file_path)

# Display the first few rows of the data to understand its structure
data.head()
```

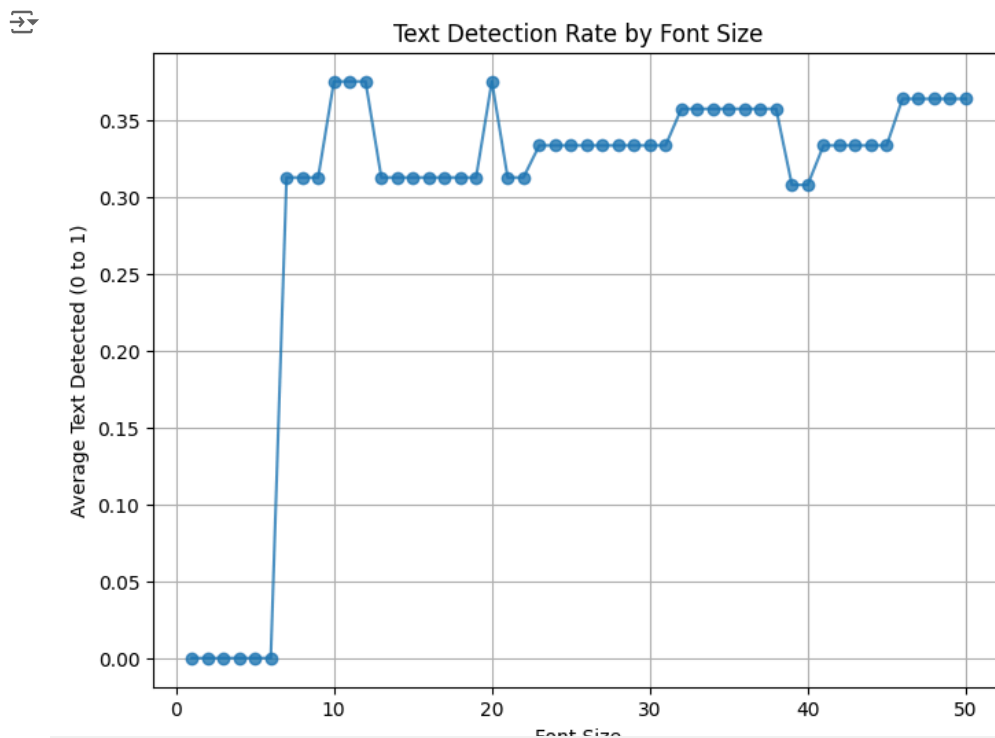
	Font Size	Number of Letters	Letterspacing	Text Detected
0	10	4	0	0
1	11	4	0	0
2	12	4	0	0
3	13	4	0	0
4	14	4	0	0

Next steps:

[Generate code with data](#)[View recommended plots](#)[New interactive sheet](#)

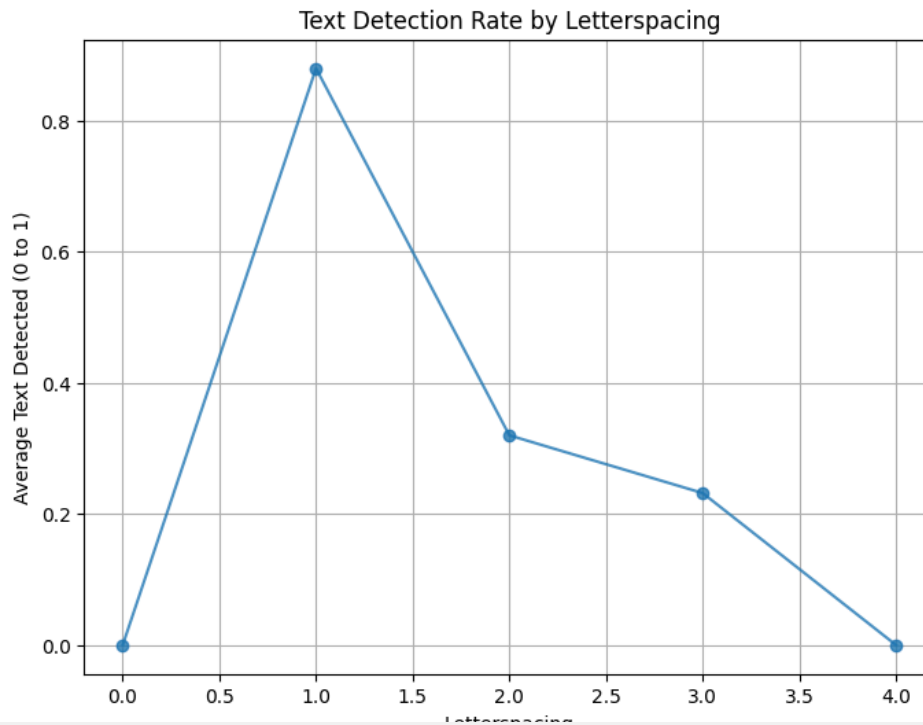
```
# Create a line graph showing Font Size vs. Text Detected
# Group the data by Font Size to calculate the average detection rate for each size
font_size_detection = data.groupby('Font Size')['Text Detected'].mean()

# Plot the line graph
plt.figure(figsize=(8, 6))
plt.plot(font_size_detection.index, font_size_detection.values, marker='o', linestyle='-', alpha=0.8)
plt.title('Text Detection Rate by Font Size')
plt.xlabel('Font Size')
plt.ylabel('Average Text Detected (0 to 1)')
plt.grid(True)
plt.show()
```



```
# Create a line graph showing Letterspacing vs. Text Detected
# Group the data by Letterspacing to calculate the average detection rate for each spacing value
spacing_detection = data.groupby('Letterspacing')['Text Detected'].mean()
```

```
# Plot the line graph
plt.figure(figsize=(8, 6))
plt.plot(spacing_detection.index, spacing_detection.values, marker='o', linestyle='-', alpha=0.8)
plt.title('Text Detection Rate by Letterspacing')
plt.xlabel('Letterspacing')
plt.ylabel('Average Text Detected (0 to 1)')
plt.grid(True)
plt.show()
```



```
# Create a line graph showing Number of Letters vs. Text Detected
# Group the data by Number of Letters to calculate the average detection rate for each count
letters_detection = data.groupby('Number of Letters')['Text Detected'].mean()

# Plot the line graph
plt.figure(figsize=(8, 6))
plt.plot(letters_detection.index, letters_detection.values, marker='o', linestyle='-', alpha=0.8)
plt.title('Text Detection Rate by Number of Letters')
plt.xlabel('Number of Letters')
plt.ylabel('Average Text Detected (0 to 1)')
plt.grid(True)
plt.show()
```

