

Telecommunication Analysis



shutterstock.com · 1889984209

Task 1 User Overview Analysis

General Statistics

	Bearer Id	Start ms	End ms	Dur. (ms)	IMSI	MSISDN/Number	IMEI	Avg RTT DL (ms)	Avg RTT UL (ms)	Avg Bearer TP DL (kbps)	Avg Bearer TP UL (kbps)	TCP DL F
count	149,010	150,000	150,000	150,000	149,431	148,935	149,429	122,172	122,189	150,000	150,000	
mean	⚠ 10138871842091135000	499.1882	498.8009	104,608.5603	208,201,691,804,244.1	41,882,818,562.27	48,474,540,644,914.984	109.7957	17.6629	13,300.0459	1,770.4286	
std	⚠ 2893169620094505500	288.6118	288.0977	81,037.6215	21,492,888,214.7297	2,447,443,061,791.689	22,416,374,220,220.31	619.7827	84.7935	23,971.8785	4,625.3555	
min	⚠ 69175400000000000000000000000000	0	0	7,142	204,047,000,000,000	33,601,001,722	440,015,000,000	0	0	0	0	
25%	⚠ 73498800000000000000000000000000	250	251	57,440.5	208,201,000,000,000	33,651,295,581.5	35,460,700,000,000	32	2	43	47	
50%	⚠ 73498800000000000000000000000000	499	500	86,399	208,202,000,000,000	33,663,706,799	35,722,000,000,000	45	5	63	63	
75%	⚠ 13042400000000000000000000000000	749	750	132,430.25	208,202,000,000,000	33,683,490,769	86,119,700,000,000	70	15	19,710.75	1,120	
max	⚠ 13186500000000000000000000000000	999	999	1,859,336	214,074,000,000,000	882,397,000,000,000	99,001,200,000,000	96,923	7,120	378,160	58,613	

Null Values:

	0
Bearer Id	991
Start	1
Start ms	1
End	1
End ms	1
Dur. (ms)	1
IMSI	570
MSISDN/Number	1,066
IMEI	572
Last Location Nam	1,153

Top 10 Handsets Used by Customers:

PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

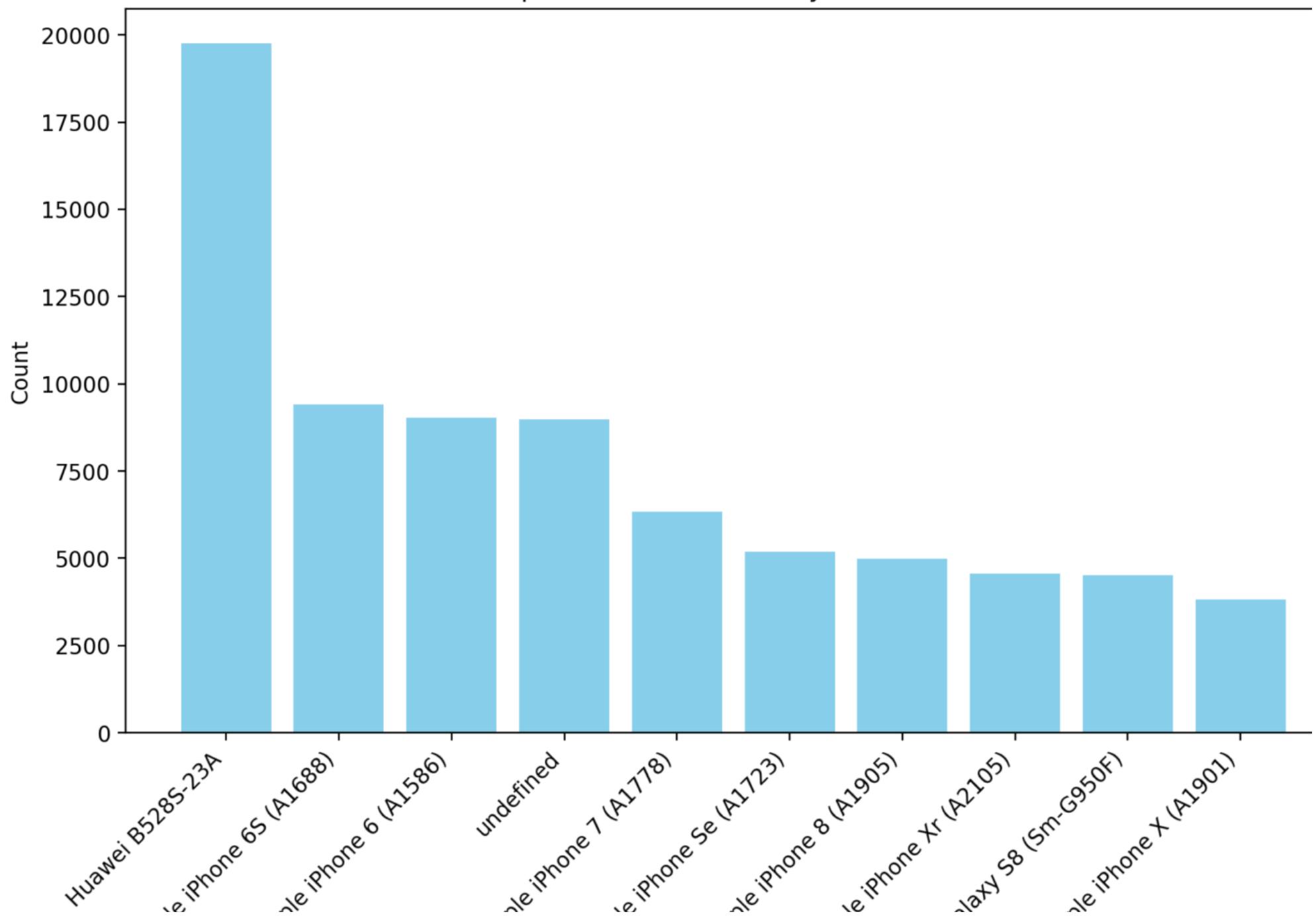
You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

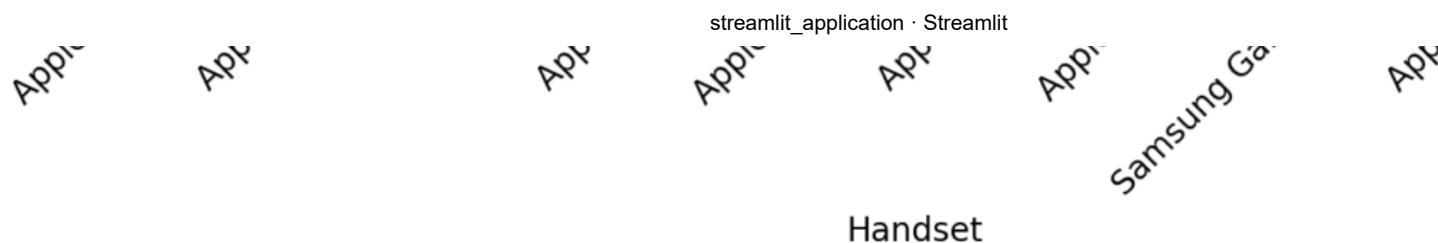
```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```

Top 10 Handsets Used by Customers





Identifying The Top 3 Handsets Manufactures

PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

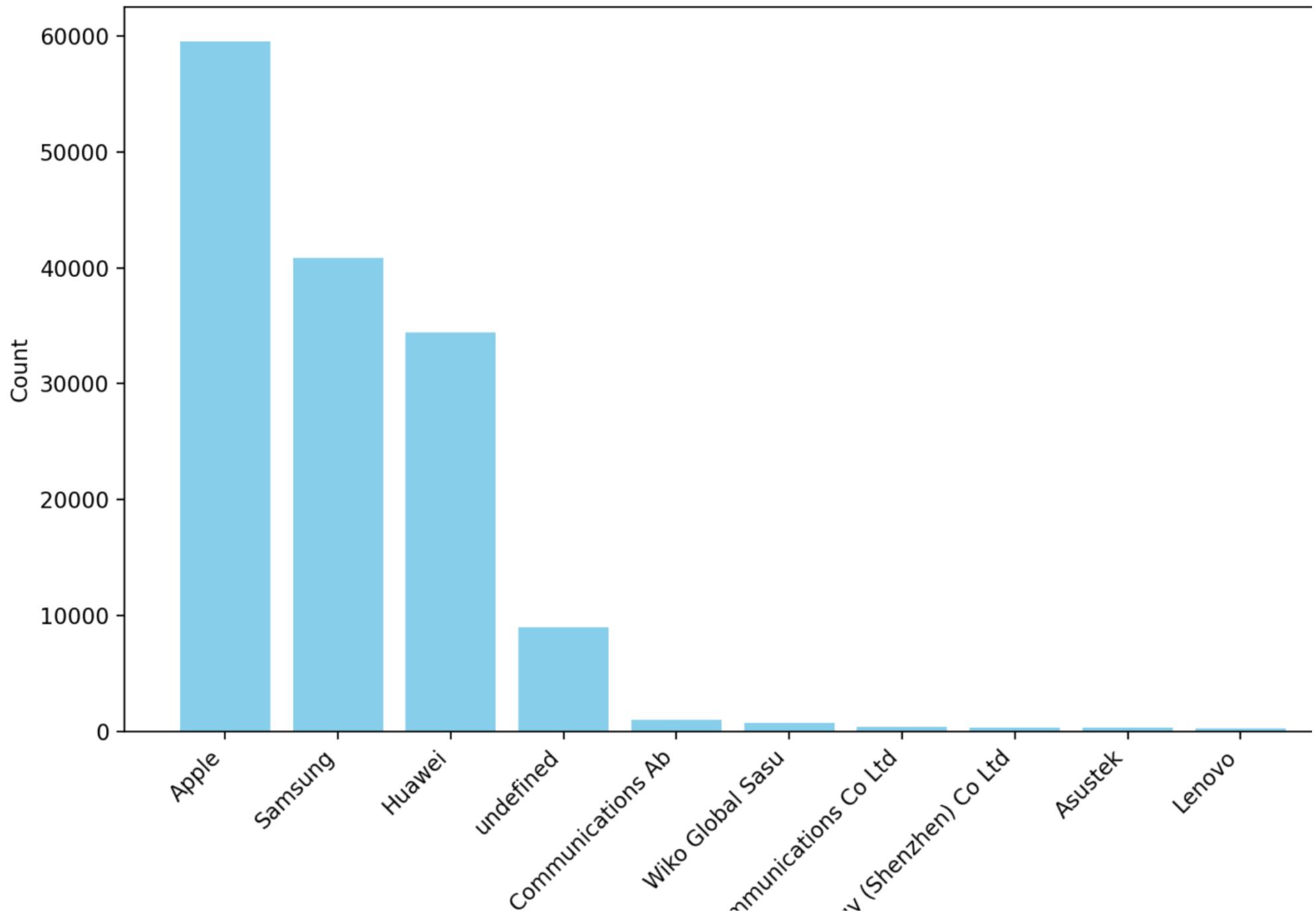
You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```

Identifying The Top 3 Handsets Manufacturers



Sony Mobile
Xiaomi Cor
Oneplus Technolog
Handset

Scatter Plots for Each Pair of Download and Upload Columns:

PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

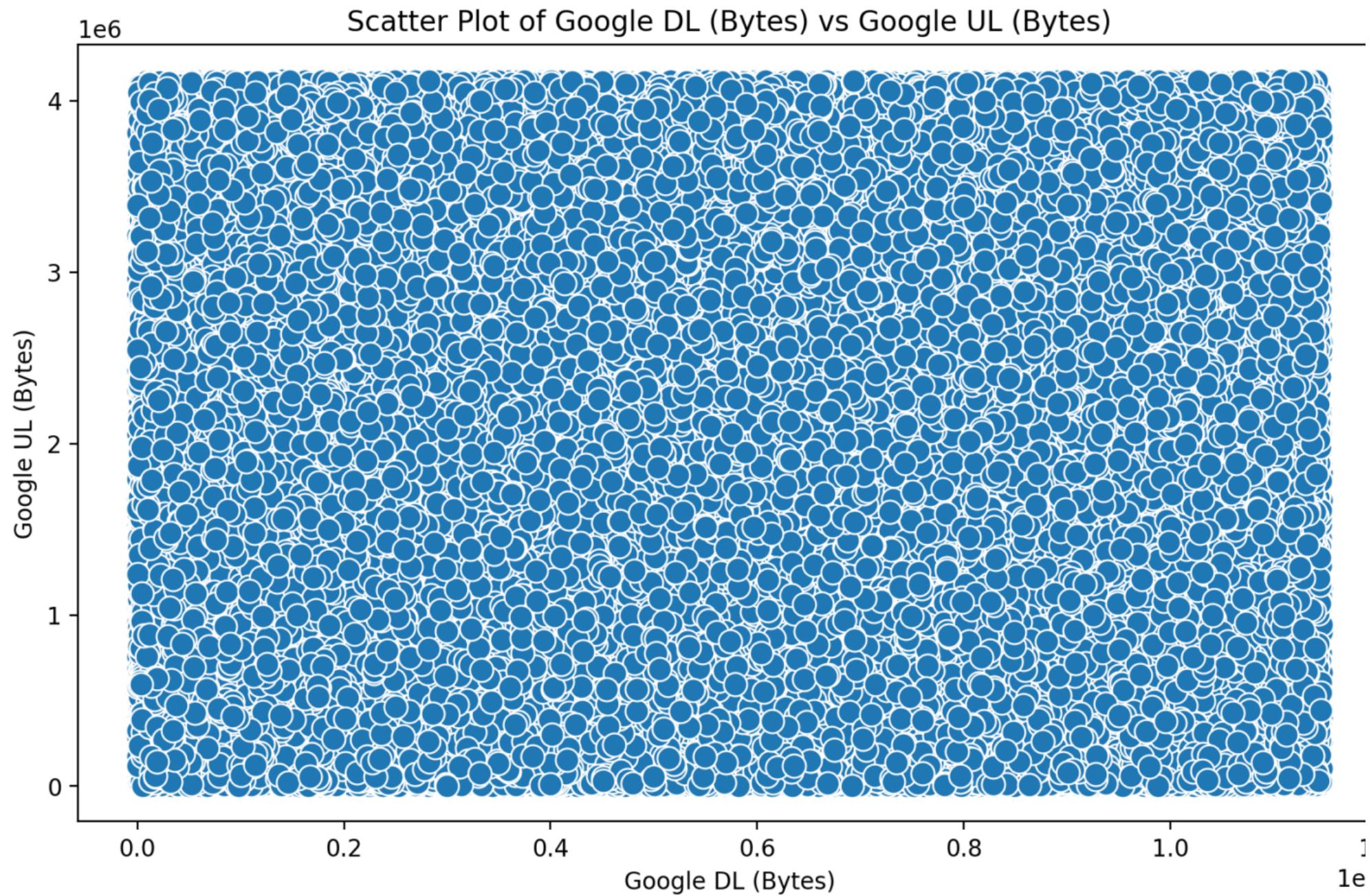
```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```



PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

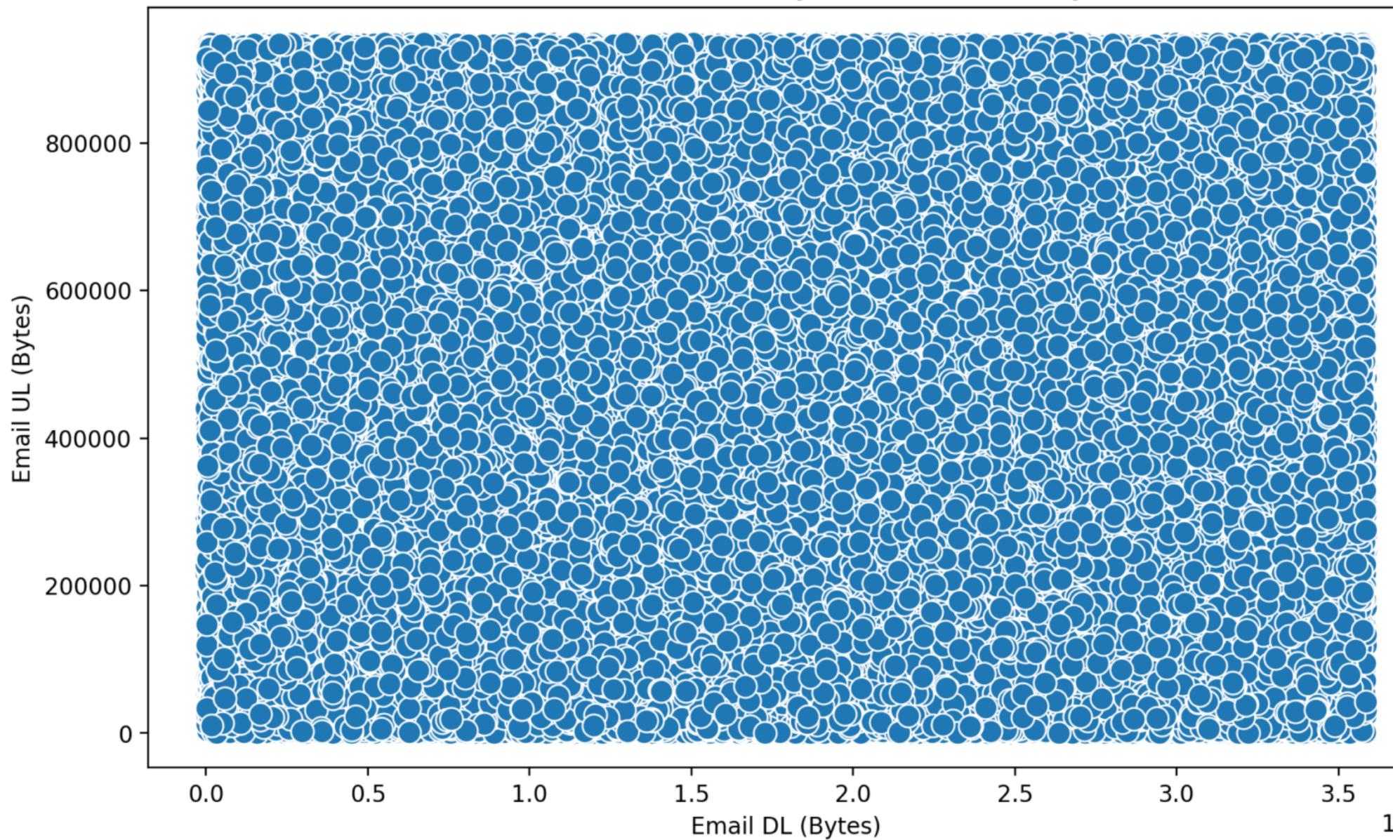
You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```

Scatter Plot of Email DL (Bytes) vs Email UL (Bytes)



PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

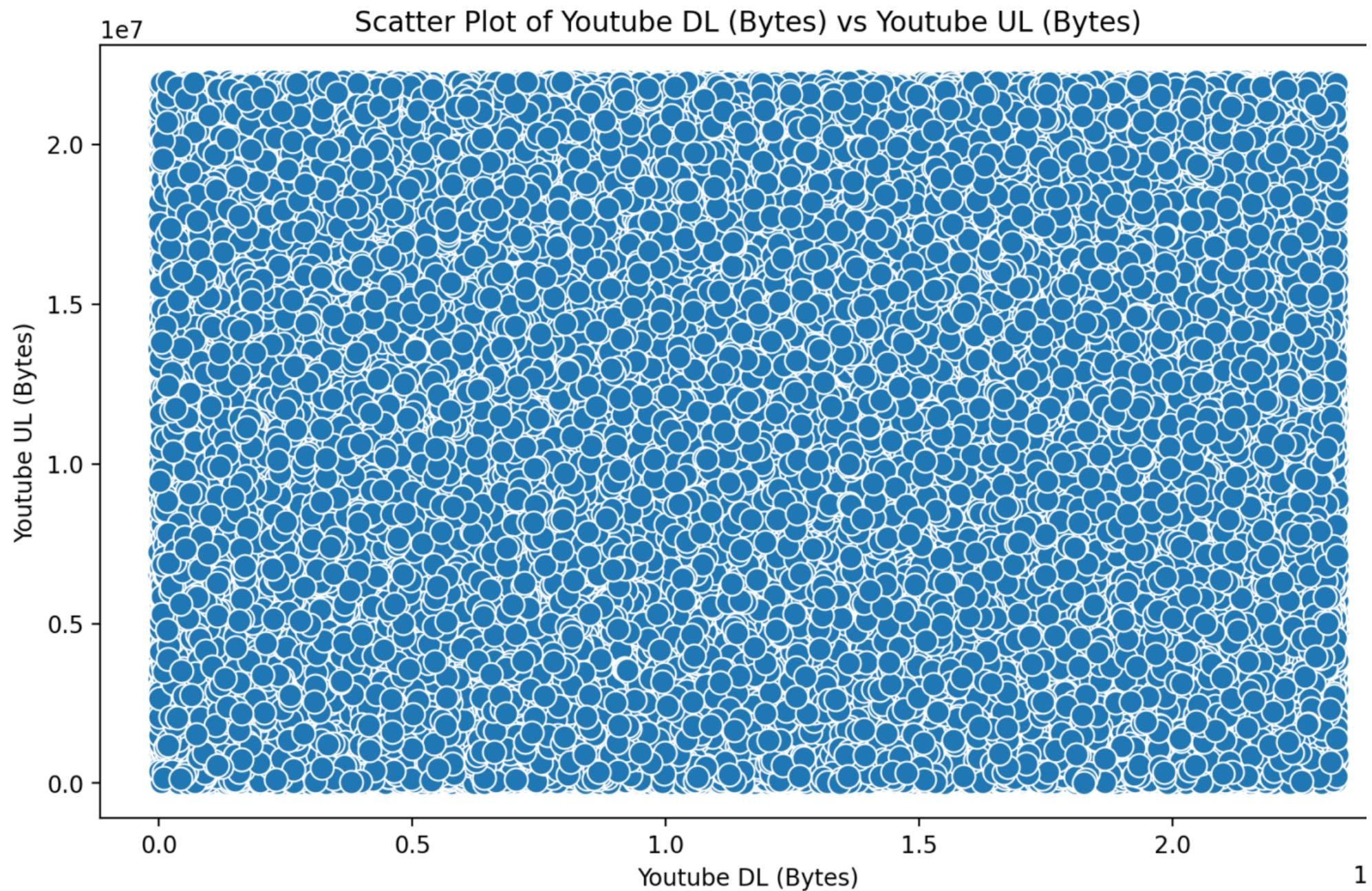
```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```



PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

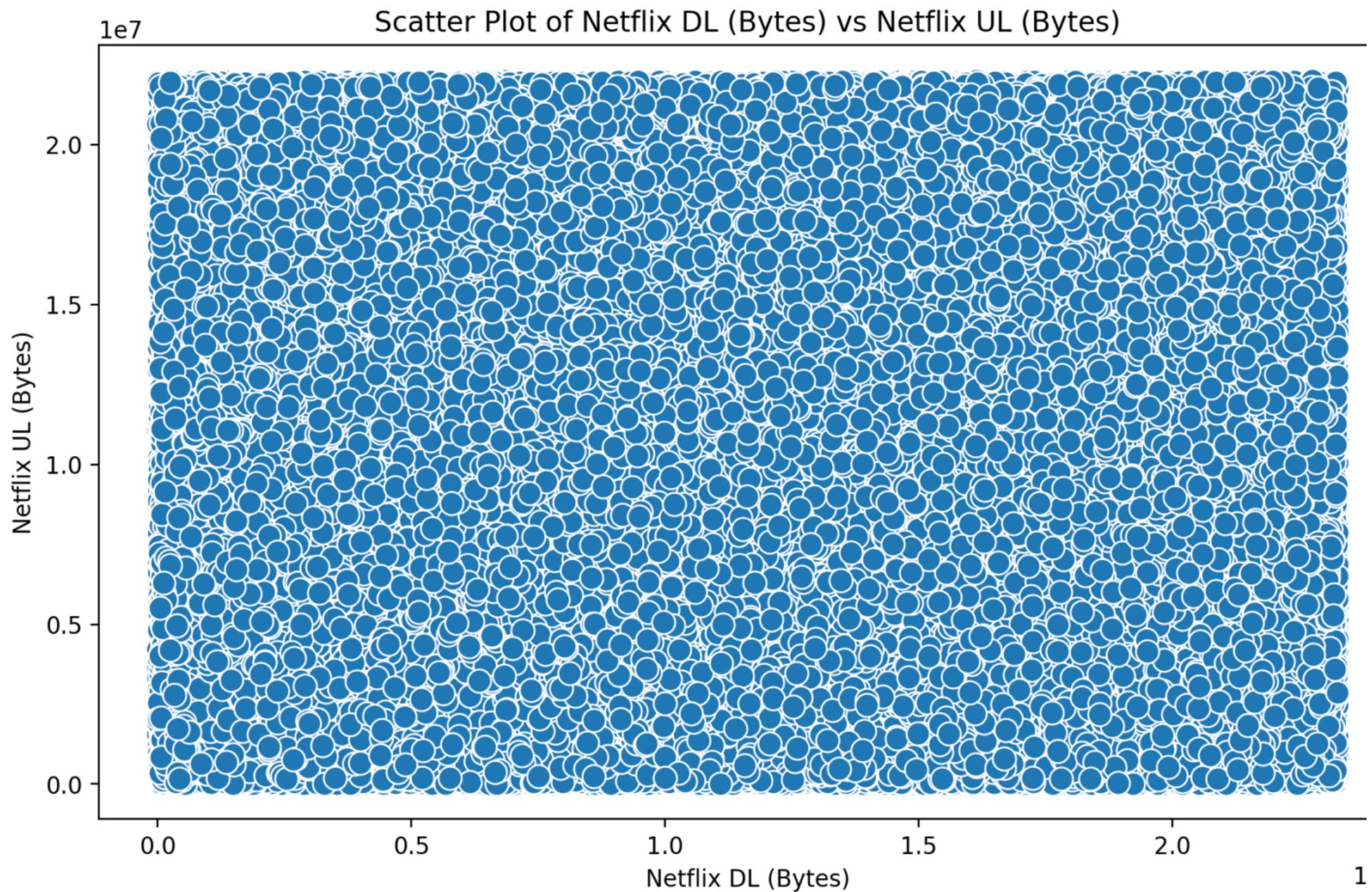
```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```



PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

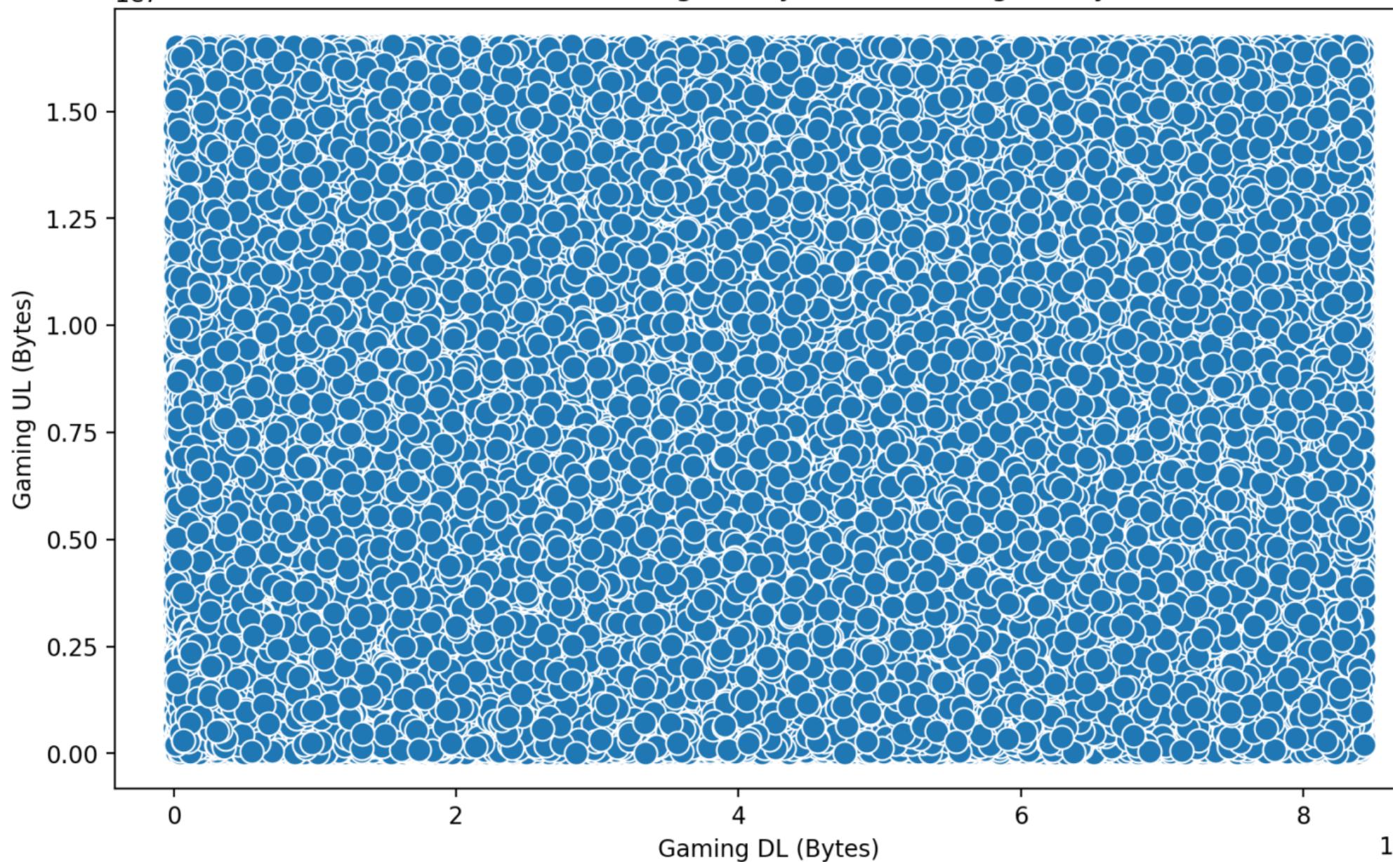
You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```

Scatter Plot of Gaming DL (Bytes) vs Gaming UL (Bytes)



PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

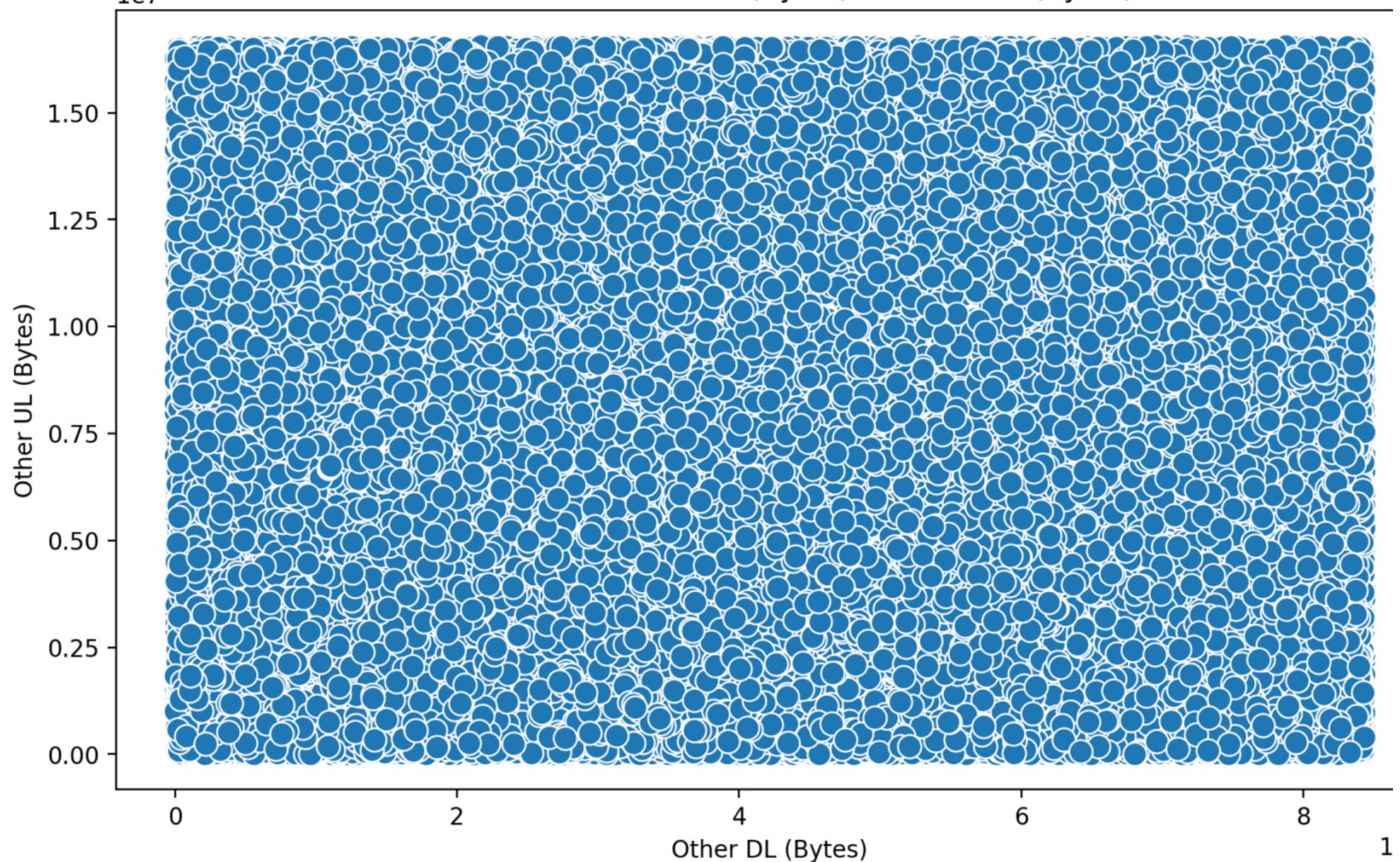
You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```

Scatter Plot of Other DL (Bytes) vs Other UL (Bytes)



PyplotGlobalUseWarning: You are calling `st.pyplot()` without any arguments. After December 1st, 2020, we will remove the ability to do this as it requires the use of Matplotlib's global figure object, which is not thread-safe.

To future-proof this code, you should pass in a figure as shown below:

```
>>> fig, ax = plt.subplots()  
>>> ax.scatter([1, 2, 3], [1, 2, 3])  
>>> ... other plotting actions ...  
>>> st.pyplot(fig)
```

You can disable this warning by disabling the config option: `deprecation.showPyplotGlobalUse`

```
st.set_option('deprecation.showPyplotGlobalUse', False)
```

or in your `.streamlit/config.toml`

```
[deprecation]  
showPyplotGlobalUse = false
```

