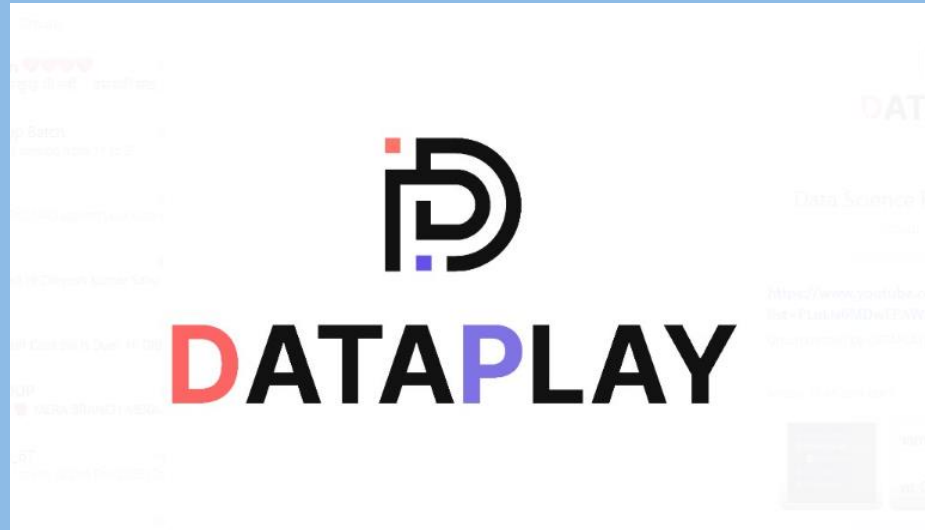


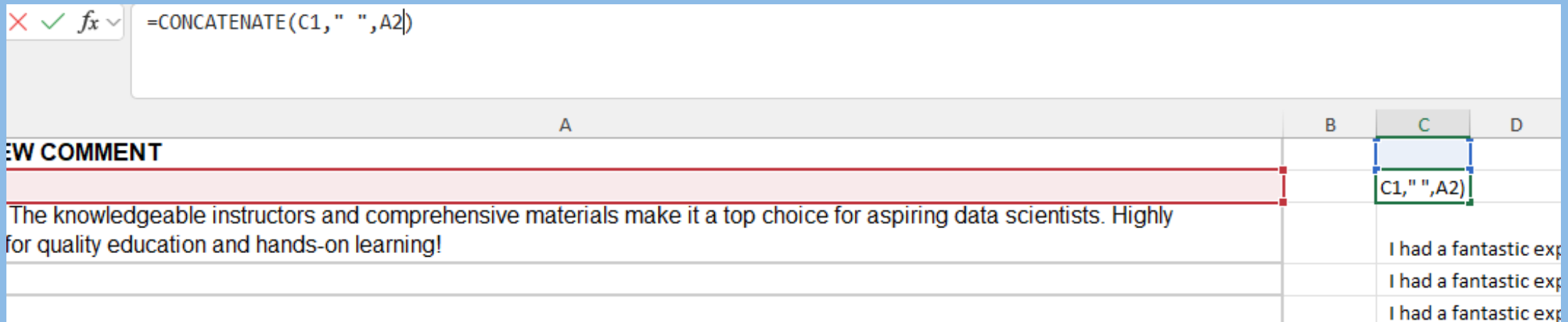
Project: Creating Wordcloud using Python in Excel



We are going to use google reviews by students for Dataplay as the dataset. We need some data cleaning before we can finally create the wordcloud. Let's proceed.

Steps

1. To get all the reviews in a single cell, let us start with using concatenate function-



Steps

By dragging down to the lowermost cell, we get all the content in a single cell.

2. Let us copy and paste the value of that particular cell into another cell.

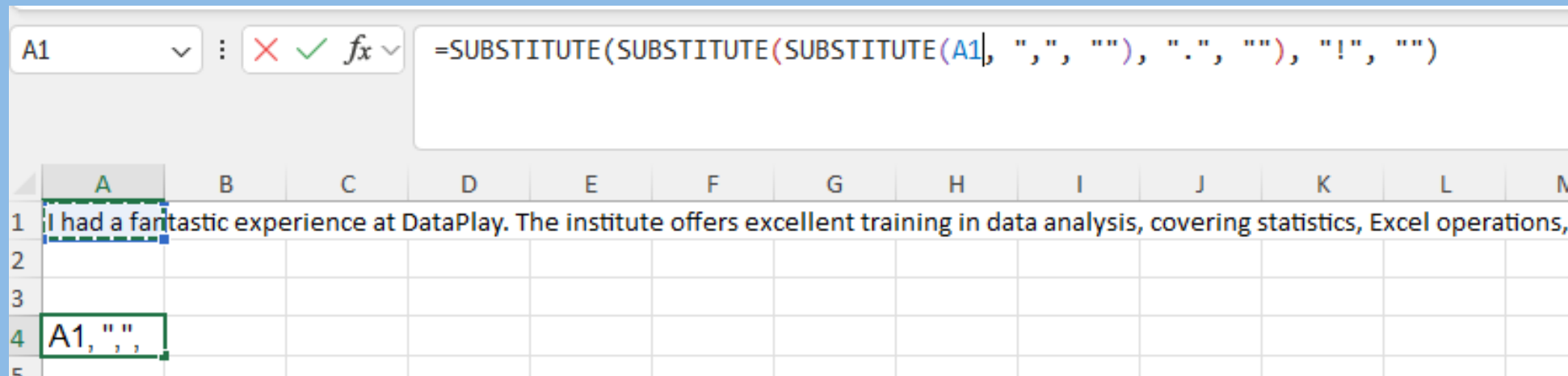
=CONCATENATE(C58,A59)

A	B	C	D
to work with.		I had a fantastic exper	
rk with.		I had a fantastic exper	
		I had a fantastic exper	
data science.		I had a fantastic exper	
		I had a fantastic exper	
rack. 🦾🦾.		I had a fantastic exper	
		I had a fantastic exper	
		I had a fantastic exper	
rk with.		I had a fantastic exper	
		I had a fantastic exper	

A1	:	✕	✓	<i>fx</i>	✕	✓	<i>fx</i>	I had a fantastic experience at DataPlay. The institu
								instructors and comprehensive materials make it a
								very good place for learning with a good hearted te
A	B	C	D	E	F	G	H	
I had a fantastic experience at DataPlay. The institute offers excellent training in dat								

Steps

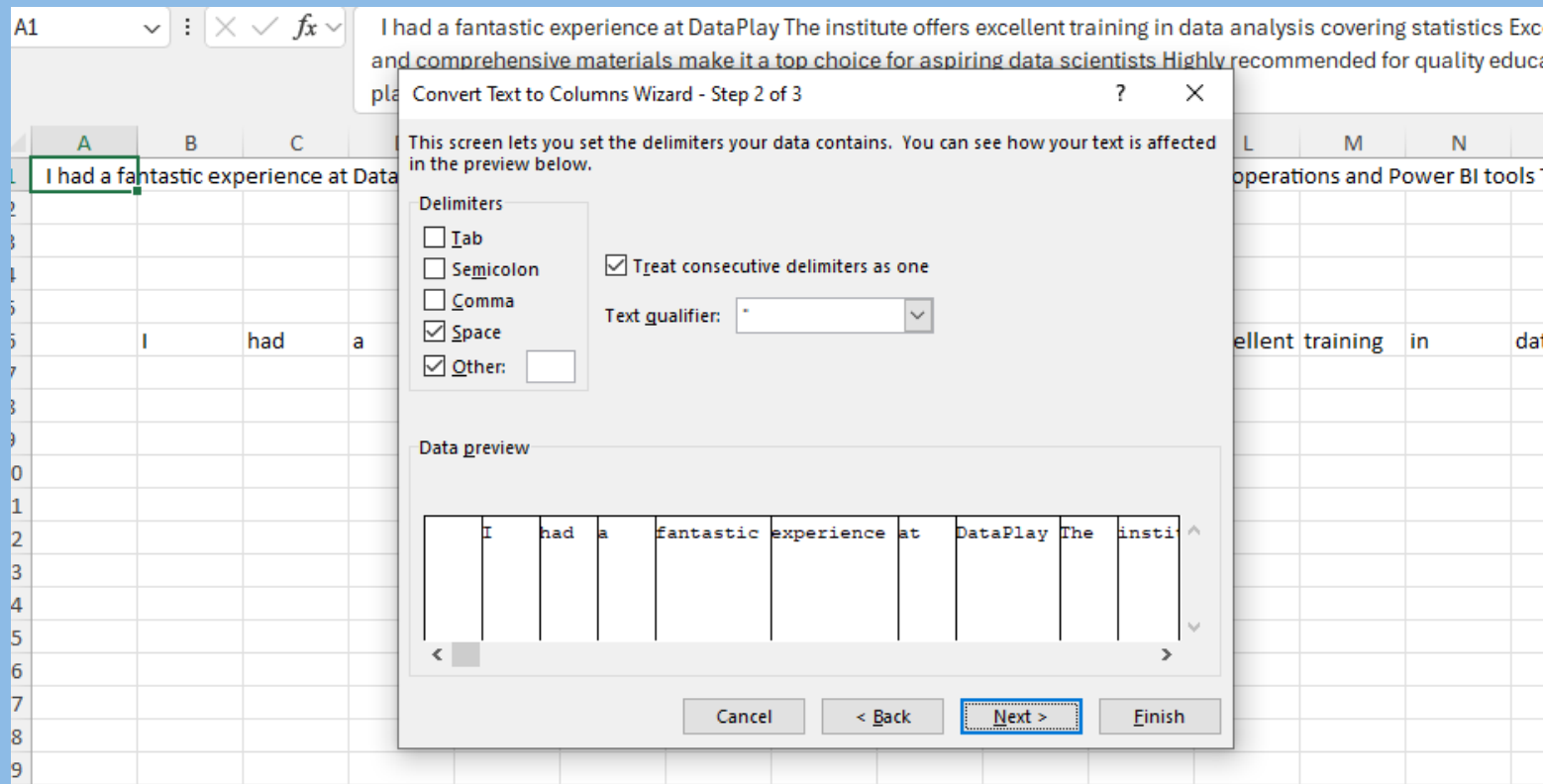
3. Let us replace all the commas, full-stops and exclamation marks with an empty string “”.



Let us copy and paste the cell value to another cell.

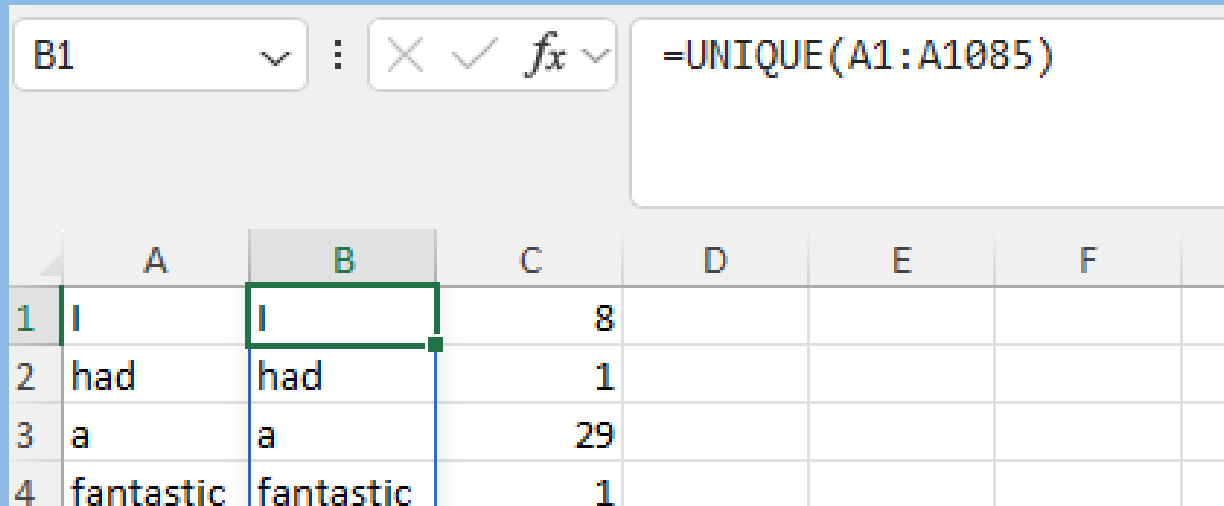
Steps

4. Now, let us separate out all the words using 'text-to-columns' option. The delimiters should be space and line-breaks. For line-breaks, we need to use ctrl+j in 'other'.



Steps

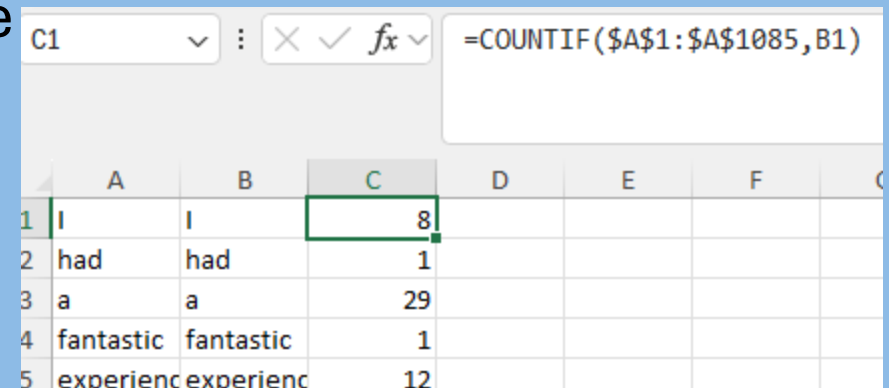
5. For proper navigation, we copied and pasted the data using transpose. Now, using the UNIQUE() function, we got a series of words which are unique without any duplication.



The screenshot shows the Excel formula bar with the formula `=UNIQUE(A1:A1085)` entered. Below the formula bar, a table displays the results of the UNIQUE function. The table has columns A, B, C, D, E, and F. Column A contains the unique words, column B contains the same words, and column C contains the count of each word. The words are 'I', 'had', 'a', and 'fantastic'.

	A	B	C	D	E	F
1	I	I	8			
2	had	had	1			
3	a	a	29			
4	fantastic	fantastic	1			

Now, using COUNTIF() function, we got the count of the repetition of each unique word.

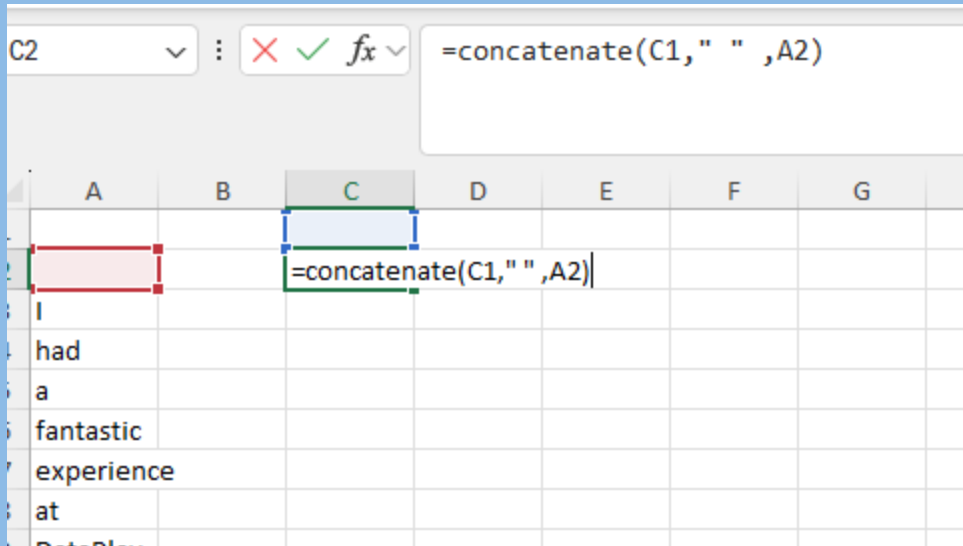


The screenshot shows the Excel formula bar with the formula `=COUNTIF(A1:A1085,B1)` entered. Below the formula bar, a table displays the results of the COUNTIF function. The table has columns A, B, C, D, E, F, and G. Column A contains the unique words, column B contains the same words, and column C contains the count of each word. The words are 'I', 'had', 'a', 'fantastic', and 'experienc'.

	A	B	C	D	E	F	G
1	I	I	8				
2	had	had	1				
3	a	a	29				
4	fantastic	fantastic	1				
5	experienc	experienc	12				

Steps

6. Again, we copied and pasted the dataset in transposed state in another sheet. Then, we joined all the single words using spaces in between to get the string data in a single cell.

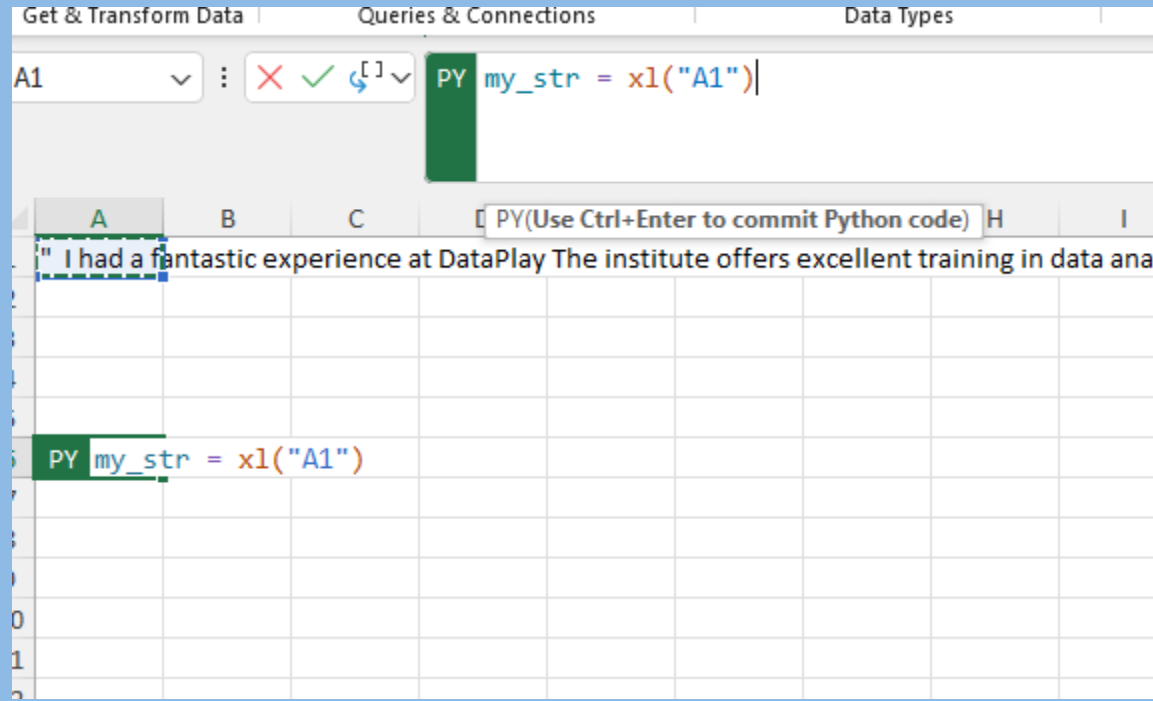


By dragging it to the last cell, we got all the words in a single cell.

Steps

7. After copying and pasting the values to another cell, we put the whole string inside “”.

8. In Python code, we saved the whole string into the variable my_str.



Steps

9. In another cell, using Python we run the following code:

```
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer

# Input text (replace this with your actual text)
text = my_str

# Define stopwords and add "data" to the List
stopwords = set(STOPWORDS)
stopwords.add("data")

# Generate bigrams from the text
def generate_bigrams(text):
    words = text.split()
    bigrams = [' '.join(pair) for pair in zip(words, words[1:])]
    return ' '.join(words + bigrams)

# Prepare the text with bigrams
text_with_bigrams = generate_bigrams(text)

# Create a WordCloud instance
wordcloud = WordCloud(stopwords=stopwords, width=800, height=400, background_color='white', colormap='viridis').generate(text_with_bigrams)

# Display the Word Cloud
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title("Word Cloud: Unigrams and Bigrams (Excluding Stopwords)", fontsize=16)
plt.show()
```

Steps

10. Finally, we got this Wordcloud.

