Back

Congrats Daniel! This project has been marked as completed.

Project Rating

Teacher's Comment

"Good"

Was this helpful?

PRO-C119: DATA PRE PROCESSING Completed

In Class 119, You Have Learned To Preprocess Your Data Using The Nltk Module. In This Project You Will Create A Bag Of Words Encoding For Our Data, So That We Can Use It To Train Our Chatbot.

Goal of the Project:

In class 119, you have learned to preprocess your data using the nltk module. In this project you will create a Bag of Words encoding for our data, so that we can use it to train our chatbot.

Story:

In the last class, we created a system for Alex where he can get the sentiments on reviews which are associated with his products. To make it a seamless experience for customers who are visiting his website, Alex has decided to introduce a customer support chatbot on his website. He already has a dataset with him which can be used to train the chatbot but what he needs are some data preprocessing skills, which will be the first step towards chatbot creation.

Can you help him preprocess his dataset and create a Bag of Words encoding for the same ?

Project Template Output

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\ITRS-1795\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\ITRS-1795\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
stem_words list :  []
```

Expected Output

1. List of stemmed words from our dataset.

```
['all', 'ani', 'anyon', 'are', 'awesom', 'be', 'best', 'bluetooth', 'bye',
 'camera', 'can', 'chat', 'cool', 'could', 'digit', 'do', 'for', 'game', '
goodby', 'have', 'headphon', 'hello', 'help', 'hey', 'hi', 'hola', 'how',
'is', 'later', 'latest', 'me', 'most', 'next', 'nice', 'phone', 'pleas', '
popular', 'product', 'provid', 'see', 'sell', 'show', 'smartphon', 'tell',
 'thank', 'that', 'the', 'there', 'till', 'time', 'to', 'trend', 'video',
'what', 'which', 'you', 'your']
```

2. Bag of Word encoding for our First pattern and First tag

```
first BOW encoding:  [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0]
first Label encoding:  [0 0 1 0 0 0 0 0 0 0]
```

*This is just for your reference. We expect you to apply your own creativity to the project.

Getting Started:

Community Link

Publish to Community

Edit Your Project

Last Submitted

Previous Submissions

23rd Feb 2024	Open Link
23rd Feb 2024	Open Link
13th Feb 2024	Open Link

Start Project

Submit Your Project

Learn how to submit your project

Paste your project URL

Submit Project

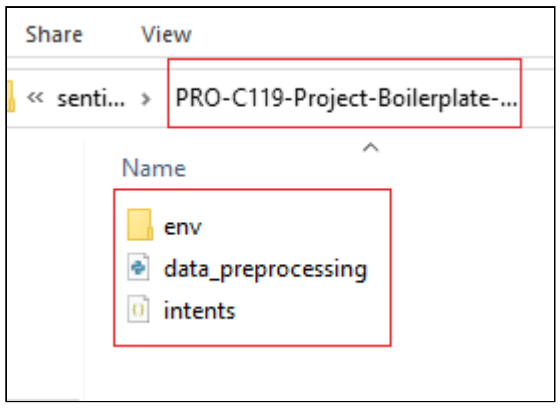
Class Summary

This project is based on your last class PRO-C119

View Class Summary

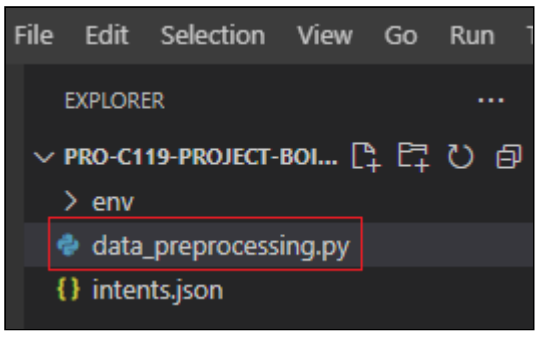
1. Open the Boilerplate link and download all the files within a new **folder** on your system.

2. Open the **command prompt**, traverse to that folder and create a python virtual environment inside it in such a way, so that the **virtual environment**, **intents.json and data_preprocessing.py** files are within the same folder

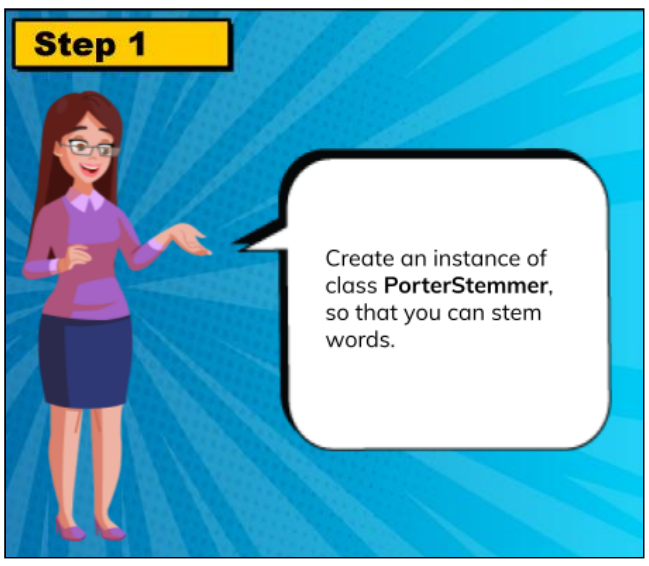


3. Activate the virtual environment and install the **nltk and Tensorflow** library in it, using **pip install nltk** and **pip install tensorflow==2.5.0.**

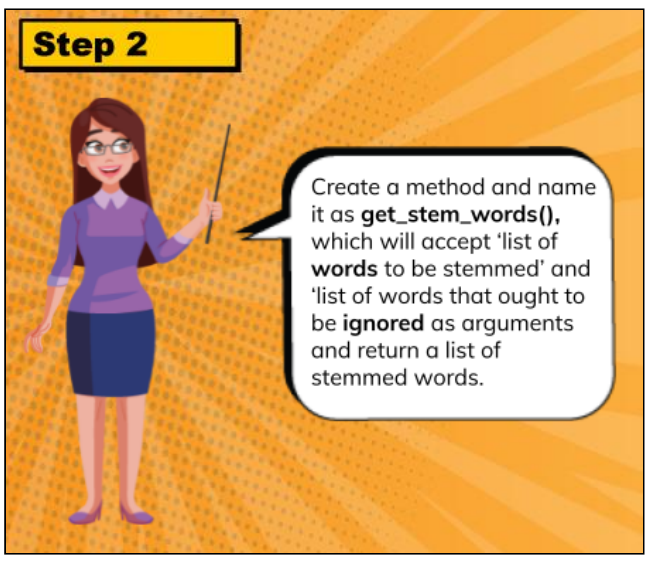
4. Open the **folder** in Visual Studio code, and click on the **data_preprocessing.py** file.



Specific Tasks to complete the Project:

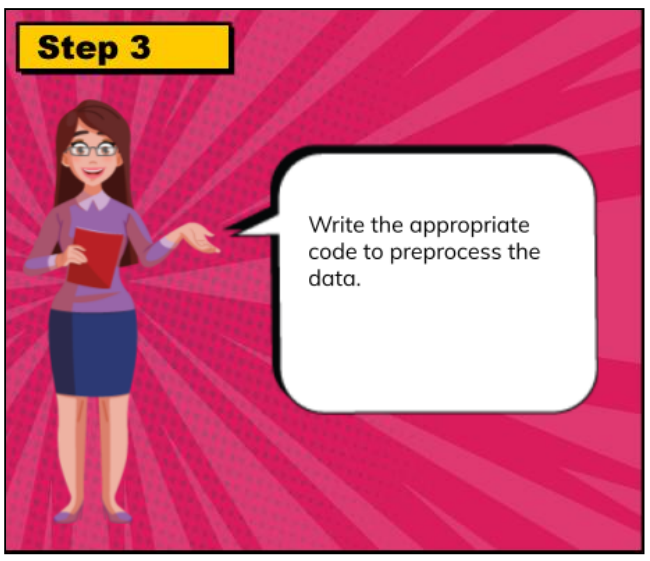



```
# create an instance of class PorterStemmer
```



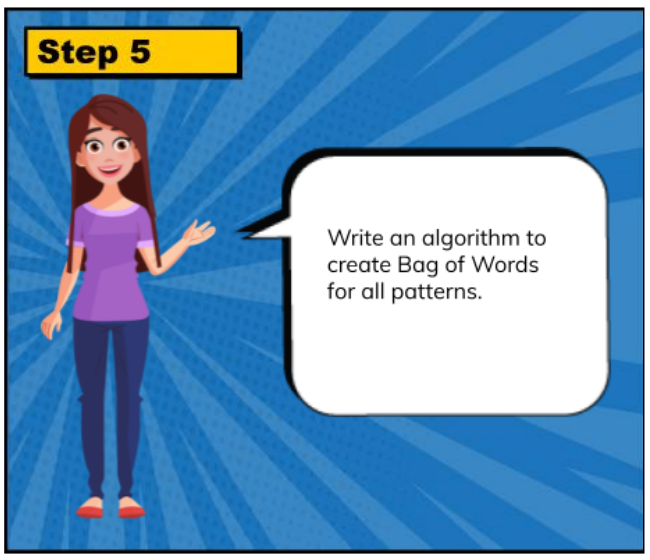
HELP

```python
# creating function to stem words
def get_stem_words(words, ignore_words):
    stem_words = []
    for word in words:

        # write stemming algorithm:
        '''
        Check if word is not a part of stop word:
        1) lowercase it
        2) stem it
        3) append it to stem_words list
        4) return the list
        '''
        # Add code here #

    return stem_words
```



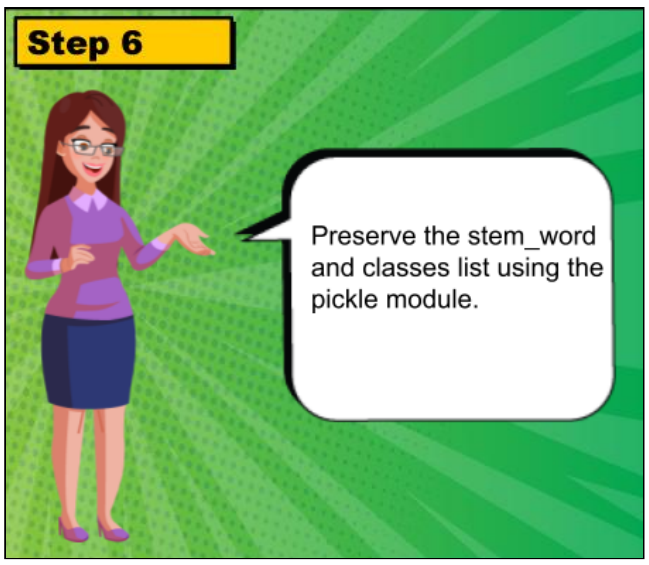

```python
# add the tokenized words to the words list

# add the 'tokenized word list' along with the 'tag' to pattern_word_tags_list
```


```python
# Remove duplicate words from stem_words

# sort the stem_words list and classes list
```




```python
# Input data encoding
'''
Write BOW algo :
1) take a word from stem_words list
2) check if that word is in stemmed_pattern_word
3) append 1 in BOW, otherwise append 0
'''

bag.append(bag_of_words)
```

HELP




```
# Convert Stem words and Classes to Python pickel file format
```

Submitting the Project:

1. **SAVE** all the changes made to the project.
2. Click on **"Run"** once to check if it is working.
3. Open GitHub and create a repository named **Project119**.
4. Upload files and click **Commit Changes**.
5. Copy the link and submit it in the Student Dashboard Projects panel against the correct class number.

Hints:

1. To run the program, activate the virtual environment, traverse to the folder create and use the command **python data_preprocessing.py [Step 7]**
2. In step 3, after tokenizing the patterns, add the tokenized words to the **words list** and the tokenized word to the pattern_word_tags _list along with their tags, using suitable list methods.
3. In step 4, sort both the list of **stem_words** and **classes**, using the appropriate list methods.

HELP