**Spell correction model**

import nltk

from nltk.corpus import words

from nltk.metrics.distance import edit\_distance

nltk.download('words')

class SpellCorrector:

def \_\_init\_\_(self):

self.word\_set = set(words.words())

def correct\_spelling(self, input\_word):

if input\_word.lower() in self.word\_set:

return input\_word # The word is correctly spelled

suggestions = self.get\_suggestions(input\_word)

if suggestions:

return min(suggestions, key=lambda x: edit\_distance(input\_word, x))

return input\_word # No suggestions found, return the original word

def get\_suggestions(self, input\_word):

# Generate suggestions based on edit distance

return [word for word in self.word\_set if edit\_distance(input\_word, word) <= 2]

if \_\_name\_\_ == "\_\_main\_\_":

spell\_corrector = SpellCorrector()

while True:

user\_input = input("Enter a word (type 'exit' to quit): ")

if user\_input.lower() == 'exit':

break

corrected\_word = spell\_corrector.correct\_spelling(user\_input)

print(f"Corrected word: {corrected\_word}")

**OUTPUT :**

C:\Users\civilsys51\PycharmProjects\pythonProject\.venv\Scripts\python.exe C:\Users\civilsys51\PycharmProjects\pythonProject\SpellCorrection.py

[nltk\_data] Downloading package words to

[nltk\_data] C:\Users\civilsys51\AppData\Roaming\nltk\_data...

[nltk\_data] Unzipping corpora\words.zip.

Enter a word (type 'exit' to quit): bt

eCorrected word: bit

Enter a word (type 'exit' to quit): exit

Process finished with exit code 0

**RESULT :**

Thus, we implement spell correction model using Python successfully.