BIS 420 PROGRAMMING FOR DATA SCIENCE

PRAJAKTA POHARE

CHAPTER 13 EXERCISE 13.2

ILLINOIS STATE UNIVERSITY

Go to Project Gutenberg (http: // gutenberg. org) and download your favorite out-of-copyright book in plain text format. Modify your program from the previous exercise to read the book you downloaded, skip over the header information at the beginning of the file, and process the rest of the words as before. Then modify the program to count the total number of words in the book, and the number of times each word is used.

Print the number of different words used in the book. Compare different books by different authors, written in different eras. Which author uses the most extensive vocabulary?

```
import string


def load_book(book):
    with open('/Users/prajaktapohare/Library/CloudStorage/OneDrive-ILStateUniversity/BIS420/Week 13/book.txt', 'r', encoding='utf-8') as f:
        lines = f.readlines()



    start = 0
    end = len(lines)
    for i, line in enumerate(lines):
        if "*** START OF" in line:
            start = i + 1
        elif "*** END OF" in line:
            end = i
            break


    return ''.join(lines[start:end])
```

```
def clean_text(text):

    translator = str.maketrans('', '', string.punctuation)
    return text.translate(translator).lower()

def count_words(text):
    words = text.split()
    word_count = {}
    for word in words:
        word_count[word] = word_count.get(word, 0) + 1
    return word_count

def analyze_book(book):
    raw_text = load_book(book)
    cleaned_text = clean_text(raw_text)
    word_counts = count_words(cleaned_text)

    total_words = sum(word_counts.values())
    unique_words = len(word_counts)

    print(f"\nAnalysis of {book}:")
    print(f"Total words: {total_words}")
    print(f"Unique words: {unique_words}")
    return unique_words

books = [
```

```
    'pride_and_prejudice.txt',
    'moby_dick.txt',
    'dracula.txt',
]

vocab_sizes = {}
for book in books:
    vocab_sizes[book] = analyze_book(book)

print("\nVocabulary comparison:")
for book, size in vocab_sizes.items():
    print(f"{book}: {size} unique words")

most_extensive = max(vocab_sizes, key=vocab_sizes.get)
print(f"\nMost extensive vocabulary: {most_extensive}")
```

```python
import string

def load_book(book):
    with open('/Users/prajaktapohare/Library/CloudStorage/OneDrive-ILStateUniversity/BIS420/Week 13/book.txt', 'r', encoding='utf-8') as f:
        lines = f.readlines()


    start = 0
    end = len(lines)
    for i, line in enumerate(lines):
        if "*** START OF" in line:
            start = i + 1
        elif "*** END OF" in line:
            end = i
            break

    return ''.join(lines[start:end])

def clean_text(text):

    translator = str.maketrans('', '', string.punctuation)
    return text.translate(translator).lower()

def count_words(text):
    words = text.split()
    word_count = {}
    for word in words:
        word_count[word] = word_count.get(word, 0) + 1
    return word_count

def analyze_book(book):
    raw_text = load_book(book)
    cleaned_text = clean_text(raw_text)
    word_counts = count_words(cleaned_text)

    total_words = sum(word_counts.values())
    unique_words = len(word_counts)

    print(f"\nAnalysis of {book}:")
    print(f"Total words: {total_words}")
    print(f"Unique words: {unique_words}")
    return unique_words


books = [
    'pride_and_prejudice.txt',
    'moby_dick.txt',
    'dracula.txt',
]

vocab_sizes = {}
for book in books:
    vocab_sizes[book] = analyze_book(book)

print("\nVocabulary comparison:")
for book, size in vocab_sizes.items():
    print(f"{book}: {size} unique words")

most_extensive = max(vocab_sizes, key=vocab_sizes.get)
print(f"\nMost extensive vocabulary: {most_extensive}")
```