# 📚 Word Frequency Analysis Library – lib.py

This Python module provides a reusable library for reading, cleaning, tokenizing, and analyzing textual review data from a JSON file. It is designed for projects involving **word frequency analysis**, **natural language processing (NLP)**, or **text preprocessing**, especially when dealing with user reviews or comments.

Date:  March 3, 2025 to March 27, 2025

## 🔧 Features

* ✅ Load and validate JSON datasets
* ✅ Tokenize and clean raw text
* ✅ Count word frequencies per review
* ✅ Handle lowercase normalization
* ✅ Save processed results to JSON

## 📁 Example JSON Input Format

[  
 {  
 "comment": "This product is excellent! Really satisfied."  
 },  
 {  
 "comment": "Poor quality and bad customer service."  
 }  
]

## 🧠 Main Functions

### read\_json\_file(path)

* **Purpose**: Loads a JSON dataset from the specified file path.
* **Raises**: Exception if the path is missing or the file is invalid.

### validate\_data(dataset)

* **Purpose**: Checks if all items in the dataset are dictionaries with a "comment" key.
* **Raises**: ValueError for any invalid entries.

### tokenize(text)

* **Purpose**: Cleans a string (removes punctuation, lowercases) and splits it into words.
* **Input**: A string of text.
* **Returns**: A list of words.

### tokenize\_review(review)

* **Purpose**: Adds a "words" key to the review dictionary with tokenized words.
* **Requires**: "comment" key in the dictionary.

### tokenize\_dataset(dataset)

* **Purpose**: Applies tokenize\_review() to each review in the dataset.
* **Input**: A list of review dictionaries.
* **Returns**: Tokenized dataset.

### compute\_word\_frequencies(words)

* **Purpose**: Computes frequency of each word in the list (case-insensitive).
* **Returns**: A Counter dictionary with word frequencies.

### compute\_word\_count\_review(review)

* **Purpose**: Adds a "word\_count" key to a review based on its "words".

### compute\_word\_count\_dataset(dataset)

* **Purpose**: Applies compute\_word\_count\_review() to all reviews.

### save\_results(data, file\_path)

* **Purpose**: Saves the final processed dataset into a JSON file.

## 🧪 Sample Workflow

from lib import \*  
  
# Load and validate data  
dataset = read\_json\_file("reviews.json")  
validate\_data(dataset)  
  
# Tokenize  
tokenized = tokenize\_dataset(dataset)  
  
# Compute word counts  
word\_counted = compute\_word\_count\_dataset(tokenized)  
  
# Save results  
save\_results(word\_counted, "output.json")

## 💡 Notes

* Assumes reviews are under the key "comment".
* Normalizes all words to lowercase.
* Removes punctuation using regex: [^\w\s].