Teammate 1 Brenda Leyva: (bleyva3):
Teammate 2 David Cardenas (dcard4):

Project files & functions:

Functions:

1. essayLength(essay)
    - This function assesses the length of an essay based on the number of complex sentences it contains. It will then return feedback about the essay's length and a score ranging from 0 to 100, depending on the count of complex sentences.
2. misspelledWords(essay, initial_score):
    - This function checks for spelling errors in the essay. It uses a spell checker (imported from SpellChecker) to identify words that are not recognized as correctly spelled words. It will then return a list of misspelled words and adjust the score by deducting points for each misspelled word from the initial score provided.
3. checkGrammar(essay, initial_score):
    - This function checks for grammatical errors in the essay. It uses a grammar checking tool to find issues and lists them. It will then return details of the grammar issues found and recalculate the score by deducting points for each grammar issue from the initial score.
4. finalGrade(grades):
    - This function calculates the final grade of the essay. It averages the scores obtained from the length, spelling, and grammar function.
5. processEssay(essay_text):
    - This function conducts a comprehensive analysis of an essay by evaluating its length, spelling, and grammar. Initially, it assesses the essay's length by counting the number of complex sentences and provides feedback and a score (from 0 to 100) based on this count. Next, it checks for misspelled words, adjusts the score accordingly, and lists the misspelled words found. Following that, it evaluates the grammar of the essay, further adjusting the score based on identified grammar issues, and lists these issues. Finally, the function calculates the final score for the essay, categorizes the overall quality based on this score, and prints a detailed report of the analysis, including feedback on length, scores for complexity, spelling, grammar, and the final assessment.
6. guess_topic
    - The guess_topic function works by first tokenizing the essay into individual words and then filtering out common stopwords that do not contribute significantly to the meaning. It then calculates the frequency of each remaining word, identifying those that appear most frequently as potential indicators of the main topics. The function finally returns a list of these most common words, which are assumed to be the key topics of the essay.

Files:

1. run_project

    a. This Python script allows users to input their essay via the command line by pasting the entire content and marking the end of their input with 'END' on a new line. After collecting the input, the script combines all lines, maintaining the original paragraph structure, into a single text string. The essay text is then processed using a function from the essay_analysis module, which evaluates the essay based on predefined criteria such as length, spelling, and grammar.
2. processEssay
    a. The processEssay function in the script s evaluates an essay's quality by assessing its length, spelling, and grammar, each impacting the overall score. After processing these elements, it provides detailed feedback on each aspect and computes a final score, categorizing the essay's performance as either "high" or "low". This analysis is then summarized and presented in the console, giving a clear overview of the essay's strengths and areas for improvement.
3. scoring
    a. The Python script provides a suite of functions designed to evaluate various aspects of essay writing, utilizing NLP tools and other libraries. It analyzes essay length by counting complex sentences, then identifies spelling errors with a spellchecker, and finally checks grammar with a dedicated tool, adjusting the essay's score based on the findings in each step. The overall assessment culminates in a final graded score, categorizing the essay's quality as either "high" or "low" based on the aggregated results from the length, spelling, and grammar evaluations.

Packages used:

- SpaCY
- SpellChecker
- Language_tool_python
- nltk