Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Semester: (Fall, Year: 2022), B.Sc. in CSE (Day)

Spell Checker

Course Title: Artificial Intelligence Lab
Course Code: CSE-316
Section: 213-D5

Students Details

Name	**ID**
Ahshanul Haquc	213002122
Mehedi Hasan Lemon	213002107

Submission Date: 6-05-2024
Course Teacher's Name: Sabah Nakshi

[For teachers use only: Don't write anything inside this box]

Lab Project Status

Marks:	**Signature:**
Comments:	**Date:**

Contents

Chapter 1

Introduction

1.1 Overview

The project aims to develop a web-based Grammar and Spell Checker tool utilizing natural language processing techniques. It provides users with a simple interface to input text or upload a file, which is then processed to identify and correct spelling mistakes and grammatical errors.

The system is built using the Flask web framework in Python, making it easy to deploy and interact with via a web browser. It incorporates two main components: the frontend interface developed using HTML and Bootstrap for styling, and the backend logic implemented in Python.

For spell checking, the system utilizes the TextBlob library, which employs machine learning algorithms to suggest corrections for misspelled words. Additionally, grammar checking is performed using the LanguageTool library, which applies rule-based analysis to detect and suggest corrections for grammatical errors.

1.2 Motivation

1. **Enhancing Communication:** Improving the correctness of written text helps to convey ideas more clearly and effectively.
2. **Education:** Users can learn from their mistakes by understanding the corrections suggested by the tool.
3. **Professionalism:** Ensuring proper spelling and grammar enhances the professionalism of documents, emails, and other written communications.
4. **Time-Saving:** Automated spell and grammar checking can save time compared to manual proofreading.

1.3 Problem Definition

1.3.1 Problem Statement

The motivation behind addressing this problem lies in the recognition of the significant impact that spelling and grammar errors can have on the clarity, credibility, and professional presentation of written text. Whether it's a business report, an academic paper, or a social media post, inaccuracies in spelling and grammar can detract from the intended message and undermine the author's authority and expertise.

Moreover, as digital communication becomes increasingly prevalent in both personal and professional spheres, the need for efficient spell checking and grammar correction tools becomes more pronounced. Users require solutions that are not only accurate and reliable but also convenient and user-friendly, enabling them to quickly and easily correct errors and enhance the overall quality of their written content.

Therefore, the problem addressed by this project is to develop a web-based Grammar and Spell Checker tool that effectively identifies and corrects spelling and grammar errors in written text, thereby improving the clarity, professionalism, and impact of written communication across various contexts.

1.3.2 Complex Engineering Problem

Table 1.1: Summary of the attributes touched by the mentioned projects

P1: Depth of knowledge required	Moderate depth of knowledge is required in natural language processing (NLP) techniques to understand and implement spell checking and grammar correction algorithms. However, the project aims to provide a user-friendly interface, reducing the depth of knowledge required for end-users.
P2: Range of conflicting requirements	The project must balance between accuracy and speed in spell and grammar checking. Accuracy is crucial to ensure correct corrections, while speed is necessary for real-time usage. Various NLP algorithms need to be evaluated to strike the right balance.
P3: Depth of analysis required	The project requires a moderate level of depth in analyzing language structures and patterns to effectively identify and correct spelling and grammar errors. This involves understanding linguistic rules and patterns and implementing algorithms to analyze and correct text.
P4: Familiarity of issues	The project addresses common issues encountered in written communication, such as spelling mistakes and grammatical errors. These issues are familiar to most users, making the project relevant and applicable across various contexts.
P5: Extent of applicable codes	The project utilizes existing libraries and frameworks for natural language processing, such as TextBlob and LanguageTool, reducing the need for extensive code development.
P6: Extent of stakeholder involvement and conflicting requirements	Stakeholder involvement may include feedback from end-users to improve the accuracy and usability of the spell checker and grammar correction tool.
P7: Interdependence	The project's success depends on the interdependence of various components, including the frontend interface, backend logic, and natural language processing algorithms. Changes or updates to one component may affect the functionality and performance of other components, requiring careful coordination and testing.

1.4 Objectives

1. **User-Friendly Web Application Design**: Create an intuitive and accessible web interface. Include text input fields and file upload functionality. Design a visually appealing layout for ease of use. Ensure responsiveness across different devices.
2. **Spell Checking and Grammar Correction**: Implement accurate spell checking algorithms. Integrate grammar correction functionalities. Provide real-time feedback on errors. Utilize reliable language processing libraries or APIs.
3. **Usability and Accessibility Enhancement**: Offer informative tooltips or error messages. Enable keyboard shortcuts for quick navigation. Ensure compatibility with screen readers. Implement undo/redo feature for mistake correction.

1.5 Application

In this section, we delve into the diverse applications of the Grammar and Spell Checker in various real-world scenarios.

1.5.1 Education Sector

In educational settings, the Grammar and Spell Checker plays a crucial role for both students and educators:

Student Writing Assignments

Students can utilize the tool to review and revise their essays, reports, and research papers before submission, thereby improving their writing skills [1].

Teacher Feedback and Grading

Educators benefit from streamlining the grading process by quickly identifying and correcting spelling and grammar errors in student submissions. This allows teachers to focus more on providing substantive feedback on content and structure.

1.5.2 Business and Professional Writing

Clear and polished written communication is vital in professional environments:

Business Documents

Professionals rely on the Grammar and Spell Checker to ensure the accuracy and professionalism of business documents such as emails, memos, proposals, and presentations, thus enhancing the organization's reputation and credibility.

Marketing Content

Marketing teams utilize the tool to refine marketing materials, including website copy, social media posts, advertisements, and press releases, ensuring consistency and accuracy in messaging across various channels.

1.5.3 Content Creation and Publishing

Content creators across industries leverage the Grammar and Spell Checker to enhance the quality and readability of their written content:

Blogging and Content Marketing

Bloggers and content creators use the tool to proofread blog posts, articles, and other online content before publishing, ensuring effective communication with their audience.

1.5.4 Language Learning and ESL Support

For language learners and non-native English speakers, the Grammar and Spell Checker serves as a valuable learning tool:

Language Learning Exercises

Students learning English as a second language (ESL) practice writing skills and receive instant feedback on errors, aiding in their language acquisition process.

ESL Teaching and Tutoring

Educators and tutors working with ESL learners provide targeted support and instruction using the tool, helping students improve their writing proficiency and confidence in English language usage.

Chapter 2

Design/Development/Implementation of the Project

2.1 Introduction

Start the section with a general discussion of the project [2] [3] [4].

2.2 Project Details

In this section, you will elaborate on all the details of your project, using subsections if necessary.

2.2.1 Subsection_name

Figure 2.1: Figure name

You can fix the height, width, position, etc., of the figure accordingly.

2.3 Implementation

All the implementation details of your project should be included in this section, along with many subsections.

2.3.1 Subsection_name

This is just a sample subsection. Subsections should be written in detail. Subsections may include the following, in addition to others from your own project.

The workflow

Tools and libraries

Implementation details (with screenshots and programming codes)

Each subsection may also include subsubsections.

2.4 Algorithms

The algorithms and the programming codes in detail should be included .
Pseudo-codes are also encouraged very much to be included in this chapter for your project.

- Bullet points can also be included anywhere in this project report.

Algorithm 1: Sample Algorithm

```
Input: Your Input
Output: Your output
Data: Testing set $x$
$\sum_{i=1}^{\infty} := 0$                                  // this is a comment
/* Now this is an if...else conditional loop                   */
if Condition 1 then
    Do something                                      // this is another comment
    if sub-Condition then
        Do a lot
else if Condition 2 then
    Do Otherwise
    /* Now this is a for loop                                  */
    for sequence do
        loop instructions
else
    Do the rest
/* Now this is a While loop                                    */
while Condition do
    Do something
```

Chapter 3

Performance Evaluation

3.1 Simulation Environment/ Simulation Procedure

Discuss the experimental setup and environment installation needed for the simulation of your outcomes.

3.1.1 Subsection

3.1.2 Subsection

3.2 Results Analysis/Testing

Discussion about your various results should be included in this chapter in detail.

3.2.1 Result_portion_1

The results of any specific part of your project can be included using subsections.

3.2.2 Result_portion_2

Each result must include screenshots from your project. In addition to screenshots, graphs should be added accordingly to your project.

3.2.3 Result_portion_3

Each result must have a single paragraph describing your result screenshots or graphs or others. This is a simple discussion of that particular portion/part of your result.

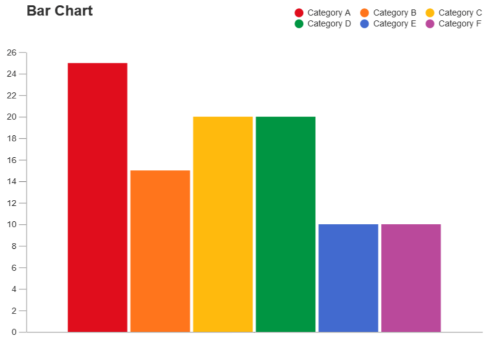

Figure 3.1: A graphical result of your project

3.3 Results Overall Discussion

A general discussion about how your result has arrived should be included in this chapter. Where the problems detected from your results should be included as well.

3.3.1 Complex Engineering Problem Discussion

[OPTIONAL] In this subsection, if you want, you can discuss in details the attributes that have been touched by your project problem in details. This has already been mentioned in the Table 1.1.

Chapter 4

Conclusion

4.1 Discussion

Discuss the contents of this chapter and summarized the description of the work and the results and observation. Generally, it should be in one paragraph.

4.2 Limitations

Discuss the limitations of the project. Limitations must be discussed, with the help of some critical analysis.

4.3 Scope of Future Work

Discuss the future work of the project, that is your plans for more work and extension of your project.

References

[1] Omid C Farokhzad and Robert Langer. Impact of nanotechnology on drug delivery. *ACS nano*, 3(1):16–20, 2009.

[2] Uthayasankar Sivarajah, Muhammad Mustafa Kamal, Zahir Irani, and Vishanth Weerakkody. Critical analysis of big data challenges and analytical methods. *Journal of Business Research*, 70:263–286, 2017.

[3] Douglas Laney. 3d data management: controlling data volume, velocity and variety. gartner, 2001.

[4] MS Windows NT kernel description. http://web.archive.org/web/20080207010024/http://www.808multimedia.com/winnt/kernel.htm. Accessed Date: 2010-09-30.