**A PRELIMINARY REPORT**

**ON**

**“Evaluating Subjective Answers with Machine Learning and Natural Language Processing”**

SUBMITTED TO THE DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY LONERE

IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

**BACHELOR OF TECHNOLOGY**

##### SUBMITTED BY

STUDENT NAME EXAM SEAT NO:

**PRASAD MANE 2030331245033**

**SAURABH CHAWARE 2030331245047**



## DEPARTMENT OF COMPUTER ENGINEERING

## DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY

LONERE

**2023 -2024**



**CERTIFICATE**

This is to certify that the project report entitled

**“Evaluating Subjective Answers with Machine Learning and Natural Language Processing”**

Submitted by

STUDENT NAME EXAM SEAT NO

**PRASAD MANE 2030331245033**

**SAURABH CHAWARE 2030331245047**

are bonafide student of this institute and the work has been carried out by them under the supervision of Mr. SANIL GANDHIand it is approved for the partial fulfillment of the requirement of Dr. Babasaheb Ambedkar Technological University, for the award of the degree of **Bachelor of Technology**

**Mr. SANIL GANDHI** **Dr. ARVIND KIVELEKAR**

Guide Head

Department of Computer Engineering Department of CS

Place: DBATU, Lonere

Date: 12/12/2023

**ACKNOWLEDGEMENT**

It gives us great pleasure in presenting the preliminary project report on ‘Evaluating Subjective Answers with Machine Learning and Natural Language Processing’.

We would like to take this opportunity to thank my internal guide **Mr. SANIL GANDHI,** for giving me all the help and guidance we needed. We are really grateful to them for their kind support. Their valuable suggestions were very helpful.

We are also grateful to **Dr ARVIND KIWELEKAR**, Head of Computer Engineering Department, Dr. Babasaheb Ambedkar Technological University, Lonere, for his indispensable support, suggestions.

**PRASAD MANE**

**SAURABH CHAWARE**

**ABSTRACT**

Subjective paper evaluation can be quite a tricky and tiresome task when done manually. One of the biggest challenges in analyzing subjective papers through Artificial Intelligence (AI) is the lack of comprehensive understanding and acceptance of data. While there have been efforts to use AI to score students' answers, many of these approaches rely on basic word counts or specific keywords. Moreover, the availability of curated datasets for training such AI models is limited. Proposed methodology involves using a Large Language Model (LLM) to generate answers to questions. This LLM will have access to a relevant textbook, from which it will derive the answer to a given question. The evaluation of a student's answer will be based on comparing embeddings of the student's answer and the LLM-generated answer using cosine similarity. The resulting similarity score will determine the quality of the student's response. This modified approach leverages the capabilities of language models and aligns them with a reference textbook, which should provide a structured and accurate method for assessing student answers. To do all of the above, OpenAI langchain would be used.

**Keywords:**

* Subjective answer assessment
* Artificial Intelligence (AI)
* Large Language Model (LLM)
* OpenAI langchain
* Textbook
* Cosine Similarity

TABLE OF CONTENTS

1. LIST OF ABBREVATIONS
2. LIST OF FIGURES

# CHAPTER TITLE PAGE NO.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr.No. | | | Title of Chapter | | Page No. |
| 01 | | | **Introduction** | | 8 |
|  | 1.1 | | Overview | | 8 |
|  | 1.2 | | Motivation | | 8 |
|  | 1.3 | | Problem Definition and Objectives | | 10 |
|  | 1.4 | | Project Scope | | 10 |
| 02 | | | Literature Survey | | 11 |
| 03 | | | Software Requirements Specification | | 13 |
|  | | 3.1 | Assumptions and Dependencies | | 13 |
|  | | 3.2 | Functional Requirements | | 13 |
|  | |  | 3.2.1 | System Feature 1(Functional Requirement) |  |
|  | |  | 3.2.2 | System Feature2 (Functional Requirement) |  |
|  | |  |  | …… |  |
|  | |  |  | …… |  |
|  | | 3.3 | External Interface Requirements (If Any) | | 14 |
|  | |  | 3.3.1 | User Interfaces |  |
|  | |  | 3.3.2 | Hardware Interfaces |  |
|  | |  | 3.3.3 | Software Interfaces |  |
|  | | 3.4 | Nonfunctional Requirements | | 15 |
|  | |  | 3.4.1 | Performance Requirements |  |
|  | |  | 3.4.2 | Safety Requirements |  |
|  | |  | 3.4.3 | Security Requirements |  |
|  | |  | 3.4.4 | Software Quality Attributes |  |
|  | | 3.5 | **System Requirements** | | 16 |
|  | |  | 3.5.1 | Software Requirements(Platform Choice) |  |
|  | |  | 3.5.2 | Hardware Requirements |  |
|  | | 3.6 | Analysis Models: SDLC Model to be applied | | 16 |
|  | | 3.7 | System Implementation Plan | | 17 |
| 04 | | | System Design | | 19 |
|  | | 4.1 | System Architecture | |  |
|  | | 4.2 | Entity Relationship Diagrams | |  |
|  | | 4.3 | UML Diagrams | |  |
| 05 | | | Other Specification | | 21 |
|  | | 5.1 | Advantages | |  |
|  | | 5.2 | Limitations | |  |
| 06 | | | **Conclusions & Future Work** 22  6.1 Conclusion  6.2 Future Scope  6.3 References | | | |  | |
|  | | |  | | | | |  | |
|  | | |  | |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure** | **Illustration** | **Page No.** |
|  |  |  |
| 1 | Waterfall Model | 21 |
| 2 | System Architecture |  |
| 4 | ER Diagram |  |
| 5 | UML Diagram | 27 |

1. **INTRODUCTION**
   1. **OVERVIEW**

The assessment of subjective papers is a challenging and time-consuming task when carried out manually. The integration of AI into this process presents a promising solution, yet it faces obstacles related to data comprehension and the scarcity of suitable training datasets. Many existing AI approaches rely on basic word counts or specific keywords, limiting their effectiveness. The proposed methodology adopts LLMs, utilizing a relevant textbook as a knowledge source for generating answers to questions. The evaluation of a student's response involves comparing embeddings of the student's answer and the LLM-generated answer using cosine similarity, determining the quality of the response. This approach leverages advanced language models and aligns them with a reference textbook, promising a more structured and accurate method for assessing student answers. The integration of OpenAI langchain further enhances the capabilities of this methodology, potentially revolutionizing subjective paper evaluation.

* 1. **MOTIVATION**

In an era marked by digitalization and globalization, the educational sector confronted a multitude of challenges. Teachers found themselves grappling with growing expectations for delivering quality education, personalized learning experiences, and a diverse array of assessment methods. However, a significant portion of their responsibilities proved to be tedious, time-consuming, and susceptible to human errors. Notably, the laborious task of comparing students' answers with correct solutions, especially in the context of large-scale or open-ended tests, was a recurring challenge.

As depicted in the figure below, the number of test takers for standardized exams like the SAT and ACT continued to rise steadily. This trend was mirrored across various tests. The reliance on human evaluators became increasingly impractical due to the growing volume of assessments, contributing to a time-intensive process.

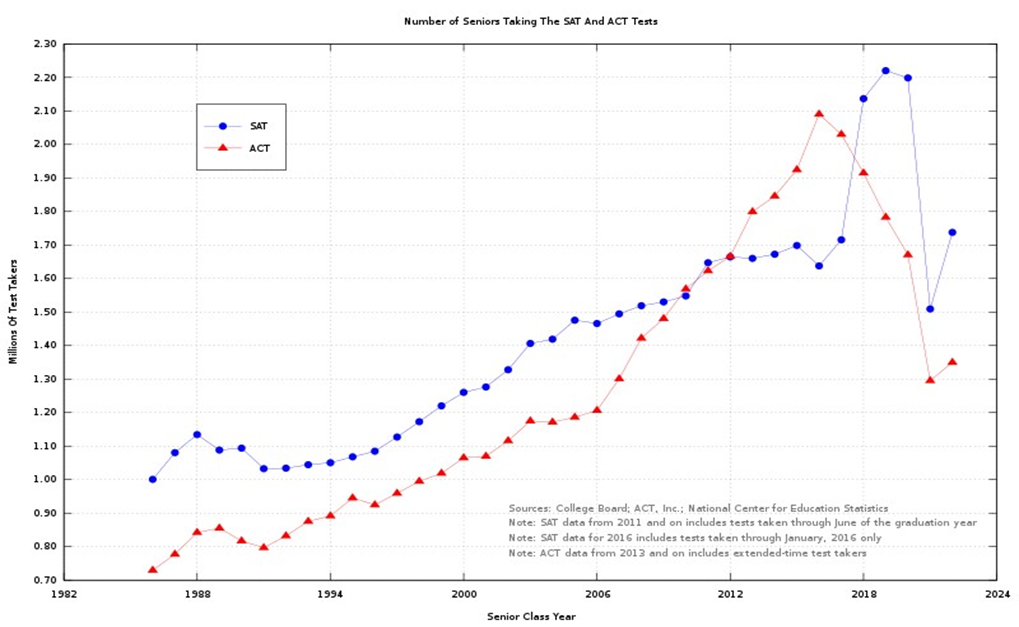


Figure : asddffg

To address these pressing issues, it became imperative to explore innovative solutions. The introduction of machine-based evaluation represented a significant leap forward for the education sector. Leveraging natural language processing and machine learning techniques, these systems could automatically assess the semantic similarity between texts and assign scores based on predefined criteria. This transformative shift relieved educators from the burdens of manual labor in routine tasks such as answer comparisons, enabling them to allocate more time to their core responsibilities.

Furthermore, machine-based evaluation significantly enhanced the quality and transparency of the assessment process. Machines provided consistent and objective feedback to both students and teachers, free from the influence of human biases or emotions. Additionally, they generated detailed reports and analytics, empowering educators to monitor student progress and performance. This, in turn, allowed for the identification of students' strengths and weaknesses, facilitating the design of more tailored curricula that catered to their specific needs and interests.

* 1. **PROBLEM STATEMENT AND OBJECTIVE**

**Problem Statement:**

To use LLM and AI agents to assign a score to answers.

**Objectives:**

* Automate answer sheet evaluation.
* Save teacher’s time.
* Focus on valuable aspects of education.
* Establish standardized evaluation criteria.
* Develop algorithms that can handle diverse reasoning patterns, linguistic styles, and language proficiency levels.
  1. **PROJECT SCOPE**

**Project Scope:**

The project's scope is to develop an advanced system for automating subjective paper evaluation, addressing the challenges of manual assessment. This system will utilize a LLM with access to a relevant textbook to generate answers to questions. The assessment process will involve comparing embeddings of student answers with LLM-generated answers using cosine similarity, resulting in a quality score for each response. The project aims to enhance the efficiency and accuracy of subjective paper evaluation in the education sector, leveraging AI and language models while ensuring alignment with a reference textbook. The implementation will incorporate OpenAI langchain for an integrated and comprehensive solution.

1. **LITERATURE SURVEY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.**  **No** | **Paper Title** | **Journal Name** | **Authors &**  **Publication Date** | **Methodology** |
| 1 | Subjective Answer Evaluation Using Machine Learning | International Journal of Pure and Applied Mathematics | Piyush Patil ,  Sachin Patil ,  Vaibhav Miniyar ,  Amol Bandal  23 May 2018 | The methodology involves using machine learning and NLP to evaluate subjective answers by:  1. Tokenizing text.  2. Part of Speech tagging.  3. Chunking, Chinking.  4. Lemmatizing words.  5. Wordnetting for evaluation.  6. Providing semantic context meaning.  The system has two modules: data extraction from images and applying ML and NLP for grading. |
| 2 | Subjective Answers Evaluation Using Machine Learning and Natural Language Processing | IEEE Xplore | Muhammad Farrukh Bashir,  Hamza Arshad,  Abdul Rehman Javed,  Natalia Kryvinska, Shahab Band.  25 November 2021 | This paper introduces an innovative approach to automatically evaluate descriptive answers in subjective papers. It utilizes a range of techniques, including machine learning and NLP tools such as Wordnet, Word2vec, WMD, cosine similarity, MNB, and TF-IDF. The system uses solution statements and keywords to grade answers, with WMD outperforming cosine similarity. It achieves an 88% accuracy without MNB, further reducing the error rate by 1.3% with MNB. |
| 3 | Evaluating Descriptive Answer Using Machine Learning And Natural Language Processing | Emperor Journal of Applied Scientific Research | T. Sai Kumari,  Hemant Kumar ,  Shaik Ahmed,  Janvee Shree ,  Josna Sravani | Data Collection: Gather descriptive answer scripts and correct answers.  Similarity Assessment: Use NLP to measure similarity between student responses and correct answers, awarding higher marks for closer matches.  Automation Benefits:  1.Reduce manual grading effort.  2.Enable online examinations and instant evaluation.  3.Enhance efficiency for educational institutions. |
| 4 | NLP-based Automatic Answer Evaluation | IEEE Xplore | Shubham Kumar Sinha,  Sachin Yadav,  Bindu Verma,  2022 | * Data Collection: Gather answer scripts and correct answers. * Text Extraction: Extract text from student answer scripts. * Similarity Calculation: Compare student responses with correct answers, using various measures including Cosine similarity. * Weight Assignment: Assign weights to similarity measures. * Summarization: Generate a summary of the student's response using keyword-based algorithms. * Final Mark Calculation: Combine similarity scores and the summary to calculate the final grade for the script. * Evaluation: Validate the algorithm's accuracy by comparing automated grades with human-assigned marks. |
| 5 | An Approach to Evaluating Subjective Answers using BERT model | IEEE Xplore | Potsangbam Sushila Devi,  Sunita Sarkar,  Takhellambam Sonamani Singh,  Laimayum Dayal Sharma,  Chongtham Pankaj,  Khoirom Rajib Singh  2022 | Data Collection: Gather subject answers in Engineering and Medical fields.  Preprocessing: Clean and prepare the text data.  Word Embedding: Use BERT to convert text into vector representations.  Vector Space: Transform BERT-embedded sentences into vectors.  Similarity Calculation: Measure sentence similarity as percentages using methods like cosine similarity.  Evaluation: Assess the semantic similarity of answers based on similarity scores.  Application: Employ machine learning for fine-tuning and classification based on similarity scores. |

1. **SOFTWARE REQUIREMENT SPECIFICATION**

**3.1 INTRODUCTIO**

3.1.1 USE **CLASSES AND CHARACTERISTICS**

Our system is divided into two class/modules:

1) User

2) System

* + 1. **ASSUMPTIONS AND DEPENDENCIES**

1. Users must have the knowledge of web-based applications.
2. Users must have knowledge of English.
3. Users must have all required software to run the application.
   1. **FUNCTIONAL REQUIREMENTS**

* The system should efficiently evaluate subjective answers provided by students.
* It should generate answers to questions using a Large Language Model (LLM) with access to a relevant textbook.
* The system must compute embeddings for student answers and LLM-generated answers for further analysis.
* It should calculate cosine similarity scores between the embeddings and the LLM-generated answers to determine the quality of student responses.
* The software should provide consistent and objective feedback to both students and teachers, reducing the impact of human biases or emotions.
* Detailed reports and analytics should be generated to assist teachers in monitoring student progress and identifying strengths and weaknesses.
* The application should be designed in modular form to facilitate easy error detection and updates, ensuring scalability and maintainability.
* Users should have access to their own information, ensuring privacy and data security.
* These functional requirements aim to address the core aspects of automating subjective paper evaluation using AI, aligning with the problem statement's objectives.
  1. **EXTERNAL INTERFACE REQUIREMENTS**

1. **USER INTERFACES**

The requirements section of hardware includes a minimum of 180 GB hard disk and 4 GB RAM with 2 GHz or higher speed. The primary requirements include a memory of 4GB for the react and flask framework.

1. **HARDWARE INTERFACES**

As this is an online application for product management we are not enabling or installing any hardware components for user interface.

It’s not an embedded system

* + - Processor - Pentium IV 2.4 GHZ
    - Speed - 1.5 Ghz and Above
    - RAM - 4 GB (min)
    - Hard Disk - 220 GB
    - Key Board - Standard Windows Keyboard
    - Mouse - Two or Three Button Mouse

1. **SOFTWARE INTERFACES**

This is the software configuration in which the project was shaped. The programming language used, tools used, etc are described here.

* Operating System : Windows
* Front End : html, css,bootstrap,javascript,react.
* backend :Flask
* Tool : VisualStudio
* Database : MySQL

1. **COMMUNICATION INTERFACES**

* Users can access the web application from a remote location.
* Standard internet connection is required.
* TCP/UDP connection will be required.
  1. **NON-FUNCTIONAL REQUIREMENTS**

1. **PERFORMANCE REQUIREMENTS**

* High Speed:

System should process requested tasks in parallel for various actions to give a quick response. Then the system must wait for process completion.

* Accuracy: System should correctly execute the process, display the result accurately. System output should be in user required format.

1. **SAFETY REQUIREMENTS**

The data safety must be ensured by arranging for a secure and reliable transmission media. The source and destination information must be entered correctly to avoid any misuse or malfunctioning. Passwords generated by the user consist of characters, special characters & numbers so that password is difficult to hack. So, that user account is safe.

1. **SECURITY REQUIREMENTS**

Secure access of confidential data (user’s details).

* + - Information security means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction.
    - The terms information security, computer security and information assurance are frequently incorrectly used interchangeably. These fields are interrelated often and share the common goals of protecting the confidentiality, integrity and availability of information; however, there are some subtle differences between them.
    - User password must be stored in encrypted form for the security reason
    - All the user details shall be accessible to only high authority persons.
    - Access will be controlled with usernames and passwords.

1. **SOFTWARE QUALITY ATTRIBUTES**

* Availability [related to Reliability]
* Modifiability [includes portability, reusability, scalability]
* Performance
* Security & testability
* Usability[includes self-adaptability and user adaptability]
  1. **SYSTEM REQUIREMENTS**

1. **SOFTWARE REQUIREMENTS**

Operating system : Windows 7 and above.

Coding Language : Python,

Web Browser : Google Chrome, Firefox, Bing etc

1. **HARDWARE REQUIREMENTS**

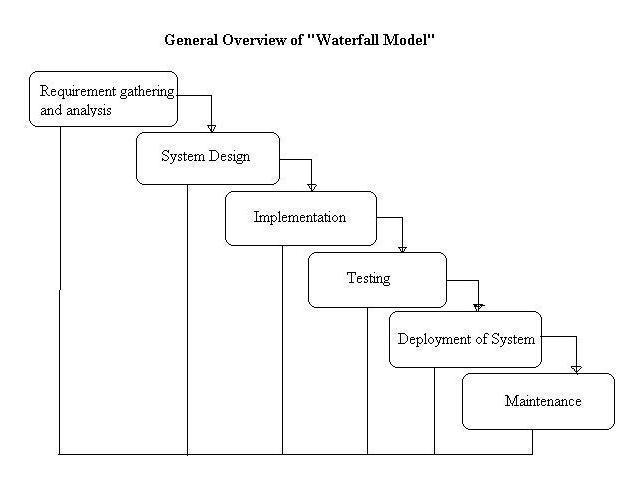
System :         Intel I3 Processor and above.

Hard Disk           : 200 GB.

Monitor : 15 VGA Color.

Ram : 4 GB.

* 1. **ANALYSIS MODELS : SDLC MODEL TO BE APPLIED**

****

**Figure 1 : Waterfall Model**

* 1. **SYSTEM IMPLEMENTATION PLAN**

**1. Requirement gathering and analysis:**

In this step of waterfall we identify what various requirements are needed for our project such as software and hardware required, database, and interfaces.

**2. System Design:**

In this system design phase we design the system which is easily understood for the end user i.e. user friendly.

We design some UML diagrams and data flow diagrams to understand the system flow and system module and sequence of execution.

**3. Implementation:**

In the implementation phase of our project we have implemented various modules required to successfully get expected outcomes at the different module levels.

With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.

**4. Testing:**

The different test cases are performed to test whether the project module are giving expected outcome in assumed time.

All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

**5. Deployment of System:**

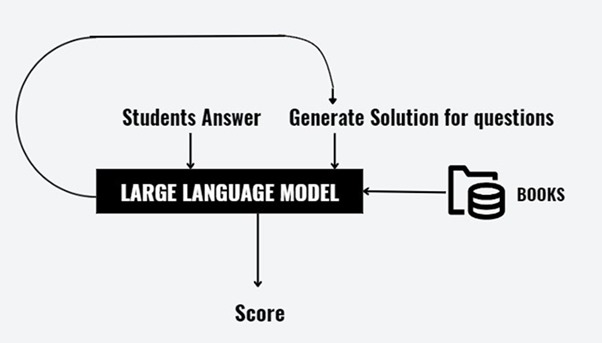
Once the functional and nonfunctional testing is done, the product is deployed in the customer environment or released into the market.

**6. Maintenance:**

There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

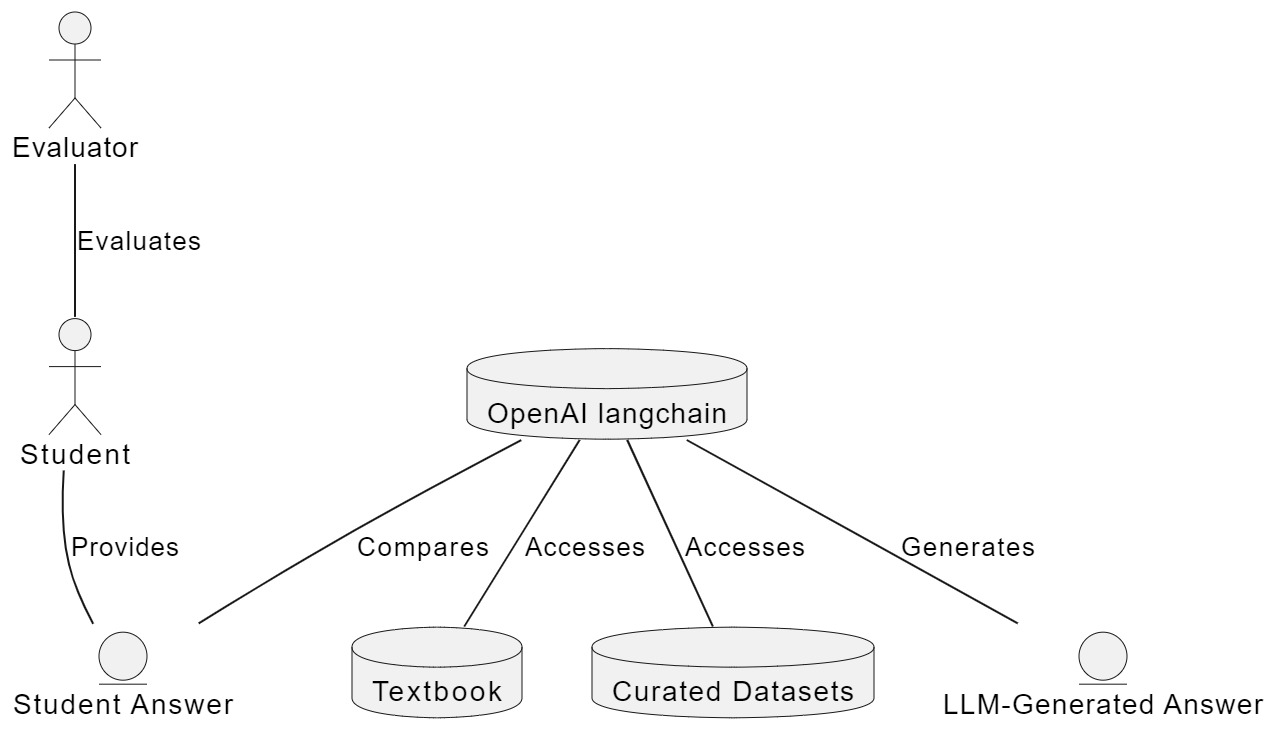
All these phases are cascaded to each other in which progress is seen as flowing steadily downwards like a waterfall through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

1. **SYSTEM DESIGN**
   1. **SYSTEM ARCHITECTURE**

****

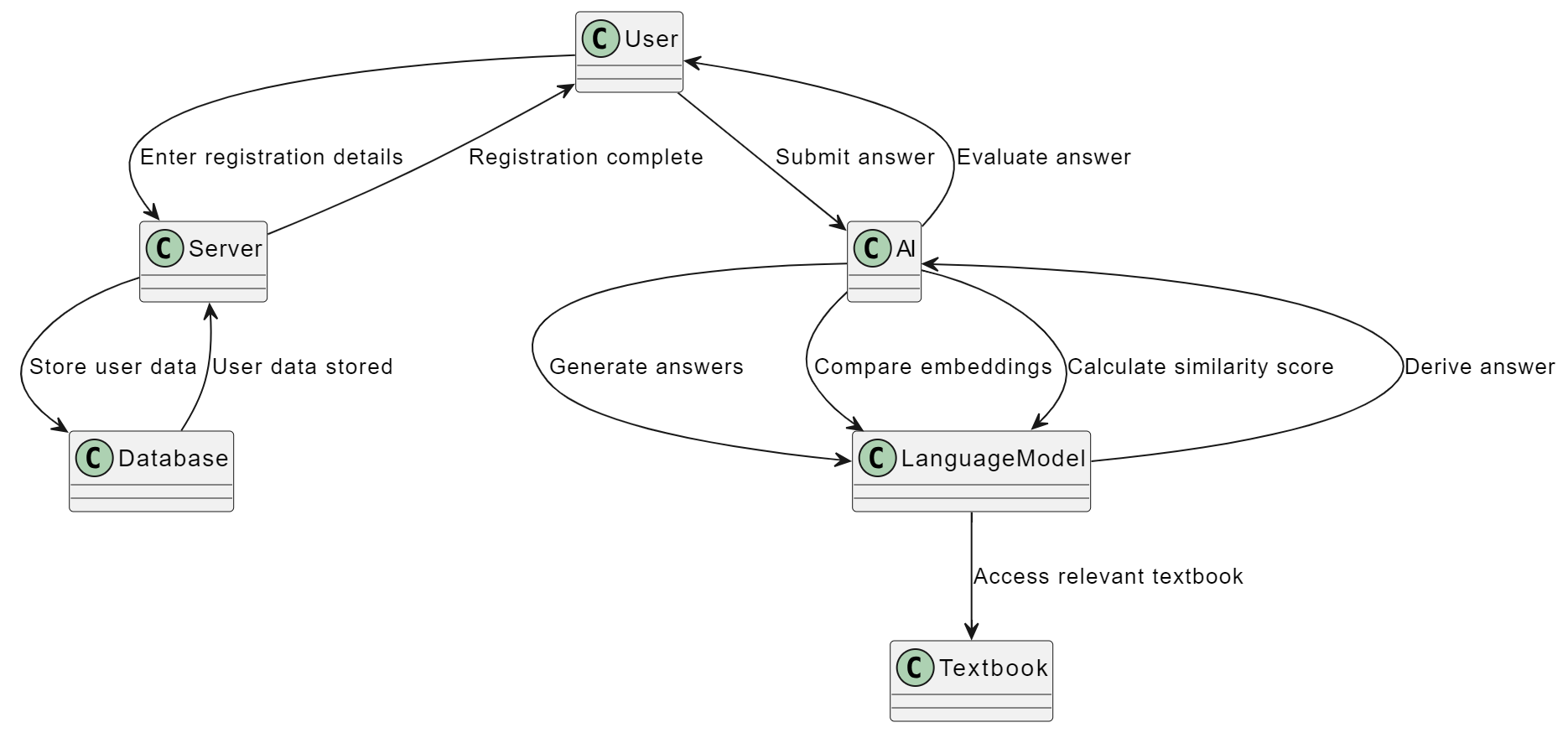
**Figure 1 : System Architecture**

* 1. **ENTITY RELATIONSHIP DIAGRAMS**



**Figure 1 : Entity Relationship Diagram**

* 1. **UML DIAGRAM**



**Figure 2 : UML Diagram**

1. **OTHER SPECIFICATION**

**5.1 ADVANTAGES**

* Automate answer sheet evaluation
* Save teachers’ time.
* Effective evaluation technique
* Confirming those infected is essential to manage and contain the virus successfully. Without reliable testing, it would be hard to determine the actual rates of cases. Thus, it is vital to identify what these available tests can and can’t do to use them appropriately**.**
* Secure and efficient system

**5.2 LIMITATIONS**

1. Proficiency in the English language is necessary.
2. Access to all the essential software required to run the application is mandatory.
3. A stable internet connection is a prerequisite.
4. It can be costly, often slow, challenging, and not consistently equitable.
5. Comparing and implementing it across various systems or models is not a straightforward task.
6. **CONCLUSION & FUTURE WORK**

**Conclusion**:

In summary, the proposed solution, as outlined in the problem statement, offers a transformative way to automate subjective paper evaluation using AI. This approach promises to reduce the manual workload on educators, enhance assessment quality, and empower a more personalized approach to student learning in the education sector.

**Future scope:**

* To develop an online learning platform that can automatically grade the subjective answers of students and provide them with instant feedback and suggestions.
* This can enhance the learning experience of students and help them improve their writing and reasoning skills.
* It can also reduce the workload of teachers and allow them to focus more on teaching and designing the curriculum.
* To develop a plagiarism detection system that can compare the similarity between different texts and identify the sources of copied or paraphrased content.
* This can help prevent academic dishonesty and promote originality and creativity among students and researchers. It can also help protect the intellectual property rights of authors and publishers.

**References:**

1. M. F. Bashir, H. Arshad, A. R. Javed, N. Kryvinska and S. S. Band, "Subjective Answers Evaluation Using Machine Learning and Natural Language Processing," in IEEE Access, vol. 9, pp. 158972-158983, 2021, doi: 10.1109/ACCESS.2021.3130902.
2. Kumari, T.S., Kumar, H., Ahmed, S., Sree, J. and Sravani, J., 2022. Evaluating Descriptive Answer Using Machine Learning And Natural Language Processing.
3. S. K. Sinha, S. Yadav and B. Verma, "NLP-based Automatic Answer Evaluation," 2022 6th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2022, pp. 807-811, doi: 10.1109/ICCMC53470.2022.9754052.
4. <https://docs.langchain.com/docs/>
5. P. S. Devi, S. Sarkar, T. S. Singh, L. D. Sharma, C. Pankaj and K. R. Singh, "An Approach to Evaluating Subjective Answers using BERT model," 2022 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bangalore, India, 2022, pp. 1-4, doi: 10.1109/CONECCT55679.2022.9865706.
6. Patil, P., Patil, S., Miniyar, V., & Bandal, A. (2018). Subjective Answer Evaluation Using Machine Learning. International Journal of Pure and Applied Mathematics, 23 May.