ISSN No:-2456-2165

Advance Assessment Evaluation: A Deep-Learning Framework with Sophisticated Text Extraction for Unparalleled Precision

Tanishq Jaiswal[1], Varsha Teeratipally[2], Ritendu Bhattacharyya[3], Bharani Kumar Depuru[4]
[1,2]Research Associate, [3]Team Leader, Research and Development [4]Director,
Innodatatics, Hyderabad, India

*Corresponding Author: Bharani Kumar Depuru
ORC ID:0009-0003-4338-8914

Abstract:- Ai-based assessment scrutiny is the most convenient and precise method to eliminate the repetitive task of answer grading; consisting of text extraction methodologies and using Deep Learning Architecture to evaluate with reference to the correct answer and Question provided. In the landscape of educational assessment, the traditional methods of answer evaluation face challenges in adapting to the dynamic and evolving nature of learning. This paper proposes a complete end-to-end answer-grading architecture that can be deployed to provide an interface for a fully automated- Deep-learning answer-grading mechanism.

This research introduces a groundbreaking approach to address these challenges, presenting a solution that seamlessly integrates advanced text extraction and deep learning architectures. Our objective is to achieve unparalleled precision in answer evaluation, setting a new standard in the field. Our method involves the extraction of audio files, precise text extraction from audio, and a Deep Neural Networks DNN-based model for answer evaluation, based on a database that provides the correct answer and relevant data is fetched. Proposing a reliable, accurate, easy-to-deploy best-in-class technology to eradicate manual repetitive tasks.

Providing a very user-friendly interface to the student, and a dynamic backend to monitor results along with the high level of precision. These AI-based evaluation methods can be used in numerous places in the evolving Education industry providing students with a convenient interface and automation. The objective is to elevate the precision and adaptability of answer assessment methodologies in the dynamic landscape of modern education. The educational landscape continues to evolve, our research not only addresses current challenges but also lays the groundwork for future advancements in the field of educational assessment, promising a new era of precision and adaptability.

This paper includes text extraction from architecture-based Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and transformers like an encoder-decoder transformer (whisper).

Keywords:- *Audio Evaluation, Text Extraction, Deep Learning, Grading Answer, Whisper, PALM2, Flask.*

I. INTRODUCTION

In today's scenario, a significant number of competitive exams adopt a multiple-choice format, posing a challenge for students to provide detailed answers. When dealing with a large student population, the manual evaluation of responses becomes practically unfeasible. With the surging demand for AI and software-related jobs, students aspire to excel in these subjects. Considering these factors, we have developed an application that allows students to verbally respond to given questions, with the system providing automated evaluations. This recording process enhances students' confidence in the subject matter and improves soft skills like verbal communication. Students can take immediate action as the score is displayed within no time. It maximizes the automation for the evaluation; this not only reduces costs by minimizing manual correction efforts but also saves time as responses are recorded rather than written.

ISSN No:-2456-2165

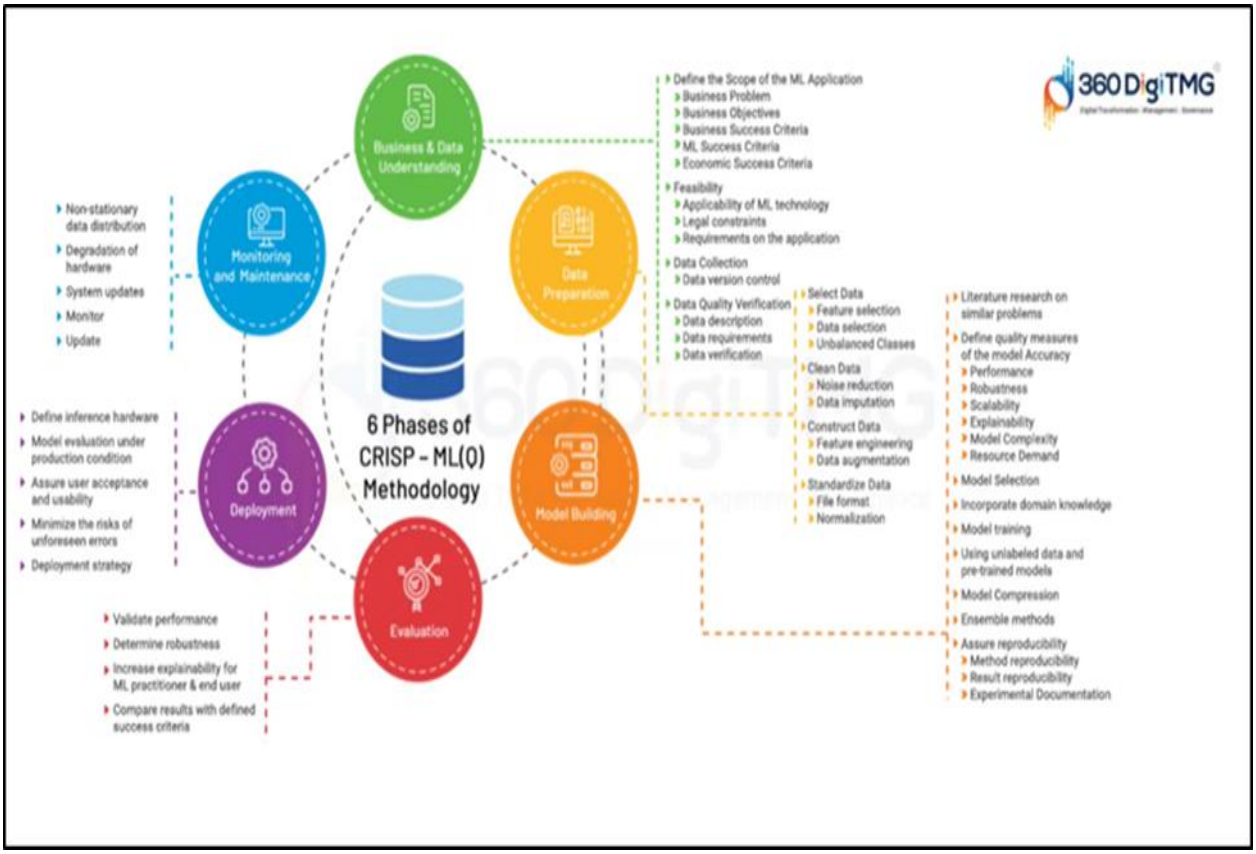

Fig. 1: CRISP-ML (Q) Methodological Framework, Outlining its Key Components and Steps Visually
Source: Mind Map - 360DigiTMG

The application employs the open-source Cross Industry Standard Practice for Machine Learning (CRISP-ML) methodology by 360DigiTMG. CRISP-ML(Q) [Fig.1] [1] is designed to guide the project lifecycle of a machine-learning solution. Deep-learning techniques are extensively utilized for text extraction from audio and subsequent evaluation, incorporating diverse architectures such as Convolutional Neural Networks [15] (CNN), Recurrent Neural Networks (RNN) [14], among others. The project initiation involved thorough research into various techniques. We recorded and gathered diverse audio samples, questions, and answers. Data visualization was performed, and a model was developed, with comparisons made to other models. The process involved the use of a NoSQL database and subsequent deployment. Monitoring confirmed the system's high accuracy.

II. METHODS AND TECHNOLOGY

A. System Requirements (Computer Hardware and Software) used

Table 1: System requirements

Operating System	Ubuntu
RAM	16 GB
Instance Type	g4dn.xlarge
GPU	16 GB

This above table [Table.1] represents the entire system requirement to build and execute this project.

B. Model Architecture

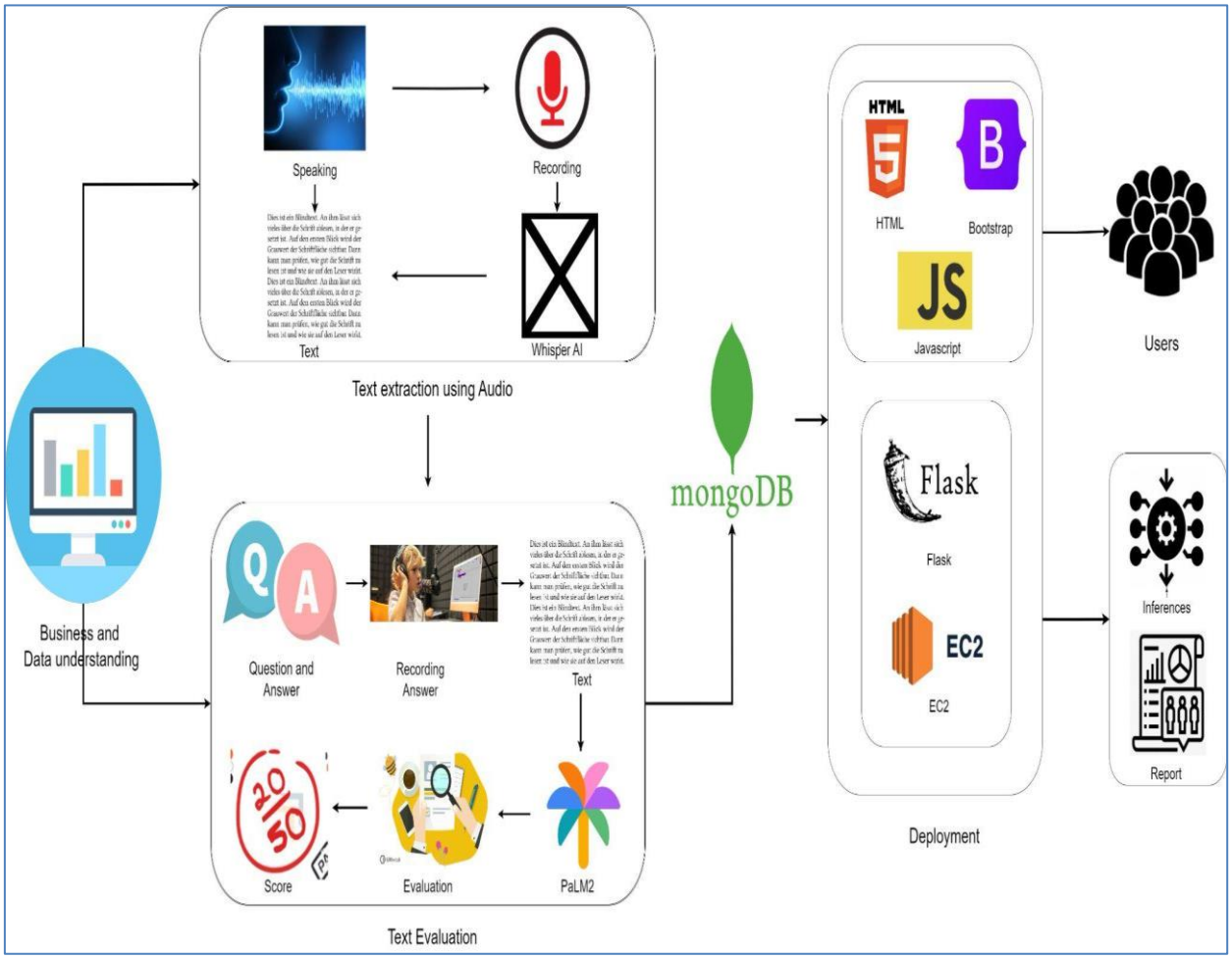

Fig. 2: Architecture Diagram: explanation of the workflow of the [AI evaluation] project
Source: https://360digitmg.com/ml-workflow

The project architecture [Fig.2] explains how the entire project has been conducted in developing the entire model. According to the business problem, relevant data was generated for text extraction via recording samples of different distributions as well as evaluation by forming a dataset for checking the evaluation performance on sample data, after the selection of models, the text extraction was fed with an audio input which generated a text output using a pipeline [2], this output was then sent to evaluation models along with respective correct answers which were extracted from a database that stored all questions and correct answer, after this the model would display the score to the user. Once the model was finalized, with the help of Flask, the application was deployed into the EC2 instance (ec2 is a service in AWS seamless deployment on a big scale for easily scalable for the end users.

For the inference, one UI will open, where the answer will be recorded which will be on the systems and audio chunks that will get converted to proper audio file format. That audio will be passed to the model for text extraction, after which the text will get further sent to the evaluation pipeline to get graded. Once the result is obtained in the backend it will then be rendered to the user's screen in the UI.

C. Data Collection: -

The data that is utilized in this project consists of data acquired through the repetitive recording of diverse audio samples from different individuals, this process was conducted in a very specific manner, ensuring that it covers various accent distributions. A total of 30 audios we have collected to create a diverse dataset. This systematic approach not only facilitates the representation of diverse accents but also enables a robust foundation for the subsequent analyses and evaluations that are within the scope of our research.

A second dataset was produced for evaluation. Here, we have prepared a set of questions covering the three topics (data structures, AI, and Python). It even includes student responses in addition to the right answer. We ensured to address the entire spectrum of student responses, including those that were partially right, wrong, and entirely right. After reviewing this, we concluded that the marks were given correctly.

D. Dataset Dimension

Table 2: Data Dimension

Audio file format	.wav
Text data format	.txt

This above table [Table.2] portrays the data related details which has been used throughout the project.

E. Model Building

The project has two parts where one focuses on text retrieval using audio [3] while the other evaluates the audio and assigns a score [4]. Our initial phase involved functioning on retrieval text from audio during this stage we checked several models that include Speech2text, Deep speech, and whisper to explore their efficacy.

➢ *Speech2text*

Speech-to-Text [6] can handle audio more quickly than real-time, averaging 15 seconds to process a 30-second audio clip. Your recognition request may take much longer if the audio quality is low. It feeds speech inputs into the encoder after reducing their length by 3/4th using a convolutional down sampler. The transcripts/translations are produced autoregressive by the model, which is trained using standard autoregressive cross-entropy loss. LibriSpeech [7], CoVoST 2, MuST-C, and other datasets have been used to refine Speech2Text for ASR and ST. An extracted float tensor of log-mel filter-bank features from the speech signal is accepted by the Speech2Text speech model. Since it's a transformer-based seq2seq model [5], the transcriptions are produced in an autoregressive manner.

➢ *Whisper*

Whisper [Fig.3] [8] exhibits a powerful ability to generalize across various datasets and domains. However, its expected performance can be enhanced for particular languages and assignments by fine-tuning using semi-supervised learning on a large dataset consisting of 680,000 hours. It is a flexible tool designed to handle recordings by dividing them into 30-second segments and processing each sequentially. Achieving an accuracy rate ranging from 95% to 98.5% without manual intervention, the model operates on the transformer architecture, featuring stacked encoder and decoder blocks with an attention mechanism facilitating information exchange between them. Developers have the flexibility to integrate it into their pipelines and customize it to suit their specific use cases, freeing them from dependency on OpenAI. Whisper excels in recognizing various accents, background noises, and technical jargon, supporting over 57 languages, such as Afrikaans, Czech, and Galician, in addition to English. Furthermore, it can translate content from 99 languages to English. Despite its impressive capabilities, Whisper remains cost-effective compared to alternative solutions.

Compare to Deep Speech [Table.3] and Speech2Text [Table.4] it depicts more accurate and error rate is also very less.

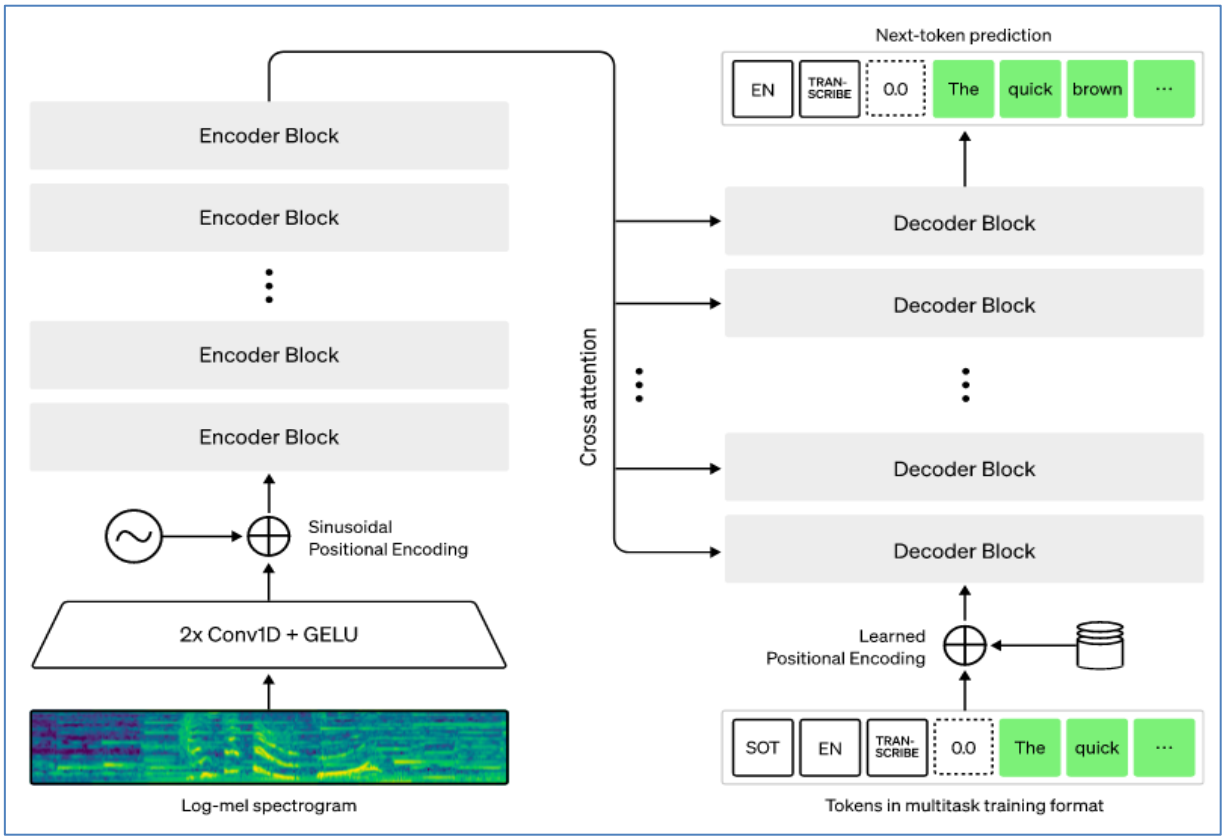

Fig. 3: Whisper Architecture Diagram
Source: Architecture Diagram)

➢ *Deep Speech*

Deep speech [9] employs an end-to-end approach characterized by five-layer activation functions in the first second third and fifth layers involve clipped rectified-linear functions while the fourth layer is recurrent the last layer incorporates a SoftMax function Remarkably deep speech demonstrates robust performance in challenging conditions such as background noise and speaker variations its training system utilizes a recurrent neural network (RNN) with multiple GPUs. Notably, the model exhibits a 160 error rate on the comprehensive test set.

Table 3: Comparison table of Whisper and Deep Speech

Feature	Whisper	Deep Speech
Architecture	Recurrent neural network (RNN) or transformer-based model	Connectionist temporal classification (CTC)
Dataset size	680,000	500,000
WER	4.3	3.55
Organization	Open-AI	Mozilla

➢ *Feature*

Table 4: Comparison table of Whisper and Speech2Text

Feature	Whisper	Speech2Text
Handling Background Noise	Excels in handling background noise, including ambient room noise, outside noise, or music playing	May face challenges with background noise, potentially impacting transcription accuracy
Music Performance	Performs well even when the speaker is performing music (singing, rapping, spoken-word poetry)	May struggle with accurate transcription during musical performances
Error Reduction	Reports 20% fewer additions of missing words, 45% fewer corrections per transcription	May have higher rates of additions and corrections in transcriptions
Accented English and Rapid Speech	Demonstrates high accuracy with English speakers having accents and rapid speech	May experience challenges with accented English and rapid speech, potentially leading to lower accuracy
Auto-Translation	additional feature - auto-translation to English text	Does not have similar unexpected features

Then now comes the second part, evaluating the audio based on the question and assigning a score to it. To do this we employed a range of llm models comprising llama-2 mistral-7b zephyr-7b and palm2.

➢ *Llama-2*

Llama 2 [10] has undergone a fine-tuning process tailored for chat-related applications encompassing training with a substantial dataset comprising over 1 million human annotations. Furthermore, its fine-tuned models undergo training with the aid of over 1 million human annotations enhancing their adaptability to various chat scenarios. Notably, llama 2 exhibits the flexibility to undergo further fine-tuning using newer data inputs. When users input a text prompt or provide text to llama 2 through alternative means the model endeavors to predict the most plausible subsequent text. This predictive capability is achieved through a neural network characterized by a cascading algorithm housing billions of variables commonly referred to as parameters. This intricate neural network architecture is designed to emulate aspects of the human brain enabling Llama 2 to generate contextually relevant and coherent text outputs in response to user inputs.

➢ *Zephyr-7b*

Zephyr-7B [11] is constructed with a combination of transformers, including transformers 4350 dev0 pytorch, 201 ku 118 datasets, and 2120 tokenizers 0140, boasting a vast parameter count of 7 billion. Trained extensively in diverse languages, it excels in tasks such as translation, summarization, analysis, and answering questions. The training process involved a blend of public and synthetic datasets, employing direct preference optimization. Fine-tuning further enhances its capabilities, ensuring accurate information retrieval tailored to specific queries. The model's training corpus encompasses a wide range of sources, including websites, articles, books, and more.

➢ *Mistral 7b*

Mistral 7b distinguishes itself as the earliest large language model [12] by utilizing Sliding window attention to effectively for bigger patterns at a lower price and group-query attention for quick interference. High throughput and low latency are made possible by its distinctive architecture but it becomes difficult to stay accurate when writing a lot of text. In spite of this mistral 7b performs better than Llama 2 in several areas.

➢ *PaLM-2*

PaLM2 has various features like Multilingualism, logic, coding, effectiveness, and economy of cost. PaLM2 [Fig.4] [13] is a language processing model that gathers diverse data, cleans it, and uses the Transformer built for efficient training. It undergoes unsupervised pre-training, fine-tuning, and fine-tuning on smaller datasets for real-world tasks. Its Pathways employs decoupling and adaptive computation for holistic understanding and accurate outputs.

Compared to Llama 2 [Table.5] and other models, Palm2 is very much accurate and also hallucinate very less. Compared to Palm 2 other models hallucinate with complicated prompts.

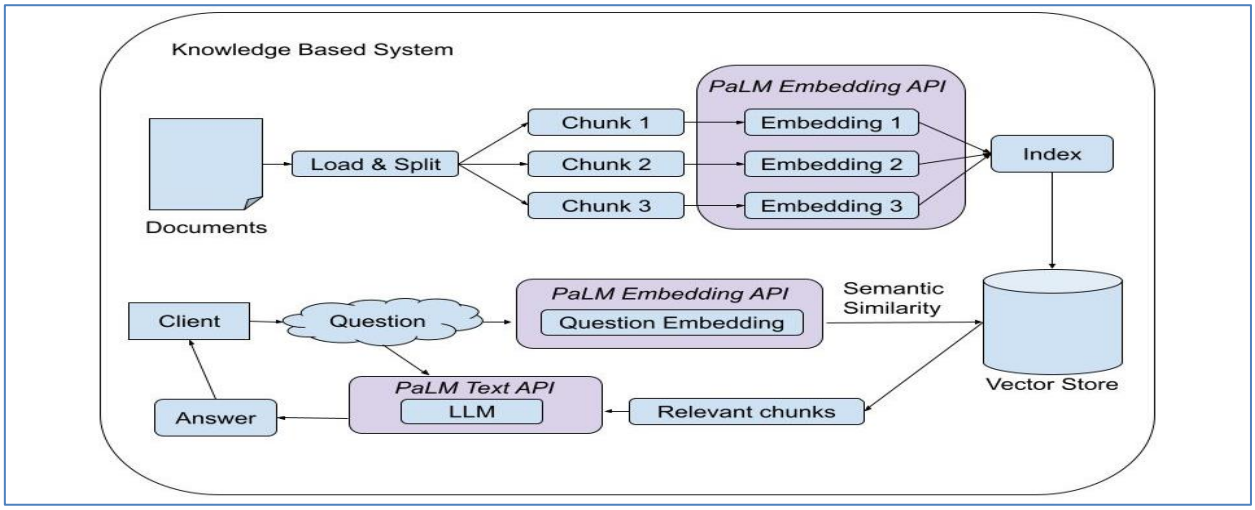

Fig. 4: PaLM-2Architecture Diagram
Source:-Architecture Diagram

Table 5: Comparison between Palm 2 and Llama 2

Feature	**PaLM 2**	**Llama 2**
Model size	540 billion parameters	70 billion parameters
Training data	560 billion words	560 billion words
Architecture	Transformer-based	Transformer-based
Training method	Self-supervised learning	Self-supervised learning

F. Model Evaluation

After recording numerous audio samples and evaluating them, we found that Whisper demonstrates superior accuracy when compared to other models like Speech2Text and Deep Speech. Whisper exhibited precision with a low word error rate, leading us to choose it over other models. Notably, it effectively eliminated background noise and provided accurate transcriptions. For evaluation, we employed several LLM models, including PaLM2, llama-2, mistral-7b, and zephyr-7b. Prompt engineering has been applied to these LLM models in several ways. PaLM2 consistently yielded scores closely resembling human evaluation standards.

III. RESULTS AND DISCUSSION

Text extraction from audio using the whisper model coupled with accurate evaluation by PaLM2 yielded successful results. Integration through the flask was accomplished leading to a successful deployment in Amazon Web Services (AWS) EC2 instance which is scalable and cost-efficient. The user-friendly design of this project has proven beneficial for students fostering confidence and optimizing time usage. This made the evaluation process easily accessible.

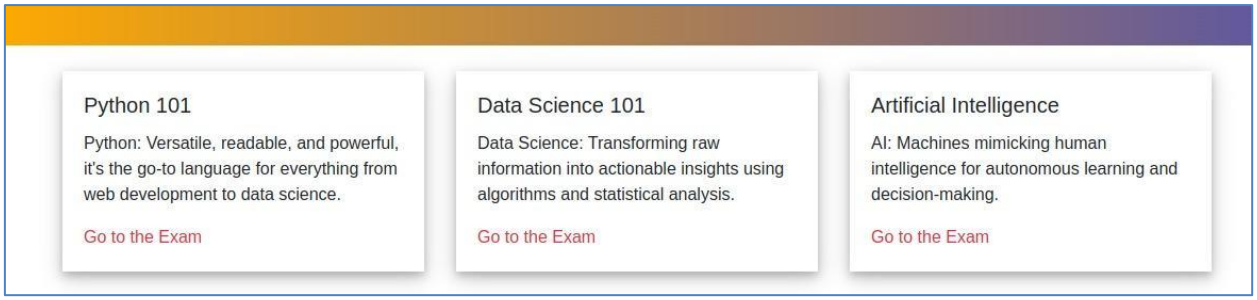

Fig. 5: Welcome page of the application

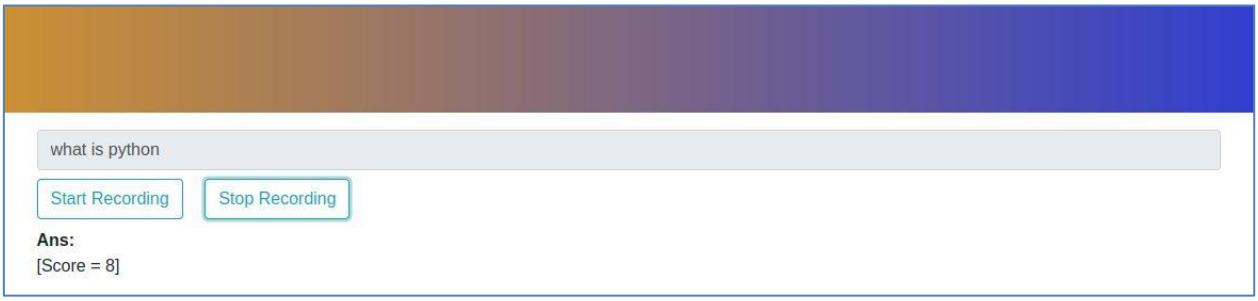

Fig. 6: Exam evaluation page of the application

First, we land on the welcome page [Fig.5] of the application which provides 3 different boxes each containing a description of the subject for the exam, apart from the discussion there is a link that redirects to a separate evaluation page [Fig.6]. On this page, we have a block in which the question is shown, and below that is the "start recording" button which upon pressing starts recording the answer that the student speaks in English. On pressing the stop recording button the audio recording gets completed the audio gets evaluated in the backend and the answer is returned which is the score of the answer is shown out of 10.

IV. CONCLUSION

Our research shows a major step toward revolutionizing answer evaluation methodologies in the quickly changing field of educational technology. Our work tackles the inherent challenges of traditional assessment approaches by integrating sophisticated Deep Learning Architectures with advanced text extraction techniques. Our project aims to improve the accuracy and flexibility of answer evaluation and to make answer grading more convenient and scalable. It is also expected to be a significant development in the field of education. We present a paradigm shift in automating the evaluation process by navigating the complexities of student responses with the seamless integration of AI technologies, particularly deep learning models.

REFERENCES

[1]. Stefan Studer, Thanh Binh Bui, Christian Drescher, Alexander Hanuschkin, Ludwig Winkler, Steven Peters and Klaus-Robert Muller, Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology, 2021, Volume 3, Issue 2.https://doi.org/10.3390/make3020020

[2]. Rafael Dantas Lero, Chris Exton, Andrew Le Gear, Communications using a speech-to-text-to-speech pipeline, Published: International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2019, DOI:https://doi.org/10.1109/WiMOB.2019.8923157

[3]. Pooja Panapana, Eswara Rao Pothala, Sai Sri Lakshman Nagireddy, Hemendra Praneeth Mattaparthi& Niranjani Meesala Towards Automatic Bidirectional Conversion of Audio and Text: A Review from Past Research, 2023, volume 716 DOI:https://link.springer.com/chapter/10.1007/978-3-031-35501-1_30

[4]. Lishan Zhang, Yuwei Huang, Xi Yang, Shengquan Yu &Fuzhen Zhuang Towards An automatic short-answer grading model for semi-open-ended questions, 2019 DOI: https://doi.org/10.1080/10494820.2019.1648300

[5]. Shuyu Li, ,Yunsick Sung towards Transformer-Based Seq2Seq Model for Chord Progression Generation, 2023 DOI: https://doi.org/10.3390/math11051111

[6]. Changhan Wang, Yun Tang, Xutai Ma, Anne Wu, Sravya Popuri, Dmytro Okhonko, Juan Pino | fairseq S2T: Fast Speech-to-Text Modeling with fairseq, 2020, DOI:https://doi.org/10.48550/arXiv.2010.05171

[7]. Vassil Panayotov, Guoguo Chen, Daniel Povey, Sanjeev Khudanpur towards Librispeech: An ASR corpus based on public domain audio books| Published in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015 DOI: https://doi.org/10.1109/ICASSP.2015.7178964

[8]. Xuedong Huang, A. Acero, F. Alleva, Mei-Yuh Hwang, Li Jian g& M. Mahajan towards whisper: Microsoft Windows highly intelligent speech recognizer, 2002, https://doi.org/10.1109/ICASSP.1995.479281

[9]. Awni Hannun, Carl Case, Jared Casper & Bryan Catanzaro Towards Deep Speech: Scaling up end-to-

end speech recognition, 2014 DOI:https://doi.org/10.48550/arXiv.1412.5567

[10]. Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, Thomas Scialom towards Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023 DOI:https://doi.org/10.48550/arXiv.2307.09288

[11]. Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush & Thomas Wolf towards Zephyr: Direct Distillation of LM Alignment,2023 DOI: https://doi.org/10.48550/arXiv.2310.16944

[12]. Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, William El Sayed towards Mistral-7b, 2023 DOI: https://doi.org/10.48550/arXiv.2310.06825

[13]. Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta towards: PaLM 2 Technical Report DOI:https://doi.org/10.48550/arXiv.2305.10403

[14]. Alex Sherstinsky towards Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network DOI:http://dx.doi.org/10.1016/j.physd.2019.132306

[15]. Keiron O'Shea, Ryan NashTowards An Introduction to Convolutional Neural Networks DOI:https://doi.org/10.48550/arXiv.1511.08458