PROJECT REPORT

On

LINGUAKIDS: EMPHASIZES LANGUAGE LEARNING TAILORED SPECIFICALLY FOR CHILDREN

Submitted in partial fulfilment for the award of degree

Of

Bachelor of Technology

In

Computer Science & Engineering

By

ANAGHA ANIL KUMAR (MLM21CS029)

ANN LITTA JOE (MLM21CS034)

JINU TREESA ABRAHAM(MLM21CS058)

JITHU REJI (MLM21CS060)

Under the Guidance of

Ms. SACHUSHA S SHAJI

(Assistant Professor, Dept. of Computer Science and Engineering)

DEPARTMENT OF COMPUTER SCIENCE&ENGINEERING

MANGALAM COLLEGE OF ENGINEERING, ETTUMANOOR

(Affiliated to APJ Abdul Kalam Technological University)

MARCH 2025

MANGALAM COLLEGE OF ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Vision

To become centre of excellence in computing and research where future generations embrace technologies wholeheartedly and use its possibilities to make the world a better place to live.

Mission

Enlight the young talents to achieve academic excellence as well as professional competence by imparting state of the art knowledge in computing and to be admirable individuals with ethical values and appropriate skills.

Program Educational Objectives

PEO 1: Graduate will have strong foundation and profound knowledge in computing and allied engineering and be able to analyze the requirements of real world problems, design and develop innovative engineering solutions and maintain it effectively in the profession.

PEO 2: Graduate will adapt to technological advancements by engaging in higher studies, lifelong learning and research, there by contribute to computing profession.

PEO 3: Graduate will foster team spirit, leadership, communication, ethics and social values, which will lead to apply the knowledge of societal impacts of computing technologies.

Program Specific Outcomes

PSO 1: Apply the Principles of Computing in solving real world problems with sustainability.

PSO 2: Apply Futuristic technology in designing and developing hardware and software solutions.

MANGALAM COLLEGE OF ENGINEERING, ETTUMANOOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MARCH 2025

CERTIFICATE

This is to certify that the Project titled ***"Linguakids: Emphasizes Language Learning Tailored Specifically For Children"*** *is the Bonafide record of the work done by* ***Anagha Anil Kumar (MLM21CS029),Ann Litta Joe (MLM21CS034),Jinu Treesa Abraham (MLM21CS058),Jithu Reji (MLM21CS060)*** *of B.Tech in Computer Science and Engineering towards the partial fulfillment of the requirement for the award of the* ***DEGREE OF BACHELOR OF TECHNOLOGY by APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY****, during the academic year* ***2024-25.***

Internal Examiner **External Examiner**

Project Guide
Ms. Sachusha S Shaji
Assistant Professor
Department of CSE

Head of the Department
Dr. Simy Mary Kurian
Associate Professor
Department of CSE

ACKNOWLEDGEMENT

We are greatly indebted to the authorities of Mangalam College of Engineering for providing us the necessary facilities to successfully complete our Project on the topic "**Linguakids: Emphasizes Language Learning Tailored Specifically for Children**".

We express our sincere thanks to **Dr. Vinodh P Vijayan**, Principal, Mangalam College of Engineering for providing us the facilities to complete this work successfully.

We thank and express our solicit gratitude to **Dr. Simy Mary Kurian**, HOD, Department of Computer Science & Engineering, Mangalam College of Engineering, for her invaluable help and support which helped us a lot in successfully completing this Project work.

We express our gratitude to our Internal Guide, **Ms. Sachusha S Shaji**, Assistant professor, Department of Computer Science & Engineering for the suggestions and encouragement which helped us in the successful completion of our work.

Furthermore, we would like to acknowledge with much appreciation the crucial role of the faculties especially Project Coordinator, **Ms. Devipriya S Kumar**, Department of Computer Science & Engineering, who gave the permission to use all the required equipment and the necessary resources to complete the Presentation and Report.

Finally, we would like to express our heartfelt thanks to our parents who were very supportive both financially and mentally and for their encouragement to achieve our goal.

ANAGHA ANIL KUMAR (MLM21CS029)
ANN LITTA JOE (MLM21CS034)
JINU TREESA ABRAHAM (MLM21CS058)
JITHU REJI (MLM21CS060)

ABSTRACT

In today's world, innovative educational tools are increasingly leveraging AI to enhance learning experiences for children. This AI-powered English language learning system is designed to make language learning enjoyable and effective for children. It catches spoken words, accurately transcribes them, corrects grammatical errors, and develops learning strategies unique to each child based on their development. It uses video avatars for interactive graphics, generative artificial intelligence (AI) for error correction and personalized learning routes, and automatic speech recognition (ASR) for accurate transcription. This method provides a rich and captivating learning environment that develops confidence and fluency in English.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

ABBREVIATION		FULL FORM
DFD	-	Data Flow Diagram
ASR	-	Automatic Speech Recognition
AI	-	Artificial Intelligence
LLM	-	Large Language Model

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

Technology is changing how kids learn in the current digital age, making learning more engaging and dynamic. Artificial intelligence (AI)-powered innovative solutions are complementing traditional educational methods to provide personalized learning experiences. One especially intriguing development in this field is an AI-powered English language learning system designed to make language acquisition fun and efficient for young students. By incorporating state-of-the-art technologies, this approach fosters a lifetime love of learning in addition to helping kids grasp the nuances of English.

The way the system works is by listening to kids talk, accurately translating what they say into text, and identifying any grammatical errors. Children can learn from their mistakes and understand how to improve their language skills thanks to this instant feedback. This system stands out for its capacity to create personalized learning plans for every child according to their unique development and progress. Every child may progress at their own pace with the right amount of support to improve their English thanks to this tailored method. By focusing on individual needs, the system helps youngsters build a strong language foundation, which makes it easier and more comfortable for them to express themselves.

To engage children, the AI system utilizes visually captivating elements such as video avatars and interactive graphics. These components transform the educational experience into one that is both informative and entertaining. Through the implementation of automatic speech recognition technology, the system is capable of accurately interpreting and responding to spoken words, while generative AI offers intelligent error correction and tailored learning paths. This synergy fosters an engaging and immersive learning atmosphere where children feel encouraged and inspired to enhance their English skills. Consequently, they not only gain fluency but also develop the confidence to communicate effectively, making language acquisition a fun and fulfilling journey.

1.2 INTRODUCTION

In the modern interconnected world, proficiency in English has grown immensely essential, particularly for young learners. English acts as a crucial instrument for interaction, learning, and even

prospects for future employment. Nevertheless, numerous children encounter considerable obstacles while trying to master the language. These obstacles may consist of challenges in pronunciation, pauses during conversations, and common errors in grammar and vocabulary. Such difficulties can render the experience of learning English intimidating and overwhelming.

These difficulties can significantly affect a child's self-esteem. When kids struggle to articulate words properly or make repeated mistakes, they may feel disheartened and become reluctant to share their thoughts. This diminished confidence can hinder their capacity to communicate effectively, resulting in frustration and an unwillingness to join discussions or take part in classroom activities. Consequently, their overall involvement with the language may decrease, making it even more challenging for them to progress.

Furthermore, the difficulties of learning English may cause one to lose all interest in the language. Children may become disengaged and look for alternative things that seem more pleasurable or fulfilling if they believe that they are not improving or that the learning process is too challenging. Finding strategies to rekindle their interest and desire is essential since this disengagement can lead to a vicious cycle where their inability to practice further impairs their capacity to learn.

To address these issues, it is essential that English learning experiences be engaging and fun. By incorporating interactive elements, creative teaching methods, and enjoyable activities, educators can create an environment that fosters enthusiasm for the language. When children find joy in learning, they are more likely to overcome their difficulties, build their confidence, and develop a lasting appreciation for English. Making learning enjoyable is key to helping children navigate the challenges they face and succeed in their language journey.

1.3 PROBLEM STATEMENT

Kids often encounter considerable difficulties with clear pronunciation while learning English, and many available tools fail to accurately capture their speech. This inaccuracy can lead to frustration, as young learners miss the precise feedback essential for their improvement. Often, the feedback they receive is too advanced for their comprehension, leaving them uncertain about how to rectify their errors. Moreover, numerous current language learning tools are devoid of engaging and interactive features, which are vital for maintaining children's interest; without these elements, the learning experience can seem monotonous and uninspiring, reducing motivation. To genuinely assist children on their language-learning journey, it is crucial to create resources that not only provide accurate evaluations of their speech but also foster a fun and captivating learning

atmosphere. By incorporating engaging activities and feedback suited for their age, we can cultivate a positive environment that inspires children to explore and enjoy learning English, nurturing a lasting affection for the language.

1.4 MOTIVATION

The inspiration for this initiative stems from the urgent need to assist children as they navigate their English language learning experiences, especially since English proficiency is becoming increasingly important for effective communication and future prospects. Numerous young learners encounter significant barriers, such as grammatical errors and inadequate feedback, which can undermine their self-esteem and enthusiasm for the language. By creating an AI-driven learning tool that effectively captures their speech and delivers engaging, age-appropriate responses, we aim to establish a nurturing environment that encourages growth and joy in the learning process. Existing language learning applications often do not address the specific needs of children, resulting in boredom and frustration. This initiative seeks to fill that void by providing an interactive and lively experience that holds children's attention and makes learning fun. Ultimately, our aim is to empower children to conquer obstacles, enhance their language abilities, and develop a lasting passion for English, thus positively influencing their educational path and equipping them with the confidence necessary to communicate well in a globalized society.

1.5 SCOPE

The project is to create and deploy a cutting-edge AI-powered English language learning application especially for kids, tackling the various obstacles they encounter when learning the language, including problems with vocabulary and grammar. The tool will efficiently record and assess spoken language by leveraging state-of-the-art voice recognition technology, offering immediate, helpful feedback that is crucial for learning success. In order to sustain high levels of engagement, the tool will include interactive features like gamified exercises and animated video avatars, which not only capture children's interest but also make learning fun and interesting. In addition to fostering engagement, the project will emphasize personalized learning pathways, delivering age-appropriate explanations and corrections tailored to each child's individual developmental needs, facilitating a more profound understanding of the language. The project will also investigate various strategies for implementation across different educational contexts including classrooms, tutoring centers, and home learning environments to ensure the tool is accessible.

CHAPTER 2

LITERATURE REVIEW

2.1 Comuniqa: Exploring Large Language Models for Improving English speaking

Key Aspects:

Large language models (LLMs) have the potential to enhance English speaking abilities, especially in nations like India where the language is essential for interpersonal, professional, and academic communication. English continues to be a non-native language for many, which causes problems with confidence and fluency. The use of human specialists in traditional speaking skill-building techniques can have limitations in terms of pricing, accessibility, and scalability. These problems have hopeful answers in the form of more scalable and approachable options for language learners thanks to recent developments in artificial intelligence (AI).

To address these challenges, a novel approach utilizing LLMs was developed specifically to enhance English speaking proficiency. This approach leverages the strengths of LLMs to offer interactive, immediate feedback that supports students in refining their speaking abilities. Our method emphasizes a human-centric assessment to identify the strengths and weaknesses of LLM-based systems for language learning, by comparing Comuniqa's results with feedback from human experts.

Participants in this assessment are split into three groups: those who use the LLM-based system, those who get advice from human specialists, and those who use both approaches. We shed light on the efficacy of these learning modes using surveys, interviews, and study sessions. Initial research indicates that whereas LLM-based systems, such as Comuniqa, provide feedback with a noteworthy level of accuracy, they sometimes fall short of human-level cognitive capacities, especially when it comes to emotional comprehension and empathy.

Despite these limitations, Comuniqa represents a significant step toward achieving Sustainable Development Goal 4: Quality Education. By offering a valuable tool for individuals who may not have access to human experts, Comuniqa democratizes language learning opportunities. This system enhances the accessibility and affordability of language education while empowering learners to practice their speaking skills in a supportive, interactive environment, ultimately fostering greater confidence among non-native speakers.

2.2 AI-based learning content generation and learning pathway augmentation to increase learner engagement

Key Aspects:

It makes use of artificial intelligence (AI) to create dynamic elements like quizzes and overviews to maintain learners' interest in online courses. These components are essential for improving interaction and clarity, which helps students comprehend the course material and evaluates their understanding. We hope to provide an educational experience that is both instructive and captivating by integrating AI-generated content, which is essential for maintaining students' motivation and active involvement in their studies.

It employs GPT-2, an advanced language model, to provide real-time content updates customized to various subjects and learning goals. This adaptability guarantees that the course content remains pertinent and in sync with the latest advancements across different domains. Through the continuous updating of overviews and quizzes, we improve the educational experience, making it more engaging and responsive to learners' needs. This level of adaptability is especially crucial in today's rapidly changing educational landscape, where information is perpetually advancing.

The effectiveness of AI-generated content is validated by positive feedback from both automated systems and human evaluations. Automated assessments enable quick analysis of the content's impact, while human feedback provides crucial insights into the learner's experience. This combination allows to continually improve our strategies, ensuring we meet the diverse needs of students while maintaining high educational standards. By integrating AI into online courses, we not only enhance engagement through interactive features but also create a more enriching and enjoyable learning environment via machine learning. The primary goal is to achieve optimal traffic control, including emergency management capabilities, through state-of-the-art innovations.

2.3Artificial Intelligence education for young children: Why, what, and how in curriculum design and implementation

Key Aspects:

Integrating artificial intelligence (AI) education into early childhood education (ECE) requires careful thought to cultivate AI literacy among young learners. First and foremost, it is essential to determine the importance and appropriateness of AI in early education, as it prepares children for a

world increasingly dominated by technology. Basic AI principles, including data-driven pattern identification and the limitations of AI systems, can be introduced in a fun and accessible way for young kids. A teaching approach centered on experiential learning and culturally relevant methods is recommended, allowing children to explore AI technologies in meaningful settings. The "AI for Kids" program exemplifies this approach, offering educators strategies to create inquiry-based learning opportunities that reflect the diverse backgrounds of children. By addressing the questions of "Why," "What," and "How" in AI education, this method not only enhances interest in STEM fields but also equips children with essential skills to effectively navigate the digital landscape.

2.4 Artificial Intelligence (AI) Literacy in Early Childhood Education: The Challenges and Opportunities

Key Aspects:

Artificial Intelligence (AI) literacy has emerged as a crucial area of study in digital literacy education, although research on the subject in early childhood education (ECE) is still in its infancy. In a recent scoping review, 16 empirical research studies that examined several facets of AI literacy, including curriculum building, AI tools, teaching strategies, and evaluation approaches, were assessed. The investigations were conducted between 2016 and 2022. This review gives educators and researchers interested in including young learners in AI education a comprehensive overview of the state of AI literacy in ECE today. The results of the analysis show a number of noteworthy obstacles to integrating AI literacy in early childhood settings. In addition to inadequate curriculum development and ambiguous teaching guidance, significant challenges include a lack of confidence, expertise, and awareness regarding AI among educators. It is crucial to address these challenges to ensure that young children have meaningful learning experiences, as these obstacles may impede teachers' capacity to effectively integrate AI concepts into their teaching methods. Nonetheless, the analysis also points out several opportunities to enhance young children's AI literacy. By promoting an understanding of basic AI concepts, techniques, and viewpoints, teachers can cultivate stimulating learning environments that encourage inquiry and exploration. As artificial intelligence increasingly shapes the digital landscape, it is essential to equip young children with the skills necessary to comprehend and utilize these tools.

2.5 Technology-Assisted Language Learning Adaptive Systems: A Comprehensive Review

Key Aspects:

Technology-supported learning is an essential component of cognitive computing that enables learning on digital platforms. This study offers a comprehensive examination of the trends and advancements in technology-assisted language learning (TALL) systems, shedding light on how these systems have evolved over time. The authors' goal is to provide insight into the state of adaptive language learning now by reviewing works from 2011 to 2021 that have been published in esteemed journals and other online resources.

To evaluate the results, the authors suggest three primary dimensions: spatial and temporal factors, characteristics of the system and the learner, and the variety of adaptations offered. These dimensions assist in clarifying the functionality and efficacy of existing TALL systems, emphasizing their ability to meet the diverse needs of learners. The review reveals that interest in this area has been increasing since 2015, particularly within the Asian region, indicating a rising focus on the role of technology in language.

English stands out as the most commonly explored language within TALL systems, primarily concerning university students. This group appears to derive the most benefit from adaptive language learning technologies, which can tailor learning experiences to individual needs. The research also examines the various methods of adaptation utilized, which enhance the effectiveness of these systems in achieving learning objectives.

The study discusses a number of obstacles and constraints faced by scholars and instructors in the field of TALL, in addition to the encouraging advancements. These difficulties include problems with accessibility, scalability, and incorporating technology into traditional language learning environments. Recognizing these challenges is essential to developing workable solutions that can increase TALL's impact.

The study concludes by providing innovative ideas and tactics that local governments and language instructors might use to improve the accessibility and commercial sustainability of TALL research. Stakeholders can work together to guarantee that adaptive language learning technologies reach different societal segments and enhance language instruction by addressing the issues that have been highlighted and investigating novel approaches.

Table 2.1 – LITERATURE REVIEW

SL. NO.	TITLE	METHODOLOGIES	MERITS	DEMERIT
1	Comuniqa: Exploring Large Language Models for Improving English Speaking Skills ACM (2023)	This paper examines how Large Language Models (LLMs) can improve English speaking skills in non-native contexts like India.	Comuniqa tailors' language lessons to individual proficiency levels. enabling targeted practice and effective skill development	Limited personalization and adaptability Dependence on Human Comparison
2	AI-based learning content generation and learning pathway augmentation to increase learner engagement Elsevier (2023)	To keep learners engaged in online courses, we are using AI to create elements, such a b s overviews and quizzes. /	Offers narrative generation, automatic definitions, and more, enhancing engagement and comprehension in language learning.	Generated content may lack relevance or accuracy. Effectiveness in classroom settings remains untested. Dataset Limitations
3	Artificial Intelligence education for young children: Why, what, and how in curriculum design and implementation Elsevier (2022)	Emphasizing an embodied perspective tied to their experiences, culturally responsive projects that connect AI to prior knowledge, Experiential learning that builds on familiar concepts	Culturally responsive projects connect AI to experiences, enhancing learning through familiarity and embodied understanding.	maintaining accuracy and relevance could be challenging. Resource Limitations.
4	Artificial Intelligence (AI) Literacy in Early Childhood Education: The Challenges and Opportunities Elsevier (2023)	Activity-based learning: which promotes self-directed activities, Experiential learning: emphasizing reflection on hands-on experiences, and Hands-on experience: encouraging direct interaction with AI tools	Activity-based learning promotes self-direction, while hands-on experiences with AI encourage reflection and interaction.	Impacting effective implementation of the approaches. Challenges in AI literacy curriculum issues.

5	Technology-Assisted Language Learning Adaptive Systems: A Comprehensive Review Elsevier (2023)	Approaches in TALL includes adaptive learning for personalized experiences, intelligent tutoring systems for tailored instruction, computer-assisted language learning (CALL) using multimedia, natural language processing (NLP) for feedback and analysis.	TALL enhances language learning through adaptive, personalized experiences, intelligent tutoring, multimedia tools, and NLP feedback.	Complexity of Implementation Limited Adaptation to Learning Styles Operational Challenges

CHAPTER 3

PROPOSED SYSTEM

The proposed "Linguakids" system is an advanced, AI-powered language learning platform designed specifically for children, addressing the limitations of existing ASR (Automatic Speech Recognition) systems. Unlike conventional systems that struggle with accurately transcribing children's speech due to variations in pitch and pronunciation, "Linguakids" employs a fine-tuned ASR model trained on diverse children's speech datasets. This enables it to recognize and adapt to the unique vocal characteristics of young learners, ensuring greater transcription accuracy.

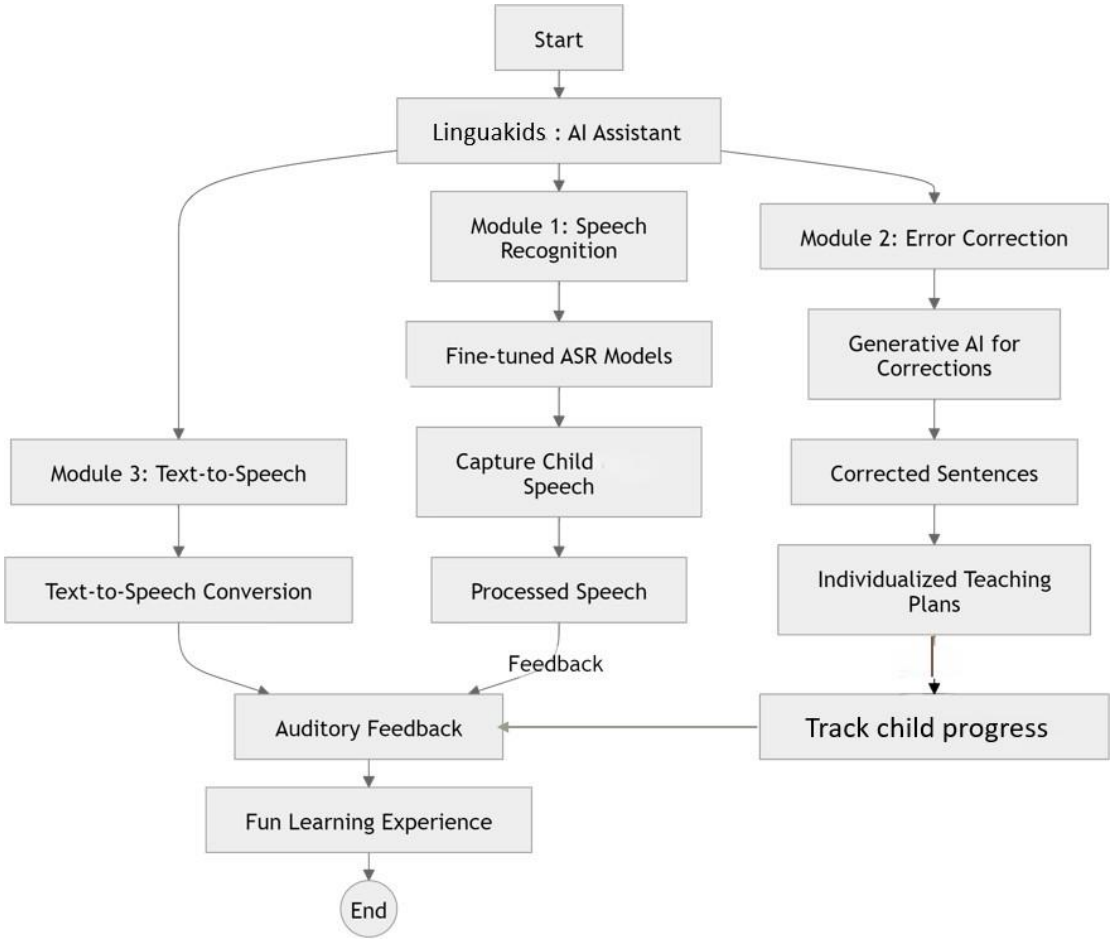

Fig 3.1: Workflow of the Application

A key feature of the system is its interactive video avatar, which enhances the learning experience by providing real-time feedback through dynamic visual and auditory interactions. The avatar not only

corrects speech and grammar mistakes but does so in an engaging manner, turning language learning into a playful and immersive experience. This visual element makes feedback more relatable and fun, maintaining children's interest while providing a natural way to learn.

3.1 Development

The implementation of the "Linguakids" platform begins with comprehensive research and development. Initially, a thorough market analysis is conducted to identify existing language learning platforms aimed at children, highlighting their strengths and weaknesses. This analysis is complemented by user research, including focus groups with parents, educators, and children to gather insights on their needs and preferences. Through these interactions, the team can develop user personas representing diverse demographics, ensuring the platform is tailored to its audience. Defining clear learning objectives is also essential, involving collaboration with educators to establish age-appropriate benchmarks for language skills such as vocabulary acquisition, pronunciation, and grammar proficiency. These foundational steps set the stage for a user-centered design approach that prioritizes the unique challenges faced by young learners.

3.2 Data Collection and ASR Model Training

The next stage of the research process involves gathering data and building a sophisticated Automatic Speech Recognition (ASR) model tailored to the speech of young children. This entails locating a variety of datasets that document a broad spectrum of accents, ages, and speech patterns, such as in-person conversations, reading assignments, and conversational speech. Preprocessing, such as cleaning and normalization, is applied to the gathered data to guarantee consistency and quality. Advanced neural network topologies, such Transformers or Long Short-Term Memory (LSTM) networks, which are especially good at identifying different pronunciations and speech subtleties, are then used to construct the ASR model.Rigorous validation with separate test datasets is essential to ensure the model's accuracy and adaptability. This phase includes establishing a continuous improvement mechanism, where user interactions contribute to ongoing model training, allowing the ASR system to evolve with its users.

3.3 Adaptive Learning Pathways

The platform uses adaptive learning algorithms that dynamically modify content based on individual progress to guarantee that every child's educational experience is customized to meet their specific needs. Real-time performance analysis by these algorithms enables the system to adjust lesson difficulty and, when necessary, introduce additional subjects. A user-friendly dashboard with comprehensive reports on accomplishments and areas for development is used to track progress and is available to parents and instructors. Children are motivated by gamification aspects like badges and awards, which

make learning more interesting. Additionally, a wide range of interactive lessons, exercises, and activities are created, enabling personalization according to each child's interests and skill level and promoting a customized learning experience.

3.4 Avatar Design and Interaction

Central to the "Linguakids" experience is the design of engaging avatars that serve as interactive learning companions for children. The design process begins with creating visually appealing characters that children can relate to, ensuring a range of options that reflect diverse backgrounds and personalities. Once the characters are established, animations are developed to allow the avatars to respond dynamically to user inputs, creating a rich, interactive experience. This involves incorporating gesture recognition technology, enabling the avatars to respond to children's speech and actions in a natural, engaging manner. A critical feature of this module is the feedback mechanism, where avatars provide real-time, constructive feedback on speech and grammar in a supportive tone. This approach not only corrects mistakes but also reinforces learning in an enjoyable way, turning potential frustration into a playful and immersive experience.

CHAPTER 4
METHODOLOGY

The first two key phases of the project approach are data collection and the development of an advanced Automatic Speech Recognition (ASR) model designed to understand children's speech patterns. This initial phase is necessary to ensure that the system accurately captures the unique characteristics of children's language, which may differ significantly from adult speech. By focusing on this population, the study aims to create a model that effectively addresses the particular needs and challenges associated with recognizing and processing children's speech.

An essential part of the research is the human-centric evaluation, which contrasts the ASR system's performance with suggestions from language experts. This comparison is essential for identifying the benefits and drawbacks of language learning models (LLMs) in the context of language acquisition.

4.1 Automatic Speech Recognition (ASR):

The methodology employs Automatic Speech Recognition (ASR) models to transcribe spoken words into text, incorporating preprocessing of the audio signal to enhance its quality. This preprocessing step ensures higher accuracy in transcription, allowing the system to better recognize and convert speech. By focusing on clear audio input, the approach lays a solid foundation for effective language learning. Ultimately, this accuracy in transcription supports the overall goal of improving children's language skills.

4.2 Generative AI Module:

It integrates a generative AI module that focuses on individualized learning paths and mistake correction. In order to provide focused feedback and enhancement, this module analyzes the transcribed text to find problems. It also creates special learning methods based on the developmental requirements of every child. The module improves the efficiency of the language learning process by catering to the unique difficulties and abilities of every student, which promotes greater engagement and advancement. Additionally, the learning tactics are kept current and efficient by the generative AI module's constant evolution based on the child's interactions. A deeper comprehension of language concepts is encouraged by this dynamic method, which also increases the child's confidence and makes learning more effective and pleasurable.

4.3 Text to Speech with Avatar:

The animated avatars are used to visually convey the updated content in an interesting way. As interactive companions, these avatars draw in kids and enhance the appeal of learning materials. The avatars assist children in comprehending the adjustments and enhancements to their speech by providing visual representations of language concepts. This method not only promotes active engagement in the learning process but also improves comprehension. Motivation can be greatly increased by the lively animations' creation of a fun and dynamic environment. Additionally, each encounter stays relevant since the avatars can modify their responses according to the child's development. Overall, this creative utilization of animated avatars makes children's language learning experiences more pleasurable and successful.

4.4 Feedback Generation:

It involves generating instantaneous feedback based on the child's utterances, which is crucial for effective language learning. This feedback is tailored to be relevant, accurate, and helpful, ensuring that children receive timely insights into their speaking performance. By promptly addressing mistakes or areas for improvement, the feedback encourages immediate correction and reinforces language skills. Additionally, it supports a positive learning environment by focusing on constructive suggestions rather than criticism. This approach helps children understand their progress and fosters a sense of accomplishment. The relevance of the feedback is enhanced by considering each child's unique learning path and challenges. Overall, this real-time support significantly boosts the child's confidence and motivation in mastering the language.

4.5 Adaptive Learning Path:

The adaptive learning path is tailored to support each child by incorporating personalized learning experiences. This approach ensures that the learning process is aligned with each child’s unique learning style, promoting a more effective and engaging experience. By recognizing individual strengths and areas for improvement, the methodology fosters a customized journey that caters to diverse needs. Importantly, it emphasizes a non-judgmental environment, allowing children to explore and practice their language skills freely. This supportive atmosphere significantly boosts their confidence in speaking English. As children progress through their personalized learning paths, they are encouraged to take risks and express themselves without fear.Ultimately, this adaptive strategy enhances their overall language acquisition and enjoyment of the learning process.

4.6 Activity-Based Learning:

This approach emphasizes self-directed activities as a core component of effective language learning for children. By encouraging initiative, it fosters a sense of ownership and independence in their studies. Experiential learning allows children to engage with language in real-world contexts, enhancing practical understanding. Hands-on experiences, such as interactive games and projects, make learning enjoyable and memorable. This active involvement promotes exploration and creativity, enabling children to learn through discovery. Ultimately, these strategies create a dynamic environment that significantly enriches the language learning experience.

CHAPTER 5
SYSTEM ARCHITECTURE

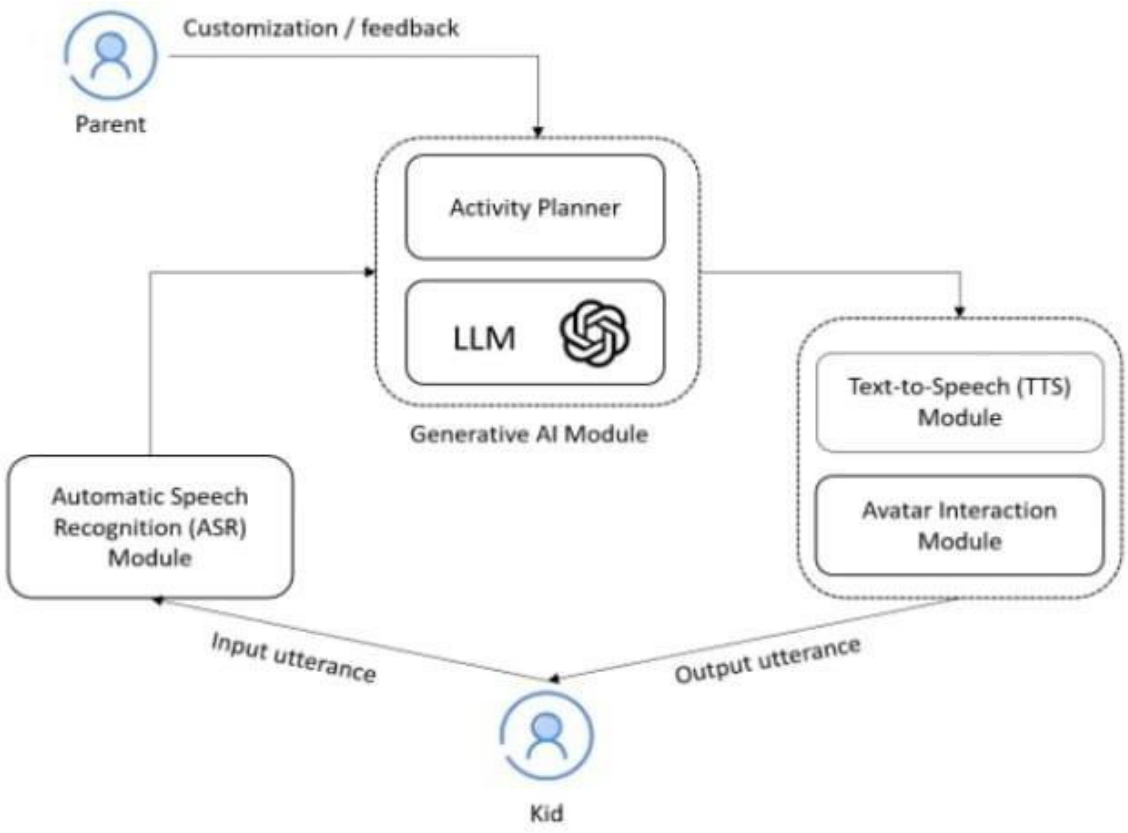

Fig 5.1: System Architecture

5.1 Pretrained ASR Model:

Pre-trained ASR models has been trained on a large dataset of audio recordings and their corresponding transcriptions. The model has learned to recognize patterns in speech and can be fine-tuned for specific use cases.

5.2 Finetuning on Custom Data:

The pre-trained ASR model is then fine-tuned on custom data for better performance. The custom data used for fine-tuning can include labeled audio-text pairs, which are pairs of audio recordings and their corresponding transcriptions. Fine-tuned ASR models are responsible for converting input speech into text.

5.3 Preprocessing Module:

Preprocessing techniques like noise removal and enhance the quality of the audio signal, were done to ensure the audio input is clean and at an optimal level for accurate transcription.

5.4 ASR Model processing:

The preprocessed audio signal is processed by the ASR model, which converts the speech into text. The preprocessed audio signal is fed into the finetuned ASR model, which processes the audio and outputs a raw text transcription of the speech.

5.5 Post processing:

The output of the ASR model is then passed through a post-processing module. This module applies various techniques to improve the accuracy and readability of the output text.

5.6 Output text:

The final output of the ASR system is the text transcription of the input speech. It can be used for making input for the next module.

5.7 Child Utterance:

This utterance is likely in the form of audio, which is then processed by an Automatic Speech Recognition (ASR) system to convert it into text. The ASR system transcribes the audio into a text format that the Generative AI Module can understand and process.

5.8 Utterance processing:

The process involves error detection to check the transcribed text for grammatical and spelling mistakes, context understanding to analyze the situation in which the child is speaking, and intent recognition to determine the underlying meaning behind the child's words.

5.9 Generative AI - Sentence Correction:

This focuses on correcting any errors in the child's sentence. This step uses the output from the Error Detection sub-process to identify the errors and correct them. The goal is to produce a corrected sentence that is grammatically correct and easy to understand.

5.10 Adaptive Learning Path:

Provides an adaptive learning path to support the child by incorporating personalized learning experience.

5.11 Feedback Generation:

It will generate feedback on the child's utterance. This feedback is likely based on the corrected sentence produced in Sentence correction part and takes into account the context and intent of the child's utterance. The goal is to provide feedback that is relevant, accurate and helpful to the child.

5.12 Conversation History Manager:

In this step it manages and tracks the ongoing conversation. This step likely stores the conversation history, including the child's utterances, the system's responses, and any relevant context or metadata. The goal is to maintain a record of the conversation that can be used to inform future responses and improve the overall interaction.

5.13 Multi-path Feedback Adaptation:

It combines inputs from both the Feedback Generation and Conversation History Manager to adapt the feedback. This step likely uses the conversation history to adjust the feedback to better match the child's needs and context. The goal is to provide feedback that is tailored to the child's individual needs and learning style.

5.14 Output - Corrected Sentence & Feedback:

This produces an output, including the corrected sentence and appropriate feedback. This output is likely in the form of text, which can be used as input for avatar interaction.

5.15 Text to Speech with Avatar:

The Text to Speech with Avatar Module displays the corrected text using an interactive avatar, making learning enjoyable for children. By combining visuals with spoken language, it helps keep young learners engaged. The avatar articulates the text and adds gestures, creating a fun environment that encourages participation and boosts confidence in language skills**.**

CHAPTER 6

MODULES

The proposed method uses following modules:

1. Automatic Speech Recognition
2 Generative AI Module
3 Text to Speech conversion

6.1 Automatic Speech Recognition

Pretrained Automatic Speech Recognition (ASR) models are developed by training on extensive datasets of audio recordings paired with their corresponding transcriptions. Through this training, the models learn to recognize patterns in speech, enabling them to effectively understand and process spoken language. To enhance performance for specific applications, these models can be fine-tuned using custom data consisting of labeled audio-text pairs. This fine-tuning process allows the ASR models to adapt to particular use cases, making them more accurate in converting input speech into text.

The ASR process begins with pre-processing techniques that improve the quality of the audio signal, such as noise removal, ensuring that the input is clean and suitable for transcription. Once the audio has been pre-processed, it is fed into the fine-tuned ASR model, which processes the signal and generates a raw text transcription. This output is then refined through a post-processing module that applies various techniques to enhance accuracy and readability. Ultimately, the final output of the ASR system is a coherent text transcription of the input speech, which can serve as input for subsequent processing modules or applications

6.2 Generative AI Module

Child utterances, typically captured in audio format, are processed by an Automatic Speech Recognition (ASR) system that converts them into text. This transcription enables further analysis by various modules, including generative AI. The ASR system not only transcribes the audio but also facilitates subsequent steps such as error detection, context understanding, and intent recognition. These processes work together to ensure that the transcribed text accurately reflects the child's spoken words, enabling the system to comprehend the underlying meaning and intent behind the utterance.Following transcription, the generative AI module focuses on sentence correction, using insights from the error

detection phase to refine any mistakes in the child's speech. This results in a grammatically correct and easily understandable output. Additionally, the system provides personalized feedback tailored to the child's learning path, leveraging conversation history to adapt responses according to individual needs. Ultimately, the output consists of a corrected sentence and relevant feedback, formatted as text for further interactions, ensuring a supportive and engaging learning experience for the child.

6.3 Text to Speech conversion

After being processed by a "Text to Speech Module," the text produced by a generative AI module is converted into spoken English. An "Avatar Generation Model," which is in charge of coordinating the avatar's facial expressions and lip movements with the spoken words, then uses the audio output. The end product is a lively exchange in which the avatar converses with a youngster both orally and visually, improving the user experience overall with realistic reactions.

CHAPTER 7
DIAGRAMS

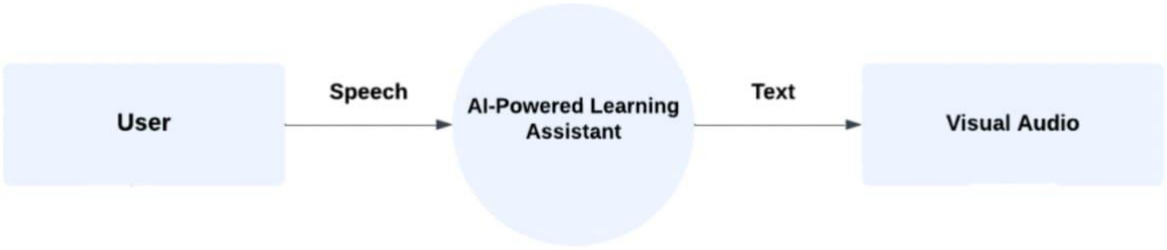

Fig 7.1 level 0 dataflow diagram

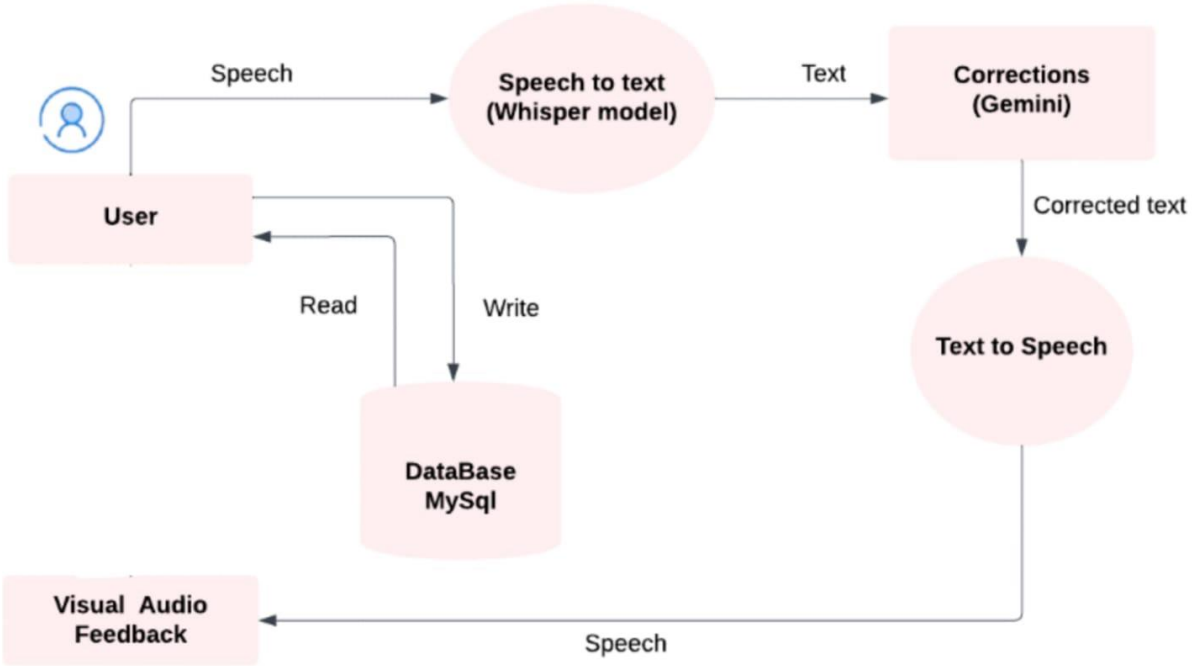

Fig 7.2 level 1 dataflow diagram

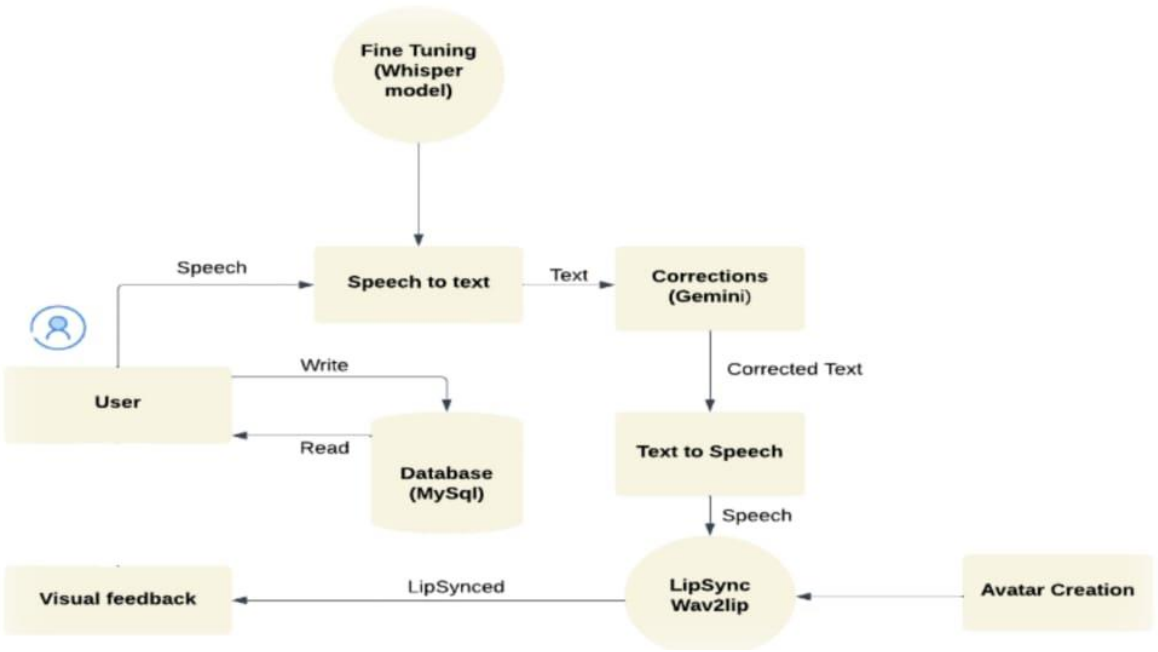

Fig 7.3 level 2 Data flow diagram

CHAPTER 8
TESTING

8.1 UNIT TEST

Unit testing verifies the functionality of key components in the AI-powered English language learning system. The speech recognition accuracy is tested by feeding samples with varying accents, pronunciations, and background noise to ensure accurate transcription of spoken words. The grammar correction function is tested by inputting sentences with errors like tense mistakes or incorrect prepositions, ensuring the system provides meaningful corrections. Transcription functionality is tested by checking how well the system transcribes complex spoken sentences, ensuring minimal errors even with challenging speech patterns. Avatar synchronization is tested to ensure the avatars' lip movements align with the spoken words, providing a smooth, interactive experience. Finally, progress tracking is tested by inputting different learning activities and ensuring the system accurately records and displays the child's progress, such as quiz results and milestones, in a clear and motivating way. These tests ensure each component works correctly in isolation, delivering a seamless and engaging learning experience.

8.2 INTEGRATION TEST

An integration test for this AI-powered English language learning system would focus on verifying the interaction between the various components that work together to deliver a seamless learning experience. This includes testing the integration between automatic speech recognition (ASR), the error correction system, and the personalized learning route generator. The test would ensure that the ASR accurately transcribes spoken words, feeds them correctly into the error correction system, and that any grammatical errors are promptly identified and corrected. Additionally, it would confirm that the personalized learning strategies adapt to each child's progress and developmental needs. The test would also verify the smooth functionality of the video avatars and their responsiveness to student inputs, ensuring that the AI-driven content and feedback are consistently aligned with the child's learning stage. Successful integration testing ensures that all components work cohesively to provide a dynamic, personalized, and effective learning experience.

8.3 FUNCUTIONAL TEST

Functional Testing ensures that the AI-powered learning system operates as expected across key functionalities. The system must accurately recognize and transcribe spoken words, even with varying accents, pronunciations, or background noise, ensuring minimal errors in capturing spoken input. It should also detect and correct grammatical errors such as tense mistakes, subject-verb agreement, and incorrect prepositions, providing clear and helpful feedback for the child. Additionally, the system must convert spoken language into text with minimal errors, especially in complex sentences, to ensure correct transcription. Testing also ensures that the video avatars used in the system respond appropriately to spoken language, with proper synchronization between the spoken words and avatar actions, creating an engaging and interactive learning experience. Finally, the AI should be able to adapt learning content to the child's individual needs by adjusting difficulty levels, selecting appropriate topics, and personalizing exercises based on the child's progress. This ensures that the system can support the child's development and provide an effective, tailored learning path.

8.4 PERFORMANCE TEST

Performance Testing ensures the system functions efficiently under different conditions. System response time is tested by measuring how quickly the system reacts to user inputs, ensuring there's no noticeable lag between speaking and transcription/response. This helps maintain a smooth, engaging learning experience. Load testing checks the system's performance under heavy usage, such as when multiple children are using it simultaneously. It ensures the system remains stable and can handle increased user demand without crashes, slowdowns, or degraded performance. Both tests help verify that the system can deliver a seamless experience for all users, even under varying usage conditions**.**

8.5 USABILITY TEST

Usability Testing focuses on ensuring the system is user-friendly and engaging for children. Ease of use is tested by evaluating how easily children of different age groups can navigate the system independently, without needing adult assistance. Engagement testing assesses the appeal of avatars, videos, and interactive features, determining if they hold the child's attention and make learning enjoyable. Motivation and progress tracking are also evaluated to ensure the system effectively monitors the child's progress and provides feedback that motivates them to keep learning. This testing ensures the system is both intuitive and captivating for young users.

CHAPTER 9

ADVANTAGES & DISADVANTAGES

9.1 ADVANTAGES

Our system offers a comprehensive and innovative approach to language learning, specifically designed for children. One of its key features is personalized learning, which adapts the content to each child's individual learning style and progress. This ensures that every child receives a tailored experience, making language learning more engaging and effective. Interactive feedback plays a crucial role by utilizing animated avatars that provide real-time, constructive feedback on pronunciation and grammar. This method not only corrects errors but also does so in a fun, non-intimidating way, helping children feel more confident and motivated.

The system stands out for its enhanced accuracy due to its fine-tuned Automatic Speech Recognition (ASR) model. Unlike traditional systems, Linguakids' ASR is specifically designed to understand the unique vocal characteristics of children, leading to higher transcription accuracy and better speech recognition. To further enhance the learning experience, gamification elements like rewards, badges, and interactive lessons keep children motivated and interested, turning learning into an enjoyable activity.

Additionally, the generative AI module helps identify and correct speech errors, allowing children to grasp language concepts more deeply and improve their overall language skills.The system increases children's confidence by creating a safe environment where they may practice without worrying about being judged. Active engagement is also encouraged via dynamic interaction with the avatars, which enhances the immersion and enjoyment of the experience.

The goal of Linguakids is constant progress. It changes over time in response to user interactions, honing its tactics to suit the demands of every youngster. The system covers every facet of language learning, from vocabulary to grammar, by fusing ASR, generative AI, and adaptive learning. In the end, the platform's emphasis on enjoyment guarantees that language acquisition is not only successful but also cultivates in kids a lifelong love of learning.

9.2 DISADVANTAGES

Dependence on technology is one concern, as the system requires access to devices and reliable internet, which may be challenging in areas with limited resources. The system's personalization may also have limitations, particularly for children with unique learning needs, making it difficult to fully adapt to all learners.

The complexity of implementation is another issue, as integrating advanced AI technologies requires significant expertise, resources, and continuous maintenance, which could be burdensome for some institutions. There is also a risk of over-reliance, where children may depend too heavily on the system for feedback and corrections, potentially hindering their ability to self-correct and develop independent language skills.

Engagement variability is another challenge, as not all children may respond well to the interactive avatars or gamification, which could impact the system's effectiveness. Data privacy concerns also arise, given the collection of children's data for personalized learning, necessitating strict adherence to privacy regulations. Additionally, the cost of development and maintenance can make the system inaccessible for some families or educational institutions.

Lastly, the system’s unproven effectiveness in real-world settings, potential technical issues, and cultural limitations may require further research and refinement to ensure it is truly effective across diverse contexts.

CHAPTER 10

RESULTS

Epoch	Training Loss	Validation Loss
1	1.628300	1.081734
2	0.557300	0.776524
3	0.142400	0.826893
4	0.046100	0.828026
5	0.027800	0.839953
6	0.012300	0.855187
7	0.013700	0.870543
8	0.001100	0.853390
9	0.000900	0.852212
10	0.000300	0.853356

Fig 10.1 Fine tuning results

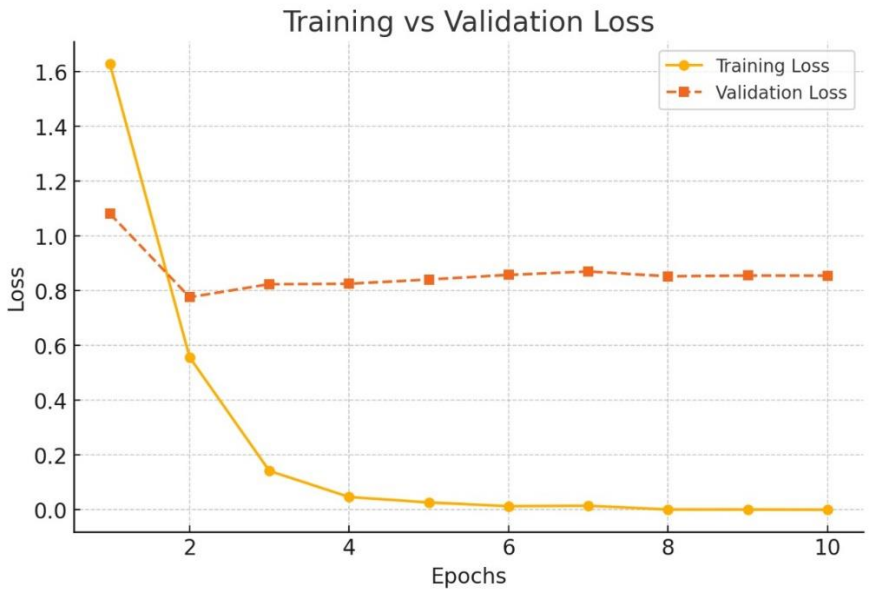

Fig 10.2 Fine tuning Whisper ASR

Fig 10.3Avatar video generation

CHAPTER 11

CONCLUSION & FUTURE SCOPE

9.1 CONCLUSION

The way kids learn English is being revolutionized by LINGUAKIDS, an AI-powered English learning platform. Through the integration of cutting-edge AI technologies such as generative AI modules, adaptive learning routes, and automatic speech recognition, the system offers a highly effective, interactive, and engaging experience that is customized for every child. With the use of captivating avatars, instant feedback, and tailored training, it emphasizes correctness, clarity, and individualized learning. In addition to encouraging a love of studying, this gives kids more self-assurance and the ability to communicate effectively in English.

LINGUAKIDS addresses common challenges in language learning, such as pronunciation difficulties, hesitation, and grammar mistakes, by creating a supportive environment that promotes precision and fluency. The system's interactive elements, including avatars, capture children's attention and offer instant corrections, ensuring an effective learning experience. The personalized learning paths adapt to each child's progress, ensuring relevant, accurate, and helpful feedback.

By leveraging generative AI for error correction and offering learning strategies that evolve with each child's development, LINGUAKIDS sets a new standard in educational technology. The system encourages self-directed learning and direct interaction with AI tools, making language acquisition more immersive.

Overall, LINGUAKIDS is a significant advancement in language learning, setting a new benchmark for personalized and engaging experiences. With its focus on adaptability, relevance, and precision, it is poised to inspire a generation of confident, fluent English speakers and instill a lifelong passion for language learning.

9.2 FUTURE SCOPE

The future scope of LINGUAKIDS is both promising and expansive. With the potential for expanding language offerings, the system could support multiple languages, enabling children from different linguistic backgrounds to benefit from its personalized learning pathways. Incorporating Augmented Reality (AR) could create immersive and interactive learning experiences, offering a more hands-on approach to language acquisition. Advancements in AI capabilities could further

enhance speech recognition and error correction, ensuring even more precise feedback and adaptability. The platform could also foster greater collaboration with educators**,** allowing seamless integration into traditional classrooms and serving as a supplementary tool. Gamification elements could be taken further, making learning even more engaging and enjoyable. Data analytics would provide deeper insights into individual progress, allowing for tailored interventions that address specific needs. As the system evolves, accessibility features for children with special needs could be enhanced, ensuring it remains inclusive. Additionally, the platform could expand its global reach, bringing high-quality language learning resources to underserved regions. By focusing on continuous research and development, LINGUAKIDS could continue to stay at the forefront of educational technology, addressing new challenges and further revolutionizing language learning for children worldwide.

APPENDICES

```
 import pandas as pd
from datasets import Dataset

# Load the CSV files
train_df = pd.read_csv("/content/drive/MyDrive/Training_dataset.csv")
test_df = pd.read_csv("/content/drive/MyDrive/test_dataset.csv")

# Rename columns for consistency
train_df.columns = ["audio_filepath", "transcription"]
test_df.columns = ["audio_filepath", "transcription"]

# Convert to Hugging Face Dataset
train_dataset = Dataset.from_pandas(train_df)
test_dataset = Dataset.from_pandas(test_df)
from datasets import Audio

train_dataset = train_dataset.cast_column("audio",
Audio(sampling_rate=16000))
test_dataset = test_dataset.cast_column("audio", Audio(sampling_rate=16000))
```

```
from transformers import WhisperFeatureExtractor, WhisperTokenizer,
WhisperProcessor

feature_extractor = WhisperFeatureExtractor.from_pretrained("openai/whisper-
small")
tokenizer = WhisperTokenizer.from_pretrained("openai/whisper-small",
language="English", task="transcribe")
processor = WhisperProcessor.from_pretrained("openai/whisper-small",
language="English", task="transcribe")
import librosa

def prepare_dataset(batch):
    # Load audio from file path
    audio_data, _ = librosa.load(batch["audio_filepath"], sr=16000)

    # Process the audio data
    batch["input_features"] = feature_extractor(audio_data,
sampling_rate=16000).input_features[0]
    batch["labels"] = tokenizer(batch["transcription"]).input_ids
    return batch

train_dataset = train_dataset.map(prepare_dataset, num_proc=4)
```

```
test_dataset = test_dataset.map(prepare_dataset, num_proc=4)
from dataclasses import dataclass
from typing import Any, Dict, List, Union
import torch

@dataclass
class DataCollatorSpeechSeq2SeqWithPadding:
    processor: Any

    def __call__(self, features: List[Dict[str, Union[List[int],
torch.Tensor]]]) -> Dict[str, torch.Tensor]:
        input_features = [{"input_features": feature["input_features"]} for
feature in features]
        batch = self.processor.feature_extractor.pad(input_features,
return_tensors="pt")

        label_features = [{"input_ids": feature["labels"]} for feature in
features]
        labels_batch = self.processor.tokenizer.pad(label_features,
return_tensors="pt")
        labels =
labels_batch["input_ids"].masked_fill(labels_batch.attention_mask.ne(1), -
100)

        if (labels[:, 0] ==
self.processor.tokenizer.bos_token_id).all().cpu().item():
            labels = labels[:, 1:]

        batch["labels"] = labels
        return batch

data_collator = DataCollatorSpeechSeq2SeqWithPadding(processor=processor)
from transformers import WhisperForConditionalGeneration

model = WhisperForConditionalGeneration.from_pretrained("openai/whisper-
small")
import evaluate
import numpy as np # Import numpy

metric = evaluate.load("wer")

def compute_metrics(pred):
    pred_ids = pred.predictions
    label_ids = pred.label_ids

    # replace -100 with the pad_token_id
    label_ids[label_ids == -100] = tokenizer.pad_token_id
```

```
    # Check if pred_ids is a tuple and extract the logits if necessary
    if isinstance(pred_ids, tuple):
        pred_ids = pred_ids[0]  # Assuming logits are the first element of
the tuple

    # Flatten predictions before decoding
    # Use numpy's argmax since pred_ids is likely a numpy array
    pred_ids = np.argmax(pred_ids, axis=-1)

    # we do not want to group tokens when computing the metrics
    pred_str = tokenizer.batch_decode(pred_ids, skip_special_tokens=True)
    label_str = tokenizer.batch_decode(label_ids, skip_special_tokens=True)

    wer = 100 * metric.compute(predictions=pred_str, references=label_str)

    return {"wer": wer}
# included training loss

training_args = Seq2SeqTrainingArguments(
    output_dir="./results",
    evaluation_strategy="epoch",  # Evaluate at the end of each epoch
    save_strategy="epoch",  # Save checkpoints at the end of each epoch
    logging_dir="./logs",  # Directory for logs
    logging_strategy="steps",  # Log at specific steps
    logging_steps=10,  # Log training loss every 10 steps
    per_device_train_batch_size=8,
    per_device_eval_batch_size=8,
    save_total_limit=2,
    num_train_epochs=10,
    report_to="none"  # Avoids sending logs to WandB or TensorBoard
)
from transformers import Seq2SeqTrainer
trainer = Seq2SeqTrainer(
    args=training_args,
    model=model,
    train_dataset=train_dataset,
    eval_dataset=test_dataset,
    data_collator=data_collator,
    compute_metrics=compute_metrics,
    tokenizer=processor.feature_extractor,

)
import os
import streamlit as st
import mysql.connector
from mysql.connector import Error
from tras_noisy import capture_audio,start_capture,stop_capture
import google.generativeai as genai
```

```
 from dotenv import load_dotenv
 import base64
 import threading
 import tempfile
 from txtspgtts import tts
#from text_to_speechnew import tts  # Import the tts function
from apitrial import generate_video
import requests
# Load environment variables
load_dotenv()
OPENAI_API_KEY = os.getenv("API_KEY")
genai.configure(api_key=OPENAI_API_KEY)
#elevenlabs api connection
ELEVENLABS_API_KEY = "sk_27e296e3930259ddd8213bbe69da1aea3d03c43830e0374b"

# Use a valid voice ID from ElevenLabs
VOICE_ID = "ZQe5CZNOzWyzPSCn5a3c"

# Function to encode the local image file as Base64
def get_base64_image(file_path):
    with open(file_path, "rb") as file:
        data = file.read()
    return base64.b64encode(data).decode()

# Function to set background image using Base64 encoding
def set_background(image_file):
    base64_str = get_base64_image(image_file)
    css = f"""
    <style>
        .stApp {{
            background-image: url("{base64_str}");
            background-size: cover;
            background-position: center;
            background-repeat: no-repeat;
            background-attachment: fixed;
        }}
        .title {{
            text-align: center;
            font-weight: bold;
            color: purple;
            text-shadow: 2px 2px 4px gray;
            font-size: 2.5em;
            margin-top: 20px;
        }}
        h2 {{
            text-align: center;
            color: gray;
            font-size: 1.5em;
```

```
        }}
        .forgot, .register {{
            font-size: 0.9em;
            text-decoration: none;
        }}
        .forgot:hover {{
            color: red;
        }}
        .register:hover {{
            color: green;
        }}
        .footer {{
            text-align: center;
            font-size: 0.8em;
            color: gray;
            margin-top: 50px;
        }}
    </style>
    """
    st.markdown(css, unsafe_allow_html=True)

# Function to create database connection
def create_connection():
    try:
        connection = mysql.connector.connect(
            host='localhost',
            database='linguakids',
            user='root',
            password=''
        )
        if connection.is_connected():
            return connection
    except Error as e:
        st.error(f"Error: {e}")
        return None

# Function for user registration
def register_user(username, password):
    connection = create_connection()
    if connection:
        try:
            cursor = connection.cursor()
            cursor.execute('SELECT * FROM users WHERE name = %s', (username,))
            existing_user = cursor.fetchone()
            if existing_user:
                st.error("Username already exists. Please choose a different one.")
                return False
```

```
            cursor.execute('INSERT INTO users (name, password) VALUES (%s, %s)',
(username, password))
            connection.commit()
            st.success("Registration successful! You can now log in.")
            return True
        except Error as e:
            st.error(f"Error: {e}")
            return False
        finally:
            connection.close()

# Function for user authentication
def authenticate_user(username, password):
    connection = create_connection()
    if connection:
        cursor = connection.cursor()
        cursor.execute('SELECT * FROM users WHERE name = %s AND password = %s',
(username, password))
        user = cursor.fetchone()
        connection.close()
        return user

 #gtts

def app():
    # Set background image (implement set_background() if needed)
    set_background("pic2.jpg")

    if "page" not in st.session_state:
        st.session_state.page = "login"

    if st.session_state.page == "login":
        st.title("User Login")
        username = st.text_input("Username")
        password = st.text_input("Password", type='password')
        focus = st.text_input("Focus")

        if st.button("Login"):
            if username and password:
                user = authenticate_user(username, password)
                if user:
                    st.session_state.update({"focus": focus, "logged_in": True,
"username": username, "page": "gemini_module"})
                    st.rerun()
                else:
                    st.error("Invalid username or password.")
            else:
                st.error("Please enter both username and password.")
```

```
        if st.button("Register Here"):
            st.session_state.page = "register"
            st.rerun()

    elif st.session_state.page == "register":
        st.title("User Registration")
        username = st.text_input("Choose a Username")
        password = st.text_input("Create a Password", type='password')

        if st.button("Register"):
            if not username or not password:
                st.error("All fields are required.")
            else:
                if register_user(username, password):
                    st.session_state.page = "login"
                    st.rerun()

        if st.button("Back to Login"):
            st.session_state.page = "login"
            st.rerun()

    elif st.session_state.page == "gemini_module":
        if 'logged_in' in st.session_state and st.session_state.logged_in:
            st.title(f"Welcome, {st.session_state.username}!")
            st.write(f"Focus on: {st.session_state.focus}")

            if "chat_history" not in st.session_state:
                st.session_state["chat_history"] = []

            chat_placeholder = st.empty()
            user_input = None

            # Speech-to-Text Button Controls
            col1, col2 = st.columns(2)
            with col1:
                if st.button("🎤 Start Recording"):
                    start_capture()
                    threading.Thread(target=capture_audio, daemon=True).start()

            with col2:
                if st.button("🛑 Stop Recording"):
                    user_input = stop_capture()

            if user_input:
                st.session_state["chat_history"].append({"role": "user", "content":
user_input})
```

```
                # Enhanced AI Prompt for Kids' Learning
                prompt_base = (
                    "You are a friendly English tutor for kids, helping them improve
their speaking and writing skills. "
                    "Your job is to: \n"
                    "1 Carefully analyze the given sentence and find any mistakes
(grammar, spelling, punctuation, sentence structure).\n"
                    "2 Provide the corrected sentence in a clear and simple way.\n"
                    "3 Explain the mistakes using fun, kid-friendly language with
examples or storytelling.\n"
                    "4 Make it interactive! Use emojis, playful expressions, and
encouraging words to keep kids engaged.\n"
                    "5 Keep explanations short and easy to understand.\n"
                    "6 If generating video, provide only the corrected sentence and
a very short explanation.\n"
                    "Now, here's the sentence to correct: \n"
                    f"\"{user_input}\""
                    "**Corrected Sentence:** <Provide only the corrected
sentence>\n"
                    "**Explanation:** <Explain the mistakes in a fun and simple
way>"
                )

                conversation_history = "\n".join(
                    [f"{msg['role'].capitalize()}: {msg['content']}" for msg in
st.session_state["chat_history"]]
                )
                prompt = f"{prompt_base}\n\n{conversation_history}"

                # **Generate AI Response**
                model = genai.GenerativeModel('gemini-2.0-flash')
                response = model.generate_content(prompt)
                gemini_reply = response.text.strip()

                # **Extract Corrected Sentence and Explanation**
                corrected_sentence =
gemini_reply.split("**Explanation:**")[0].replace("**Corrected Sentence:**",
"").strip()
                explanation = gemini_reply.split("**Explanation:**")[1].strip() if
"**Explanation:**" in gemini_reply else ""

                # Append AI response to chat history
                st.session_state["chat_history"].append({"role": "assistant",
"content": gemini_reply})

                # **Display Conversation**
                chat_placeholder.empty()
                for msg in st.session_state["chat_history"]:
```

```
                    role = "You" if msg['role'] == "user" else "Gemini"
                    chat_placeholder.write(f"**{role}:** {msg['content']}")

                # **Generate AI Voice Response for Only the Corrected Sentence**
                  # Fixed file path to overwrite
                #audio_file=elevenlabs_tts(corrected_sentence, audio_files)
                tts(corrected_sentence, lang='en', output_file="response.mp3")

                st.audio("response.mp3", format="audio/mp3")

                # **Generate Lip-Sync Video**
                video_input = corrected_sentence + "\n\nShort Explanation: " +
explanation
            video_path, error_message = generate_video(video_input)

                if error_message:
                    st.error(error_message)
                elif video_path:
                    st.success("✅ Lip-Sync Video Generated!")
                    st.video(video_path)

                    # **Provide Download Button**
                    with open(video_path, "rb") as file:
                        st.download_button("⬇️ Download Lip-Sync Video", file,
"lipsync_output.mp4", "video/mp4")
if __name__ == "__main__":
    app()
import torch
import pyaudio
import numpy as np
import threading
from transformers import WhisperProcessor, WhisperForConditionalGeneration

# Device setup
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")

# Load the Whisper model and processor
processor = WhisperProcessor.from_pretrained("openai/whisper-small")
model = WhisperForConditionalGeneration.from_pretrained("openai/whisper-
small").to(device)
model.eval()  # Set the model to evaluation mode

# Audio recording settings
FORMAT = pyaudio.paInt16  # 16-bit resolution
CHANNELS = 1  # Mono channel
RATE = 16000  # Whisper requires 16kHz
CHUNK = 1024  # Buffer size
```

```
# Global variables
audio = pyaudio.PyAudio()
stream = None
recording = False
audio_buffer = b""
audio_thread = None  # Thread for capturing audio


def transcribe_audio(audio_buffer):
    """Transcribe the recorded speech into English text."""
    # Convert audio buffer to NumPy array (normalize to -1.0 to 1.0 range)
    audio_np = np.frombuffer(audio_buffer, dtype=np.int16).astype(np.float32) /
32768.0

    # Process audio input for Whisper
  inputs = processor(audio_np, sampling_rate=RATE, return_tensors="pt")
  input_features = inputs["input_features"].to(device)

  # Generate transcription with English language constraint
  with torch.no_grad():
      predicted_ids = model.generate(
          input_features,
          forced_decoder_ids=processor.tokenizer.get_decoder_prompt_ids(language="en
", task="transcribe")
      )

  # Decode transcription
  transcription = processor.batch_decode(predicted_ids, skip_special_tokens=True)[0]
  return transcription


def capture_audio():
    """Continuously read audio while recording."""
    global stream, recording, audio_buffer

    while recording:
        try:
            audio_chunk = stream.read(CHUNK, exception_on_overflow=False)
            audio_buffer += audio_chunk  # Accumulate audio data
        except Exception as e:
            print(f"Error reading audio: {e}")
            break

    print("Stopped capturing audio.")


def start_capture():
    """Start capturing audio in a separate thread."""
```

```
    global stream, recording, audio_buffer, audio_thread

    if recording:
        print("Already recording...")
        return

    stream = audio.open(format=FORMAT, channels=CHANNELS, rate=RATE, input=True,
frames_per_buffer=CHUNK)
    recording = True
    audio_buffer = b""
    print("Listening...")

    # Start audio capture in a separate thread
    audio_thread = threading.Thread(target=capture_audio)
    audio_thread.start()


    def stop_capture():
        """Stop capturing audio and return transcription."""
        global stream, recording, audio_buffer, audio_thread

        if not recording:
            print("Not recording...")
            return ""

        recording = False
        if audio_thread:
            audio_thread.join()  # Wait for the thread to finish

        stream.stop_stream()
        stream.close()
        print("Processing audio...")

        transcription = transcribe_audio(audio_buffer)
        return transcription


    # Example usage
    if __name__ == "__main__":
        start_capture()
        input("Press Enter to stop recording...\n")
        result = stop_capture()
        print(f"Transcription: {result}")

        # Close the PyAudio instance properly
        audio.terminate()

    from gtts import gTTS
```

```
  import os
  #from pydub import AudioSegment

  def tts(text, lang='en', output_file="response.mp3"):
      """
      Converts the given text into speech and saves it as an MP3 file.

      Parameters:
      - text (str): The text to convert to speech.
      - lang (str): Language for the text-to-speech conversion (default: 'en').
      - output_file (str): Name of the output MP3 file.
      """
      try:
          # Create TTS object
          tts = gTTS(text=text, lang=lang)
          tts.save(output_file)
          # audio = AudioSegment.from_mp3(output_file)

    except Exception as e:
        print(f"An error occurred: {e}")
import streamlit as st
import requests
import json
import os
import time

# Set your API key
GOOEY_API_KEY = "sk-MUzcf4wihx7uqWga2gLYglfe0FPOaFeRwUapXTW1ZSmSGUx0"
API_URL = "https://api.gooey.ai/v2/Lipsync/form/"

# st.title("🎭 AI Lip-Sync Avatar Generator")

# Manually set the local file paths (modify these paths as needed)
#image_path = "./avatar4.jpg"   # Change this to your image file path
#audio_path = "./mp3avatar.mp3"    # Change this to your audio file path

# Run the video generation automatically when the page loads
def generate_video(video_input):
    image_path = "./avatar4.jpg"   # Change this to your image file path
    audio_path = "./response.mp3"

    # Ensure both files exist
    if not os.path.exists(image_path):
        return None, "❌ Error: Image file does not exist!"
    if not os.path.exists(audio_path):
        return None, "❌ Error: Audio file does not exist!"

    print(f"📂 Using image: {image_path}")
```

```
    print(f"🗁 Using audio: {audio_path}")

    with st.spinner("Processing... Please wait ⌛"):
        files = {
            "input_face": open(image_path, "rb"),
            "input_audio": open(audio_path, "rb")
        }

        payload = {"language": "en"}

        try:
            response = requests.post(
                API_URL,
                headers={"Authorization": f"Bearer {GOOEY_API_KEY}"},
                files=files,
                data={"json": json.dumps(payload)}
            )
        finally:
            files["input_face"].close()
         files["input_audio"].close()

     if response.status_code == 200:
         result = response.json()
         if "output" in result and "output_video" in result["output"]:
             video_url = result["output"]["output_video"]

             # Download the video
             video_response = requests.get(video_url)
             if video_response.status_code == 200:
                 output_path = "lipsync_output.mp4"
                 with open(output_path, "wb") as video_file:
                     video_file.write(video_response.content)

                 return output_path, None  # Return video path, no error
             else:
                 return None, "❌ Failed to download video"
         else:
             return None, "❌ Unexpected API response format"
     else:
         return None, f"❌ API Error: {response.status_code} - {response.text}"
```

REFERENCES

[1] Manas Mhasakar, Shikhar Sharma, Apurv Mehra, Utkarsh Venaik, Ujjwal Singhal, Dhruv Kumar, and Kashish Mittal. 2024. Comuniqa : Exploring Large Language Models for Improving English Speaking Skills. In ACM SIG-CAS/SIGCHI Conference on Computing and Sustainable Societies (COMPASS'24), July 08–11, 2024, New Delhi, India. ACM, New York,
NY, USA, 12 pages. https://doi.org/10.1145/3674829.3675082

[2] Chaitali Diwan, Srinath Srinivasa a, Gandharv Suri a, Saksham Agarwal a, Prasad Ram. AI-based learning content generation and learning pathway augmentation to increase learner engagement. 2666-920X/© 2022 Elsevier doi.org/10.1016/j.caeai.2022.100110.

[3] Weipeng Yang, Artificial Intelligence education for young children: Why, what, and how in curriculum design and implementation 2666-920X/© 2022 The Author. Published by Elsevier Ltd doi.org/10.1016/j.caeai.2022.100061.

[4] Jiahong Su, Davy Tsz Kit Ng, Samuel Kai Wah Chu, Artificial Intelligence (AI) Literacy in Early Childhood Education: The Challenges and Opportunities. 2666-920X/© 2023 The Authors. Published by Elsevier Ltd doi.org/10.1016/j.caeai.2023.100124.

[5] Parneet Kaur, Harish Kumar, Sakshi Kausha, Technology-Assisted Language Learning Adaptive Systems: A Comprehensive Review. 2666-3074/© 2023 The Authors. Published by Elsevier Ltd doi.org/10.1016/j.ijcce.2023.09. 002.