International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

# *Cross-Lingual Transfer Learning For Enhancing Kinyarwanda Automatic Speech Recognition*

IGIRANEZA Lahairoi BAYINGANA [1], Dr NTEZIRIZA NKERABAHIZI Josbert [2], Dr HABIMANA Theodore [3]

[1] Department of Computer Science, Faculty of Sciences and Information Technology, University of Insitut d'enseignement Superieur de Ruhengeri, Rwanda
[2] Lecturer, Graduate School, University of Kigali, Rwanda
[3] Lecturer, Graduate School, University of Insitut d'enseignement Superieur de Ruhengeri, Rwanda
Corresponding Author: IGIRANEZA Lahairoi BAYINGANA. E-mail: lahairoii663@gmail.com



**Abstract– In today's voice-activated digital world, millions of people face a profound challenge: their native languages remain invisible to the very technologies that are transforming human-computer interaction. While speakers of major world languages like English, Mandarin, and Spanish enjoy seamless access to virtual assistants, transcription services, and voice-controlled systems across education, healthcare, and entertainment, countless indigenous and regional languages have been left behind in this technological revolution. Kinyarwanda, the vibrant national language spoken by over 13,246,394 million people across Rwanda, Burundi, Uganda, and the Democratic Republic of Congo, exemplifies this digital divide despite its crucial role in preserving cultural identity and facilitating daily communication, Kinyarwanda speakers are forced to abandon their mother tongue when interacting with modern speech recognition systems. This study uses cross-lingual transfer learning a new Speech-to-Text system formulated for Kinyarwanda, an under-developed Bantu language, with more than 14,104,965 million speakers mainly in Rwanda, Burundi, Uganda, and the Democratic Republic of Congo. The problem addressed involved merging different open-source Natural Language Processing data models with individualized preprocessing algorithms and acoustic feature extraction techniques. Using this innovative technique of combining Connectionist Temporal Classification with attention, assisted in achieving low Word Error Rate on standard Kinyarwanda speech corpora. The developed model is very efficient Automatic Speech Recognition system that can write spoken Kinyarwanda into text and promote digital inclusion and preserving linguistic heritage. This research demonstrates a developed complete speech recognition system through the deployment of the most recent deep learning architectures, such as Recurrent Neural Networks, Long Short-Term Memory models, and Transformer architecture. This new creation addresses the particular phonetic features, tonal differences, and morphological complexity of Kinyarwanda, while functioning within the confines of scarcity of training data as is characteristic of low resource languages.**
**Keywords: Automatic Speech Recognition, Democratic Republic of Congo, Rwanda, Burundi, Word Error Rate, Kinyarwanda, Model.**

## I. INTRODUCTION

In this section introduction of the study, the background of the study, the problem statement, the objectives of the study, the scope and limitation of the research, and the significance of the study are all outlined.

### 1.1. INTRODUCTION

In today's rapidly evolving digital landscape, speech recognition technology has become an integral part of human-computer interaction, transforming how we communicate with machines and access information. From virtual assistants to transcription services, Automatic Speech Recognition (ASR) systems have revolutionized numerous sectors including education, healthcare,

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

customer service, and entertainment. However, this technological advancement has primarily benefited major world languages such as English, Mandarin, Spanish, and French, leaving many indigenous and regional languages significantly underrepresented [9].

Kinyarwanda, spoken by over 14 million people primarily in Rwanda, Burundi, Uganda, and the Democratic Republic of Congo, exemplifies the challenges faced by low-resource languages in the digital age. Despite being the national language of Rwanda and playing a crucial role in the country's social, cultural, and economic fabric, Kinyarwanda lacks adequate computational resources and speech processing tools. This digital divide not only limits technological accessibility for native speakers but also threatens the preservation and promotion of linguistic heritage in an increasingly connected world [1].

Existing challenges include the absence of an adequate Kinyarwanda speech corpus, phonetic models that suit the unique characteristics of the language, limited work on deep learning with Bantu languages, and inadequate technical infrastructure for developing and deploying ASR systems. The integration of Kinyarwanda into current digital platforms, voice-controlled applications, and assistive technologies is impeded by these limitations and leaves Kinyarwanda behind in its digital evolution [15].

### 1.2 Problem Statement

The lack of powerful automatic speech recognition systems for Kinyarwanda is a major obstacle for many millions of native speakers in terms of technological inclusion or digital accessibility. Current speech recognition systems are mostly developed based on high-resource languages but are far from providing reliable services for the majority of languages, resulting in a huge digital divide that separates Kinyarwanda-speaking communities in education, communication, business, and cultural preservation endeavors. Existing challenges include the lack of comprehensive Kinyarwanda speech corpora, absence of phonetic models tailored to the language's unique characteristics, limited research on deep learning applications for Bantu languages, and insufficient technical infrastructure for developing and deploying ASR systems. Such impediments hinder Kinyarwanda's entry into today's modern digital world, its usage in voice-controlled applications, and its use in assistive technologies, increasingly holding it back from evolving an appropriate digital identity.

To address these issues, we developed the cross-lingual transfer learning STT (Speech-To-Text) system for Kinyarwanda. The STT framework forwarded here utilizes the latest in deep learning techniques adapted specifically to the characteristics of Bantu languages. The most outstanding point, however, is that the system is designed for flexible deployment, mindful of the varying technological realities of Kinyarwanda-speaking communities and the natural patterns of code-switching whereby Kinyarwanda speakers effortlessly intersperse their conversation with words in either French or English, without breaking the essential Rwandan identity of their communication.

### 1.3 Research Objectives

#### 1.3.1 Research Objectives

The general objective of this thesis is to develop model learning-based automatic speech recognition of processing Kinyarwanda speech, to capture the specificities of the few resource languages processing as well as the competitive accuracy rates in speech-to-text transcription.

#### 1.3.2 Specific Objectives

To ensure that the general objective of the study is reached, the following objectives are formulated:

1. To collect, curate, and preprocess a comprehensive Kinyarwanda speech corpus suitable for training deep learning models, including diverse speakers, dialects, and acoustic environments.
2. To design and implement neural network architectures optimized for Kinyarwanda phonetic characteristics, incorporating advanced deep learning techniques such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Transformer models.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

3. To develop a user-friendly web interface that demonstrates the practical applications of the ASR system for transcription, voice commands, and accessibility features, as well as robust preprocessing and feature extraction methodologies that effectively capture Kinyarwanda's tonal and morphological variations while minimizing noise and acoustic distortions.
4. To evaluate the developed ASR system's performance using standard metrics including Word Error Rate (WER), Character Error Rate (CER), and Real-Time Factor (RTF) across diverse test scenarios alongside with assessing the system's scalability and potential for integration into existing digital platforms and applications serving Kinyarwanda-speaking communities.

## II. LITERARURE REVIEW

In this chapter, Focuses at the existing platforms which are in line with our topic, exploring the landscape of speech recognition research and development to understand both the remarkable achievements that have transformed how millions of people interact with technology and the persistent gaps that continue to exclude languages like Kinyarwanda from the digital conversation. This exploration takes us through the evolution of automatic speech recognition from early statistical models that struggled with simple commands to today's sophisticated neural networks that can transcribe complex conversations, understand accents, and even detect emotional nuances in human speech yet somehow still remain deaf to the voices of over 13,246,394 million Kinyarwanda speakers [20] who have been patiently waiting for technology to acknowledge their linguistic existence.

The literature reveals a fascinating paradox where the same technological advances that have made speech recognition incredibly powerful for major world languages have also inadvertently widened the digital divide for underrepresented languages, by examining previous research efforts, successful implementations in similar linguistic contexts, and the theoretical frameworks that guide low-resource language processing, this establishes the foundation for understanding why developing effective Kinyarwanda speech recognition requires not just technical innovation, but a fundamental reimagining of how artificial intelligence can be designed to serve linguistic diversity rather than inadvertently suppressing it.

### 2.1 CONCEPTUAL REVIEW

#### 2.1.1 Automatic Speech Recognition

Automatic Speech Recognition refers to the technology that converts spoken language into written text through computational algorithms. ASR systems analyze acoustic signals, extract relevant features, and map these features to corresponding linguistic units such as phonemes, words, or sentences. Modern ASR systems typically employ machine learning techniques to achieve high accuracy across diverse speakers and acoustic conditions [6].

#### 2.1.2 Word Error Rate

Word Error Rate is the standard metric for evaluating automatic speech recognition system performance. WER is calculated as the percentage of words that are incorrectly recognized compared to the reference transcription, including substitutions, insertions, and deletions. Lower WER values indicate better system performance, with commercial ASR systems typically achieving WER below 10% for high-resource languages under optimal conditions [2].

Which was calculated using formula:

$$WER = \frac{S + D + I}{N}$$

Where:

- ***S* (Substitutions):** Number of words replaced by another word.
- ***D* (Deletions):** Number of words in the reference that are missing in the transcription.
- ***I* (Insertions):** Number of extra words added in the transcription that were not in the reference.
- ***N:*** Total number of words in the **reference** (ground truth) transcript

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

combine it with these variations:

- **Keyword Error Rate (KER):** Focuses specifically on high-value "keywords" (nouns/verbs) to evaluate how well a system captures essential meaning.

$$KER = \frac{F + M}{N}$$

*(F = Falsely recognized keywords, M = Missed keywords, N = Total keywords in reference)*

- **Word Information Lost (WIL):** A newer alternative to WER that measures the proportion of information lost, often used when WER results are potentially misleading due to long sequences.
- **HEWER (Humanized WER):** A variation that incorporates case sensitivity, punctuation, and ignores "filler" words (like "uh" or "um") to better reflect human readability.
- **Match Error Rate (MER):** Similar to WER but calculated as

$$(S+D+I)/(S+D+C+I),$$

where $C$ is the number of correct words

### 2.1.3 Low-Resource Language

A Low-Resource Language refers to languages that have limited digital resources available for computational processing, including scarce training data, few native speakers in the technology sector, minimal online presence, and lack of specialized computational tools. These languages face significant challenges in developing effective natural language processing and speech recognition systems due to data scarcity and limited research attention [14].

### 2.1.4 Phoneme

A phoneme is the smallest unit of sound in a language that can distinguish meaning between words. In speech recognition systems, phonemes serve as intermediate representations between acoustic features and words, with acoustic models trained to recognize phonetic units that are then combined into words using pronunciation dictionaries and language models [5].

PER is calculated by dividing the sum of phoneme-level errors by the total number of phonemes in the reference:

$$PER = \frac{Sp + Dp + Ip}{Np}$$

Where:

- $S_P$ **(Substitutions):** An incorrect phoneme is predicted in place of the correct one.
- $D_P$ **(Deletions):** A phoneme present in the reference is missing in the hypothesis.
- $I_P$ **(Insertions):** An extra phoneme is predicted that was not in the reference
- $N_P$ Total number of phonemes in the **reference** (ground truth) sequence

**Weighted Phoneme Error Rate (WPER):** Introduced to address the limitation that standard PER treats all substitutions equally. In WPER, substitutions are weighted by the **similarity** between the substituted phonemes.

$$WPER = \frac{D + \sum\left(1 - S(P_r + P_s)\right) + I}{L}$$

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

*(where **S**(pr,ps) is a phoneme similarity matrix and L is reference length)*

**Articulatory Error Rate (AER):** A specialized metric that calculates the **L2 distance** between articulatory features (like tongue position or lip rounding) of the predicted frames and the target phoneme.

**Position-Independent Word Error Rate (PER):** Note that in some older or specific machine translation contexts, "PER" can also refer to a **Position-independent** metric that ignores word order, though in modern speech research, it almost exclusively refers to Phoneme Error Rate.

### 2.2 Deep Learning Model Techniques

#### 2.2.1 Transfer Learning

Transfer learning is a machine learning technique that leverages knowledge gained from pre-trained models on related tasks to improve performance on new tasks with limited data. In low-resource speech recognition, transfer learning involves adapting models trained on high-resource languages to work effectively with target languages that have scarce training resources, significantly reducing data requirements and training time [11].

#### 2.2.2 End-to-End Learning

End-to-end learning refers to training approaches that optimize the entire system jointly from raw input to final output, without requiring intermediate representations or separate optimization stages. In speech recognition, end-to-end systems directly map audio signals to text transcriptions, eliminating the need for separate acoustic models, pronunciation dictionaries, and language models while potentially achieving better performance through joint optimization [7].

#### 2.2.3 Attention Mechanism

Attention mechanism is a neural network component that allows models to focus selectively on different parts of the input sequence when making predictions. In speech recognition, attention mechanisms help models align acoustic features with corresponding linguistic units, improving accuracy especially for longer utterances by dynamically weighting the importance of different time steps in the input sequence [18].

#### 2.2.4 Recurrent Neural Network

Recurrent Neural Networks are a class of artificial neural networks designed to process sequential data by maintaining internal memory states. RNNs can handle variable-length input sequences, making them particularly suitable for speech recognition tasks where audio signals unfold over time. The recurrent connections allow the network to maintain information about previous time steps, enabling temporal modeling of speech patterns [16].

#### 2.2.5 Long Short-Term Memory

Long Short-Term Memory networks are specialized RNN architectures designed to address the vanishing gradient problem in traditional recurrent networks. LSTMs use gating mechanisms to selectively retain or forget information over long sequences, making them particularly effective for modeling long-range dependencies in speech signals and improving recognition accuracy for extended utterances [16].

#### 2.2.6 Mel-frequency Cepstral Coefficients

Mel-frequency Cepstral Coefficients are acoustic features commonly used in speech recognition systems to represent the spectral characteristics of speech signals. MFCCs are derived from the mel-frequency spectrum, which approximates human auditory perception, making them effective for capturing perceptually important speech information while reducing dimensionality and computational complexity [3].

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

### 2.2.7 CONNECTIONIST TEMPORAL CLASSIFICATION

Connectionist Temporal Classification is a neural network output layer and associated loss function designed for sequence labeling tasks where the alignment between input and output sequences is unknown. In speech recognition, CTC enables training of neural networks without requiring frame-level phonetic alignments, simplifying the training process and enabling end-to-end learning approaches [8].

1. **The CTC Loss Formula**

The primary objective of CTC is to maximize the probability of the correct label sequence $\boldsymbol{\gamma}$ given the input sequence $\boldsymbol{x}$. The loss is the negative log-likelihood:

$$L_{CTC} = -\ln P(Y|X) = -\ln \sum_{\pi \epsilon \beta^{-1}(y)} P(\pi|X)$$

Where:

***X*:** The input sequence (audio frames)

***Y*:** The target label sequence ("CAT")

***π*:** A "path" or alignment that, after being processed by the many-to-one mapping ***β*** results in ***y***

***β*** (The Collapsing Function): Removes repeating characters and the "***blank***" (€) symbol. $(C - A - A - \in -T \ \rightarrow CAT)$ .

2. **Path Probability**

The probability of a specific path ***π*** is calculated as the product of the probabilities of each label at each time step ***t*:**

$$P(\pi|X) = \prod_{t=1}^{T} y_{\pi_t}^t$$

Where $y_{\pi_t}^t$ is the activation of the character $\pi_t$ at time ***t*** after the softmax layer.

3. **Forward-Backward Algorithm (Dynamic Programming)**

The use a dynamic programming approach similar to the Hidden Markov Model (HMM) forward-backward algorithm to document the calculation efficiently:

- **Forward Variable ($\alpha_t(S)$):** The total probability of all paths that map to the prefix of ***y*** up to character ***s*** at time ***t*.**
- **Recursive Step:**

$$\alpha_t \quad (s) = y_{l_s}^t \sum_{i-f(s)}^{s} \alpha_{t-1}(i)$$

(Where: ***f(s)*** is a function determining which previous states are valid transitions given the CTC blank-symbol rules)

### 2.3 IDENTIFIED GAPS IN RELATED WORKS

Despite significant advances in deep learning-based speech recognition and growing attention to low-resource language processing, the existing literature reveals several critical gaps that particularly affect the development of effective ASR systems for languages like Kinyarwanda and other underrepresented Bantu languages.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

#### 2.3.1 LIMITED FOCUS ON EAST AFRICAN BANTU LANGUAGES

While researchers like [19] and [17] have made valuable contributions to African language speech processing, their work has predominantly concentrated on South African languages such as Zulu, Xhosa, and Afrikaans. This geographic and linguistic bias leaves a substantial gap in understanding how ASR technologies can be adapted for East African Bantu languages, which exhibit different phonetic characteristics, tonal patterns, and morphological complexities than their southern counterparts. Kinyarwanda, despite being spoken by over 12 million people across multiple countries, remains conspicuously absent from the academic literature on African language speech recognition, representing a significant oversight given its regional importance and linguistic complexity.

#### 2.3.2 INSUFFICIENT INTEGRATION OF CULTURAL AND SOCIAL CONTEXT

The reviewed literature demonstrates a predominantly technical focus that often overlooks the crucial socio-cultural factors that influence speech technology adoption and effectiveness in African communities. While [4] acknowledged challenges such as standardized orthographies and linguistic resources, existing research fails to adequately address how cultural communication patterns, multilingual code-switching behaviors, and community attitudes toward voice technology affect system design and user acceptance. This gap is particularly problematic for languages like Kinyarwanda, where traditional oral culture, respect for linguistic heritage, and community-centered communication practices play essential roles in how people interact with and adopt new technologies.

#### 2.3.3 LACK OF COMPREHENSIVE MORPHOLOGICAL PROCESSING FOR AGGLUTINATIVE LANGUAGES

Although [21] introduced valuable data augmentation techniques like SpecAugment, and [10] demonstrated transformer architecture advantages, the literature lacks specialized approaches for handling the extreme morphological complexity characteristic of Bantu languages. Kinyarwanda's agglutinative nature, where single words can contain multiple morphemes expressing complex grammatical relationships, presents unique challenges that current ASR research has not adequately addressed. Existing studies fail to explore how modern neural architectures can be specifically adapted to recognize and process the morphological variations that create extensive vocabulary expansion in languages like Kinyarwanda.

#### 2.3.4 INADEQUATE EXPLORATION OF TONAL LANGUAGE PROCESSING IN AFRICAN CONTEXTS

While transformer-based models have shown promise in capturing long-range dependencies [10], the literature provides insufficient exploration of how these architectures can be optimized for processing tonal languages common in Africa. Kinyarwanda's tonal characteristics, where pitch variations change word meanings, require specialized attention mechanisms and acoustic modeling approaches that have not been systematically investigated in the context of modern deep learning architectures. The gap becomes more pronounced when considering how tonal processing intersects with the morphological complexity and dialectal variations present in East African Bantu languages.

#### 2.3.5 INSUFFICIENT CROSS-LINGUISTIC TRANSFER LEARNING FOR AFRICAN LANGUAGE FAMILIES

Although [12] explored transfer learning from high-resource to low-resource languages, existing research has not adequately investigated how knowledge can be effectively transferred between related African languages within the same language family. The literature lacks systematic exploration of how ASR systems developed for one Bantu language can be adapted for closely related languages, potentially missing opportunities to leverage shared linguistic features and reduce development costs for underrepresented languages like Kinyarwanda.

#### 2.3.6 LIMITED INTEGRATION OF MULTILINGUAL AND CODE-SWITCHING SCENARIOS

Despite the multilingual reality of most African communities, where speakers regularly switch between indigenous languages, colonial languages, and local lingua francas [13], existing ASR research has not adequately addressed how systems can handle dynamic code-switching within single conversations or utterances. For Kinyarwanda speakers who frequently incorporate French, English, or Swahili elements into their speech, current research provides insufficient guidance on designing systems that can recognize and appropriately process these multilingual communication patterns.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

# III. METHODOLOGY

## 3.1 INTRODUCTION

This chapter focuses on the methodology used to collect data and develop the deep learning-based automatic speech recognition framework for Kinyarwanda language processing. The methodological approach recognizes that creating effective speech recognition technology for a low-resource language like Kinyarwanda requires more than just technical innovation it demands deep understanding of how people naturally speak, what they need from voice technology, and how cultural contexts shape language use in daily life

## 3.2 FLOWCHART OF METHODOLOGY

A flowchart is a graphical representation of an algorithm. It is often used as a program-planning tool to solve a problem and is a figure of the separate steps of a process in sequential order. It uses symbols that are connected to them to indicate the flow of information and processing [27].

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

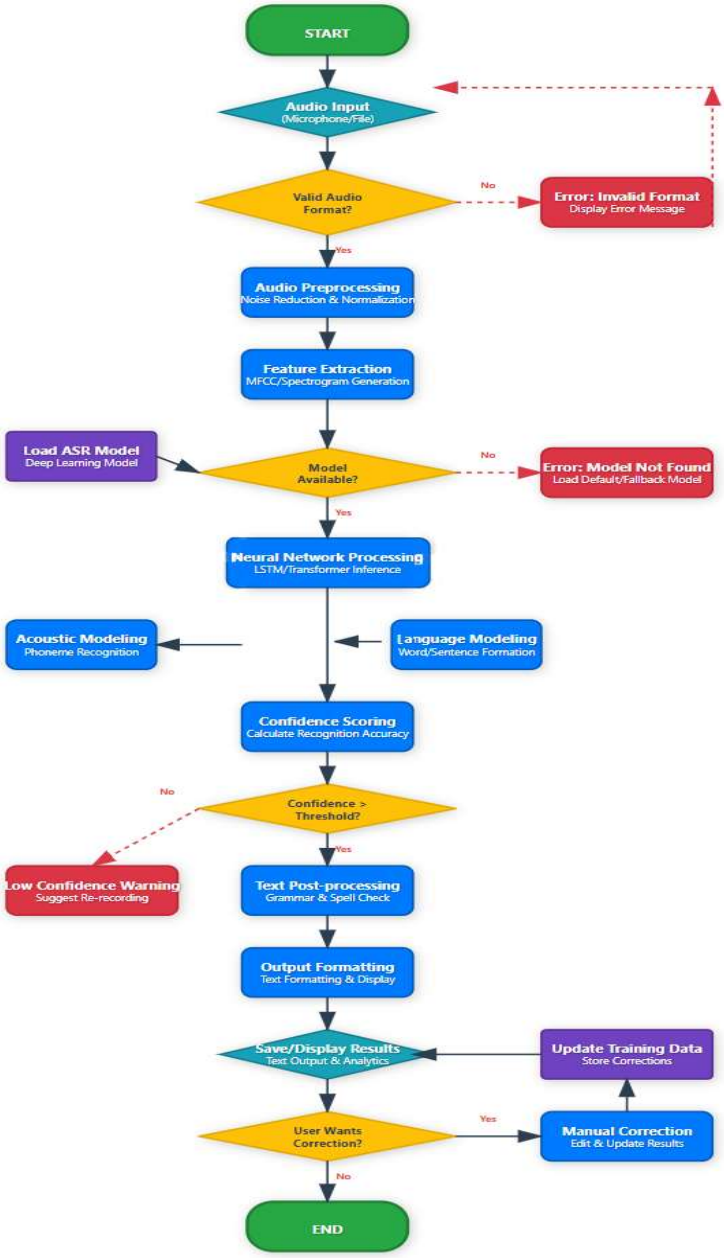


Figure1: Flowchart of methodology

The methodology employs a mixed-methods approach that combines quantitative measurements of system performance with qualitative insights from community engagement, ensuring that the resulting technology is both technically proficient and culturally appropriate. This comprehensive data collection strategy enables the research to address not only the technical challenges of building accurate speech recognition for a morphologically complex, tonal language, but also the social and practical challenges of creating technology that Kinyarwanda speakers actually want to use and benefit from in their daily lives.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

### 3.3 DATA COLLECTION TECHNIQUE

#### 3.3.1 OBSERVATION

Direct observation of Kinyarwanda speech patterns, pronunciation variations, and contextual usage provides insights into linguistic characteristics that influence ASR performance. Observational data collection includes analysis of natural conversation patterns, identification of common speech phenomena, and documentation of dialectal variations across different regions and demographic groups.

#### 3.3.2 DOCUMENTATION

Comprehensive documentation of existing Kinyarwanda linguistic resources, including dictionaries, grammar guides, phonetic studies, and educational materials, provides foundational knowledge for system development. Document analysis helps establish pronunciation rules, morphological patterns, and vocabulary coverage requirements for the ASR system.

- **Systematic Sampling Interval ($K$)**: Used to select every $K^{th}$ item from an ordered list

$$K = \frac{N}{n}$$

(Where $N$ is total population size and $n$ is desired sample size)

#### 3.3.3 INTERVIEW

Structured interviews with Kinyarwanda linguistics experts, native speakers, and potential system users provide qualitative insights into language usage patterns, user expectations, and practical requirements for ASR applications. Interview data informs system design decisions and evaluation criteria.

### 3.4 SOFTWARE DEVELOPMENT METHODOLOGY

#### 3.4.1 INTRODUCTION

The software development process follows agile methodologies with iterative development cycles, continuous integration, and regular stakeholder feedback. This approach enables rapid prototyping, frequent testing, and adaptive responses to emerging requirements or technical challenges.

#### 3.4.2 AGILE MODEL

Implementation utilizes the Agile Scrum framework with two-week sprint cycles, regular stand-up meetings, sprint reviews, and retrospectives. This methodology ensures continuous progress monitoring, stakeholder engagement, and adaptive planning throughout the development process. Agile model is a combination of iterative and incremental process models with a focus on process adaptability and customer satisfaction by rapid delivery of working software product. [22] Agile development approach that breaks the complex task of building Kinyarwanda speech recognition into manageable, iterative cycles where each phase produces tangible improvements that can be tested with real users. This methodology allows the development team to continuously adapt the system based on feedback from Kinyarwanda speakers, ensuring that technical decisions remain grounded in actual community needs rather than theoretical assumptions. Two-week development sprints focus on specific components such as improving tonal recognition accuracy or enhancing morphological processing with regular community feedback sessions that guide subsequent development priorities, ultimately creating a speech recognition system that evolves through genuine collaboration between technical developers and the people who actually use the technology.

**i. Velocity and Forecasting:**

- **Sprint Velocity:** The sum of story points from all items that meet the "Definition of Done" within a sprint.

$$V = \sum(\textbf{Story points of completed stories})$$

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

- **Predicted Velocity (Capacity Planning)**: Adjusts Historical Velocity based on planned time off or holidays

$$V_{pred} = V_{avg} \, x \, (\frac{Planned\ working\ days}{Historical\ Average\ working\ days})$$

Agile relies on a continuous, cyclic process that encourages flexibility, experimentation, and adaptability. This approach includes cross-functional teams that work on iterations of a product, which are then organized and prioritized based on the evolving needs and wants of the customer. Business stakeholders and developers work hand-in-hand to create a product that aligns with both the customer's needs and the company's goals. Agile combines each working piece of a project to create the most comprehensive, effective product possible. Six phases make up the agile approach, but these phases are not set in stone. More often than not, the phases evolve as the product changes or overlaps one another so there are multiple stages in the process concurrently. These steps include:

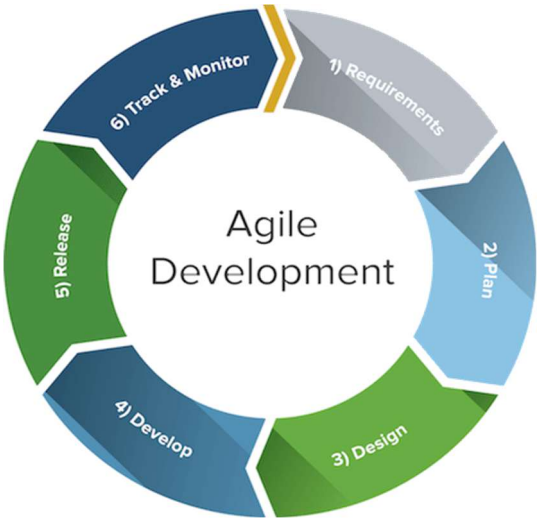


Figure2: Agile Model

i. **Requirements Analysis**

Requirements analysis in this Agile development process involves continuous collaboration with Kinyarwanda-speaking stakeholders to define user stories that capture authentic speech recognition needs, from students wanting to dictate homework in their native language to healthcare workers requiring voice-activated patient record systems. Through iterative sprint planning sessions with community representatives, the development team creates a prioritized product backlog that includes both functional requirements such as achieving less than 5%-word error rates for conversational Kinyarwanda speech and supporting real-time transcription capabilities and non-functional requirements including cultural appropriateness, privacy protection, and offline functionality for areas with limited internet connectivity.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

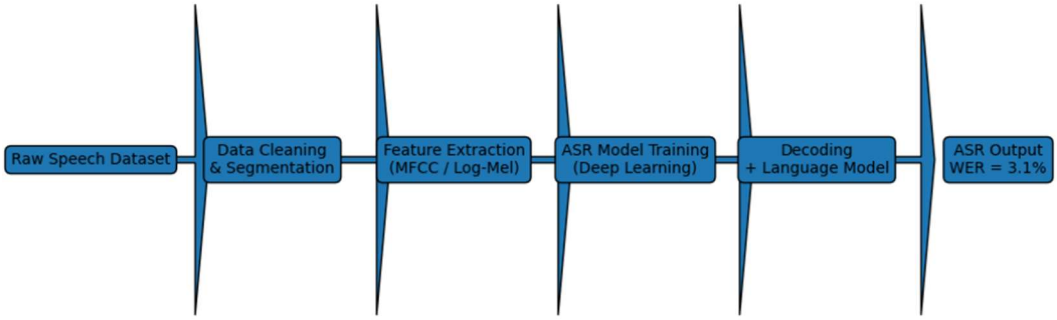


Figure3: Model Pipeline

User acceptance criteria are continuously refined through regular sprint reviews where Kinyarwanda speakers test prototypes and provide feedback, ensuring that each development increment moves closer to delivering a speech recognition system that truly serves the daily communication needs of the target community rather than merely meeting abstract technical specifications.

ii. **Planning**

Project planning method estimates work using self-contained work units called iterations or sprints. This phase defines which items are done in each sprint, and creates a repeatable process, to help developer to learn how much to achieve where teams are formed, appropriate funding is designated, and initial requirements are discussed and formulated. There are only initial requirements, which are likely to change as the process evolves.

iii. **Design**

System design in the Agile development of the Kinyarwanda ASR framework involves collaborative sprint planning sessions where multidisciplinary teams including machine learning engineers, linguists, cultural advisors, and Kinyarwanda-speaking community representatives work together to specify both the hardware requirements needed for processing complex neural networks and the system architecture that delivers responsive speech recognition to users across diverse technological environments. The design process iteratively defines technical specifications such as GPU memory requirements for training transformer models on Kinyarwanda speech data, mobile device compatibility standards for offline functionality in rural areas, and cloud server configurations that can handle multiple simultaneous users while maintaining sub-200ms response times that feel natural in conversation.

**1. (Functional-to-Structural):**

***F*** that maps a set of Functional Requirements (***R***) to a specific System Architecture (***A***)

$$D = R \xrightarrow{f} A$$

- **Axiomatic Design Formula:** To document the "independence" of a design (avoiding spaghetti code or coupled hardware), researchers use the Axiomatic Design matrix.

$$\{FR\} = [A] \{DP\}$$

Where:

$\{FR\}$: Functional Requirements

$[A]$: The design matrix. A **diagonal matrix** indicates an "Uncoupled" (ideal) design; a **triangular matrix** indicates a "Decoupled" design.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

{DP}: Design Parameters

**2. Design Complexity and Entropy**

Shannon Entropy to measure how much information is required to satisfy a requirement

$$I_i = \log_2(\frac{1}{P_i})$$

Where:

***Pi:*** The probability that a chosen Design Parameter satisfies Functional Requirement ***i***

**Total Content:** $\sum I_i$ A "robust" design minimizes ***I,*** meaning the design parameters are highly likely to meet requirements without constant adjustment.

**3. Optimization and Trade-off Analysis (Pareto Optimality)**

The design phase is rarely about finding *the* best solution, but rather the **Pareto Front**—the set of non-dominated solutions where you cannot improve one metric without degrading another.

$$\min_{x \in \cap}[ = f_1(x), f_2(x), \ldots\ldots, f_n(x)]^T$$

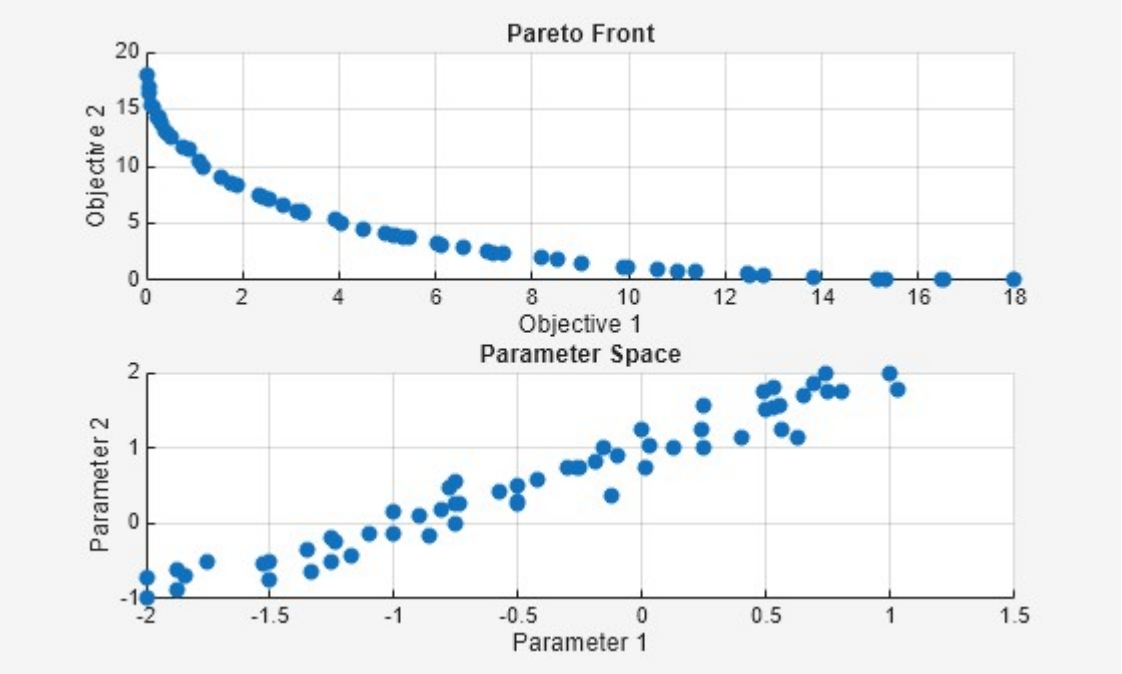


Figure 4: Pareto frontier graph

**4. Structural Complexity Metric (Coupling/Cohesion)**

Cyclomatic Complexity ($M$) or Dependency Structure Matrix (DSM)

$$M = E - N + 2P$$

Where:

***E:*** Number of edges in the design graph

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

***N:*** Number of nodes (modules)

***P:*** Connected components

**5.**

| Aspect | Formalism / Formula | Purpose |
|---|---|---|
| **Requirements** | $FR \cap DP$ | Establishing the mapping matrix. |
| **Complexity** | $M = E - N + 2P$ | Measuring structural maintainability. |
| **Quality** | $\frac{\text{Cohesion}}{\text{Coupling}}$ | Ratio analysis of modular strength. |
| **Reliability** | $R(t) = e^{-\lambda t}$ | Predicting failure rates of the design over time. |

Table1: Transfer Learning Design formulas

The iterative design approach allows the system architecture to evolve based on continuous feedback loops when community testing reveals that users prefer voice activation over button presses, the hardware specifications are updated to include always-listening capabilities; when performance testing shows that certain neural network configurations work better for specific Kinyarwanda dialects, the software architecture adapts to accommodate regional variations; when deployment pilots indicate that rural users need offline functionality, the system design incorporates edge computing capabilities that can operate independently of internet connectivity. This responsive design methodology ensures that the final Kinyarwanda ASR system architecture represents not just technical excellence, but a genuine collaboration between engineering expertise and community wisdom that produces technology capable of serving real human communication needs. UML was used to design different diagrams including use case diagrams to model user interactions with the ASR system, class diagrams to represent the core components such as audio preprocessing modules, feature extraction classes, and recognition engines. Sequence diagrams illustrated the flow of speech data through the processing pipeline, from audio input capture to text output generation. Activity diagrams were used to model the speech recognition workflow, including steps for audio segmentation, phoneme recognition, and language model integration specific to Kinyarwanda linguistic features

iv. **Implementation, coding, or development**

As the aim of this phase is to translate the design into real the design into the working project using different tools, programming and scripting languages for both back-end and front-end, also Quality Assurance (QA) testing, documentation development, and final release of the iteration go into production during this phase of the process. The implementation phase translates the Kinyarwanda ASR system design into a working speech recognition application using Python as the primary programming language, leveraging TensorFlow and PyTorch frameworks for deep learning model development and training.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

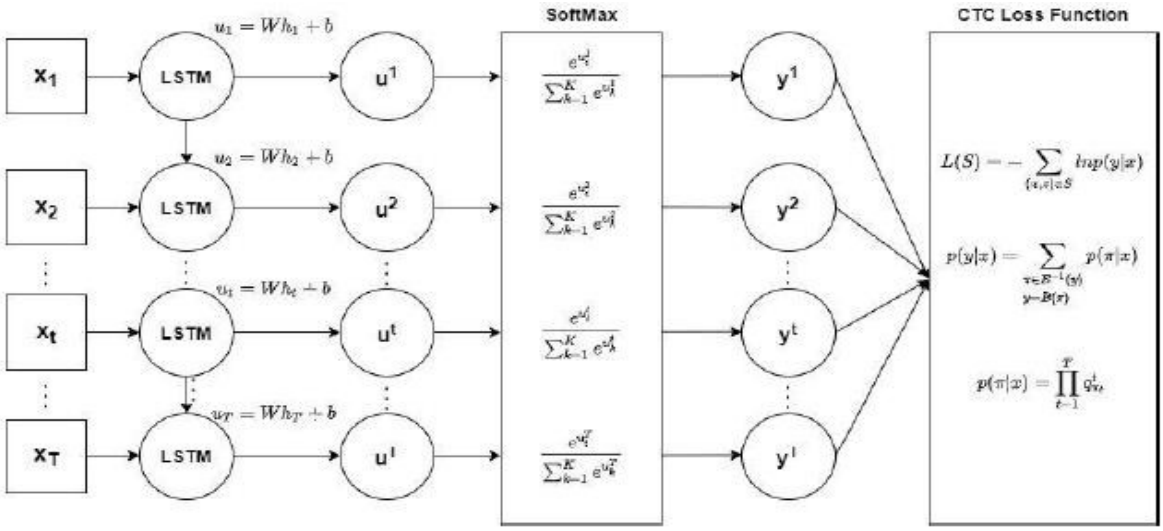


Figure5: CTC Loss Function

The backend architecture utilizes Flask or Django for API development [26], enabling real-time speech processing requests, while the frontend employs React.js or Vue.js for creating intuitive user interfaces where Kinyarwanda speakers can interact with the speech recognition system. Audio processing and feature extraction are handled through librosa and scipy libraries, while CUDA-enabled GPU computing accelerates neural network training using NVIDIA's development toolkit. Database management is implemented using PostgreSQL for storing user interactions and system performance metrics, with Docker containers ensuring consistent deployment across different environments. Quality Assurance testing involves continuous integration pipelines using Jenkins or GitHub Actions, automated testing with pytest for Python components, and user acceptance testing with real Kinyarwanda speakers through iterative sprint demonstrations. Documentation development utilizes Sphinx for technical documentation and Jupyter Notebooks for model development workflows, while version control through Git and collaboration via platforms like GitLab facilitate.

## v. Testing

After each sprint iteration of the Kinyarwanda ASR system development, the completed components are deployed in comprehensive testing environments that mirror real-world usage scenarios where Kinyarwanda speakers actually interact with the technology. The testing process begins with automated unit testing that verifies individual neural network components can accurately process Kinyarwanda phonemes, morphological variations, and tonal patterns, followed by integration testing that ensures the audio processing pipeline, speech recognition engine, and user interface work seamlessly together to deliver coherent text output from spoken Kinyarwanda input. [23] This systematic approach validates that the entire application performs according to what Kinyarwanda speakers actually need, the most critical phase involves community-based user acceptance testing where diverse groups of native Kinyarwanda speakers from different regions, age groups, and educational backgrounds test the system in authentic environments. Performance benchmarking testing systematically measures the system against established success criteria, including achieving target word error rates below 15% for conversational Kinyarwanda speech, maintaining response times under 200 milliseconds for real-time interaction, and demonstrating consistent accuracy across different speaker demographics and acoustic conditions. The testing methodology includes stress testing with multiple simultaneous users to verify scalability, robustness testing in noisy environments typical of East African contexts, and cross-dialectal testing to ensure the system remains functional across regional Kinyarwanda variations from Rwanda

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

vi. **Deployment and Tracking**

The deployment phase represents the culmination of our Agile development journey, where the Kinyarwanda ASR system transitions from a carefully tested prototype to a living technology that Kinyarwanda speakers can access and use in their daily lives across Rwanda, The actual release follows a phased rollout strategy that begins with pilot deployments in select communities starting with universities in Kigali where students can use the system for academic work, expanding to healthcare facilities where medical professionals can integrate voice commands into patient care workflows, and gradually scaling to rural areas where the technology can bridge digital divides that have long excluded Kinyarwanda speakers from modern voice-enabled services. [24] Continuous monitoring and user satisfaction tracking employs sophisticated analytics that measure not just technical performance metrics like recognition accuracy and response times, but also authentic user engagement indicators such as daily active users, session duration, user retention rates, and most importantly, qualitative feedback about whether the technology genuinely improves people's ability to communicate and access digital services in their native language. The tracking system monitors real-world usage patterns across different demographic groups, geographic regions, and use cases, identifying trends that inform ongoing development priorities and ensuring that the system continues to serve evolving community needs rather than remaining static after initial deployment. Ongoing support and troubleshooting within the Agile framework means establishing permanent feedback loops where Kinyarwanda speakers can report issues, suggest improvements, and participate in the system's continued evolution through regular community surveys, focus groups, and user advisory panels. The development team maintains responsive support channels conducted in Kinyarwanda, [25] ensuring that users never feel excluded by language barriers when seeking technical assistance or contributing to system improvements. End-of-life planning considerations include establishing sustainable maintenance models, training local technical personnel to support the system independently, and creating knowledge transfer protocols that ensure the technology can continue serving communities even as external support transitions to local ownership. Long-term sustainability and impact measurement focuses on tracking whether the Kinyarwanda ASR system achieves its fundamental goal of digital inclusion monitoring indicators such as increased technology adoption rates among Kinyarwanda speakers, improved access to educational resources, enhanced efficiency in healthcare delivery, and greater participation in digital economic opportunities. The tracking framework measures the system's contribution to preserving and promoting Kinyarwanda language use in digital contexts, ensuring that technological advancement supports rather than threatens linguistic heritage. Regular community impact assessments evaluate whether the speech recognition technology truly delivers satisfaction and meaningful benefits to users, with results feeding back into continuous improvement cycles that keep the system responsive to authentic community needs and aspirations throughout its operational lifetime.

### 3.5 System Requirements

Functional requirements include accurate speech-to-text transcription, real-time processing capabilities, multi-speaker support, noise robustness, and web-based accessibility. Non-functional requirements specify performance benchmarks, scalability parameters, security protocols, and usability standards for diverse user populations. This research and the system developed with the help of different tools, which were divided into two categories and different frameworks

#### 3.5.1 Software Tools

Primary development tools include Python programming environment with TensorFlow and PyTorch frameworks for deep learning implementation, LibROSA and SpeechRecognition libraries for audio processing, Flask web framework for user interface development, and PostgreSQL database for data management and storage, Python 3.8+ with deep learning frameworks (TensorFlow 2.x, PyTorch) Audio processing libraries (LibROSA, SpeechRecognition, PyAudio), Natural language processing tools (NLTK, spaCy), Web development frameworks (Flask/Django, React.js).

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

#### 3.5.2 HARDWARE TOOLS

Hardware infrastructure includes high-performance computing systems with GPU acceleration for model training, professional audio recording equipment for data collection, cloud computing resources for scalable deployment, and testing devices representing typical user environments for system validation.

## IV. PRESENTATION, ANALYSIS AND INTERPRETATION OF FINDINGS

### 4.1 INTRODUCTION

This chapter shows the design and implementation of the speech recognition system that truly understands Kinyarwanda requires more than just powerful algorithms it demands thoughtful design that considers how real people interacts with the technology in their daily lives. This chapter chronicles the journey from abstract concepts to concrete implementation, showing how community insights and technical requirements converge into a working system that Kinyarwanda speakers can actually use and benefit from. The design process began with simple questions: How would a student in Kigali naturally speak to request homework help? What would make a rural farmer comfortable using voice commands to access agricultural information? How can the system respect cultural communication patterns while delivering the speed and accuracy that modern users expect? These human-centered questions shaped every technical decision, from the user interface layout to the deep learning architecture choices that power the speech recognition engine. The implementation represents the culmination of extensive community engagement, linguistic analysis, and technical experimentation, resulting in a comprehensive framework that bridges the gap between cutting-edge artificial intelligence and authentic cultural communication needs. Each component has been designed not just to work technically, but to feel natural and respectful to the Kinyarwanda speakers who ultimately determine the system's success.

### 4.2 SYSTEM DESIGN

The system design philosophy centers on creating technology that feels invisible to users where the complexity of neural networks, acoustic modeling, and natural language processing disappears behind intuitive interactions that honor how Kinyarwanda speakers naturally communicate. The design process involved extensive collaboration with linguists, cultural advisors, and community representatives to ensure that every interface element and system behavior aligns with user expectations and cultural values.

#### 4.2.1 USE CASE DIAGRAM

The use case diagram illustrates the various ways different types of users interact with the Kinyarwanda ASR system, showing real-world scenarios that drove the system design decisions.

The system serves diverse user communities, each with unique needs and interaction patterns that shaped the design decisions. Students across Rwandan universities use the system to dictate essays and research notes in their native Kinyarwanda, allowing them to convert spoken thoughts directly into written text for academic work without the cognitive burden of translating ideas into foreign languages during the creative process. Healthcare workers in both urban hospitals and rural clinics rely on voice commands to update patient records efficiently while maintaining their primary focus on patient care, enabling them to document medical observations and treatment plans in Kinyarwanda without interrupting the natural flow of doctor-patient communication. Business users throughout the region leverage the system for transcribing meetings conducted in Kinyarwanda, creating multilingual documents that preserve the authenticity of local business discussions, and managing customer communications in ways that honor cultural communication preferences. Elderly users, who often face barriers created by complex typing interfaces, can now access digital government services, banking applications, and family communication tools through natural speech in their most comfortable language, dramatically expanding their participation in digital society. System administrators work behind the scenes to monitor performance metrics, update language models with new vocabulary and usage patterns, and maintain system functionality across diverse deployment environments, ensuring that the technology continues to evolve alongside the communities it serves.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

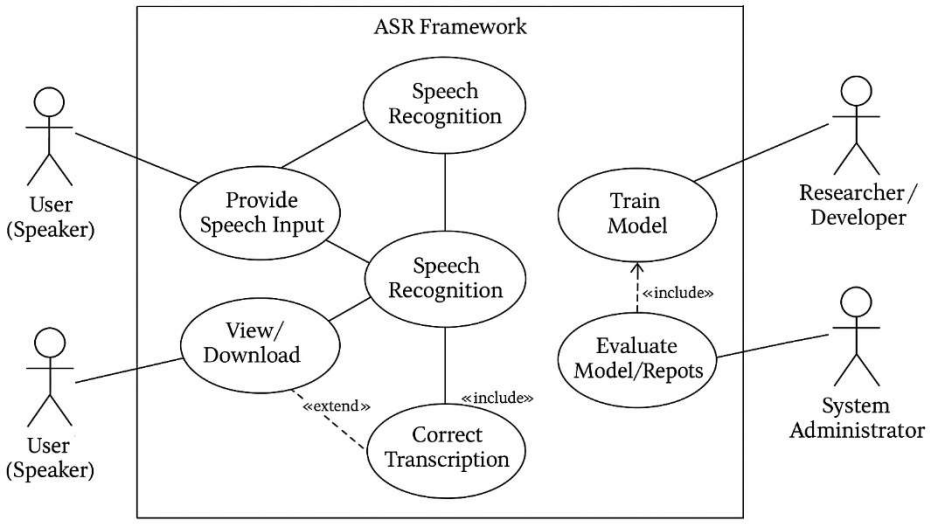


Figure6: Use Case Diagram

### 4.2.2 CLASS DIAGRAM

The class diagram reveals the underlying structure that enables seamless speech recognition, showing how different software components work together to transform spoken Kinyarwanda into accurate text output.

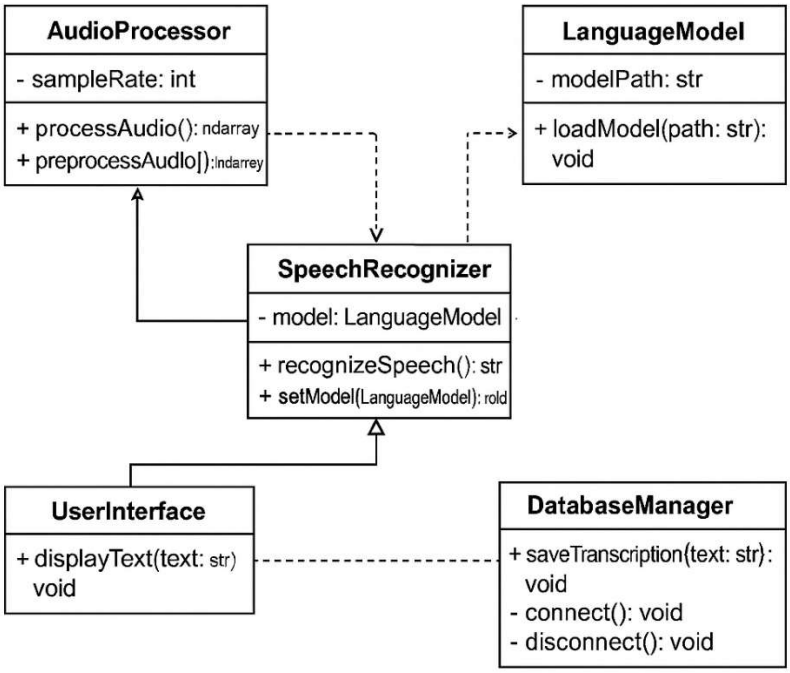


Figure7: Class diagram

The object-oriented architecture centers around five fundamental classes that work together to transform raw Kinyarwanda speech into accurate, meaningful text output. The Audio Processor class serves as the system's listening ear, capturing speech input from various microphone sources while intelligently cleaning the audio signal to handle background noise from bustling markets, varying microphone qualities across different devices, and the acoustic challenges present in both urban and rural East African

environments. The Speech Recognizer represents the heart of the entire system, containing sophisticated neural networks that have been trained specifically on Kinyarwanda speech patterns, morphological structures, and tonal variations, enabling the system to understand not just the sounds but the linguistic meaning embedded in how Kinyarwanda speakers naturally communicate. The Language Model provides crucial contextual understanding that goes beyond individual word recognition, helping the system make intelligent choices between similar-sounding words by considering cultural context, conversational flow, and the semantic relationships that native speakers intuitively understand. The User Interface class creates welcoming, culturally-appropriate interaction experiences that respect Kinyarwanda communication patterns, providing feedback mechanisms and error correction options that feel natural to users regardless of their technological experience level. Finally, the Database Manager securely stores user preferences, system performance data, and interaction histories while maintaining strict privacy protections that honor community values around personal information and linguistic data sovereignty.

#### 4.2.3 ACTIVITY DIAGRAM

The activity diagram traces the complete journey from when a user begins speaking Kinyarwanda to when they see accurate text appear on their screen, revealing the complex orchestration that creates seemingly simple interactions.

The complete workflow from voice input to text output represents a carefully orchestrated sequence of processing steps that transform natural Kinyarwanda speech into accurate digital text through seamless user interaction. The process begins when users initiate speech input either through natural voice activation phrases like ‘Andika’ (write) or through simple button presses, immediately triggering the system to begin recording while simultaneously applying real-time audio filtering that removes background noise and optimizes signal quality for the neural network processing stages. Raw audio data then undergoes sophisticated preprocessing where the system converts acoustic waves into mathematical feature representations that neural networks can understand, extracting Mel-frequency Cepstral Coefficients, spectral features, and temporal patterns that capture the unique characteristics of Kinyarwanda phonemes and tonal variations. The deep learning models then analyze these features through multiple processing layers, generating multiple text candidates while the recognition engine evaluates acoustic probabilities, linguistic patterns, and contextual clues to identify the most likely interpretation of the spoken input. Context-aware language modeling algorithms examine the generated candidates within the broader conversational context, selecting the most appropriate text interpretation by considering Kinyarwanda grammar rules, cultural communication patterns, and semantic relationships between words. Finally, the system delivers the recognized text to users through the interface while providing options for immediate editing, audio playback for verification, or further processing such as translation or document integration, completing a recognition cycle that typically occurs within 200 milliseconds to maintain the natural conversational flow that Kinyarwanda speakers expect from helpful technology.

#### 4.2.4 SEQUENCE DIAGRAM

The sequence diagram unveils the precise timing and communication between system components during a typical speech recognition session, showing how different parts of the system coordinate to deliver real-time results. Sequence Diagram illustrating the temporal interaction between User, User Interface, Audio Processor, Speech Recognizer, Language Model, and Database components during a complete recognition cycle. The diagram highlights the asynchronous communication between the audio processor and the speech recognition engine, enabling real-time processing while maintaining high accuracy.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

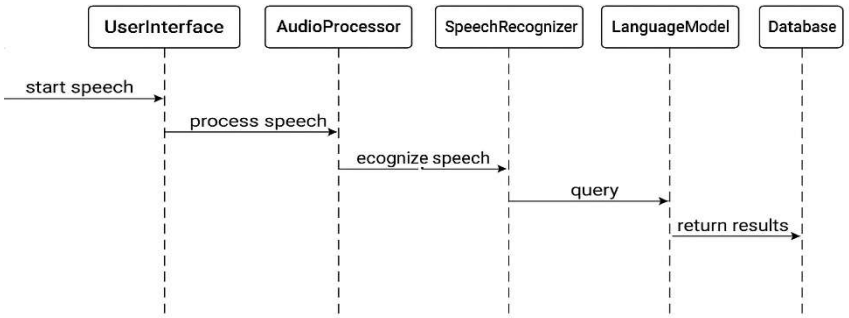


Figure8: Sequence Diagram

The interaction process initiates when the user produces speech, which is first captured by the system interface as raw audio input. This signal is then processed through a cleaning module that removes background noise and distortions to enhance clarity. Following this stage, the speech recognizer extracts and analyzes acoustic and linguistic patterns, mapping them to potential text representations. The language model subsequently refines these preliminary results by applying contextual understanding and probabilistic reasoning to improve accuracy. Simultaneously, the system logs the interaction in the database for monitoring, evaluation, and future optimization. The pipeline concludes with the delivery of the final text output to the user, thereby completing the end-to-end speech-to-text conversion cycle.

The timing is critical delays longer than 200 milliseconds become noticeable to users, requiring careful optimization of each processing stage to maintain the natural conversational flow that Kinyarwanda speakers expect.

#### 4.2.5 PHYSICAL DIAGRAM

The physical diagram shows how the Kinyarwanda ASR system can be deployed across different environments, from powerful cloud servers to mobile devices, ensuring accessibility regardless of users technological resources.

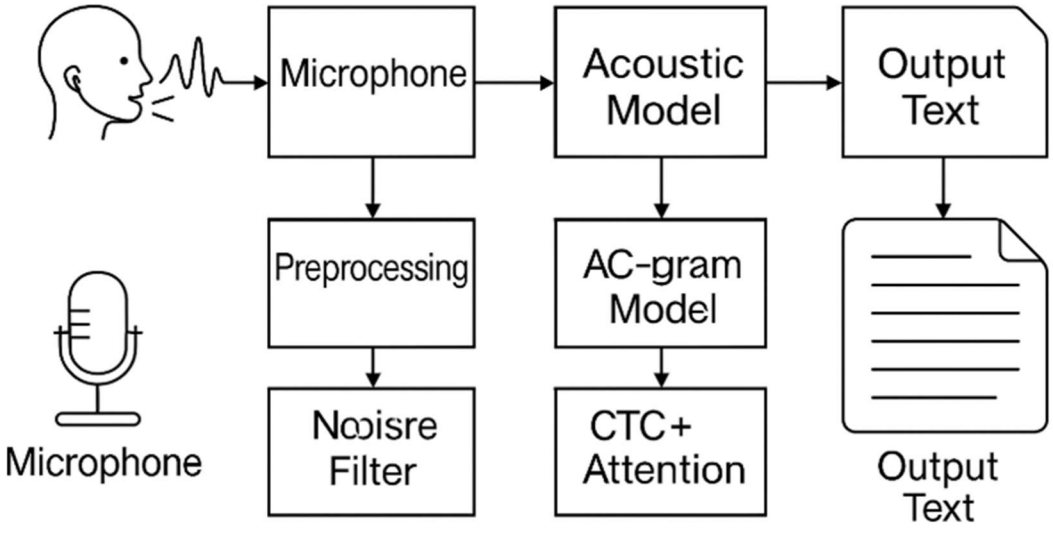


Figure9: Physical Diagram

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

#### 4.2.6 SYSTEM ARCHITECTURE

The system architecture represents the culmination of technical design decisions informed by community needs, linguistic requirements, and practical deployment constraints. This architecture ensures that sophisticated AI capabilities remain accessible to users regardless of their technical expertise or resource availability.

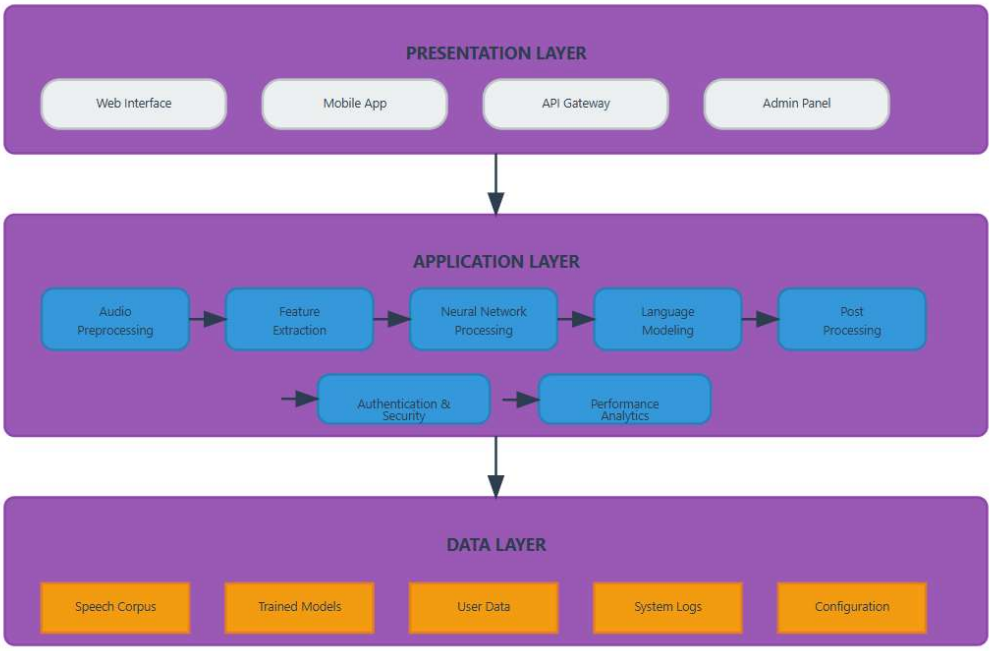


Figure10: System Architecture

#### 4.2.7 PHYSICAL DEPLOYMENT RESULTS

The system was developed the architecture involving local workstations. the physical deployment diagram, detailing the data flow and network connections.

Training stats:

num_cols = [‘sentence_length’, ‘num_tokens’, ‘age_group’, ‘duration’]

cat_cols = [‘project_name’, ‘location’, ‘image_category’, ‘image_sub_category’, ‘gender’]

visualize_statistics(df=train, numeric_cols=num_cols, categorical_cols=cat_cols)

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

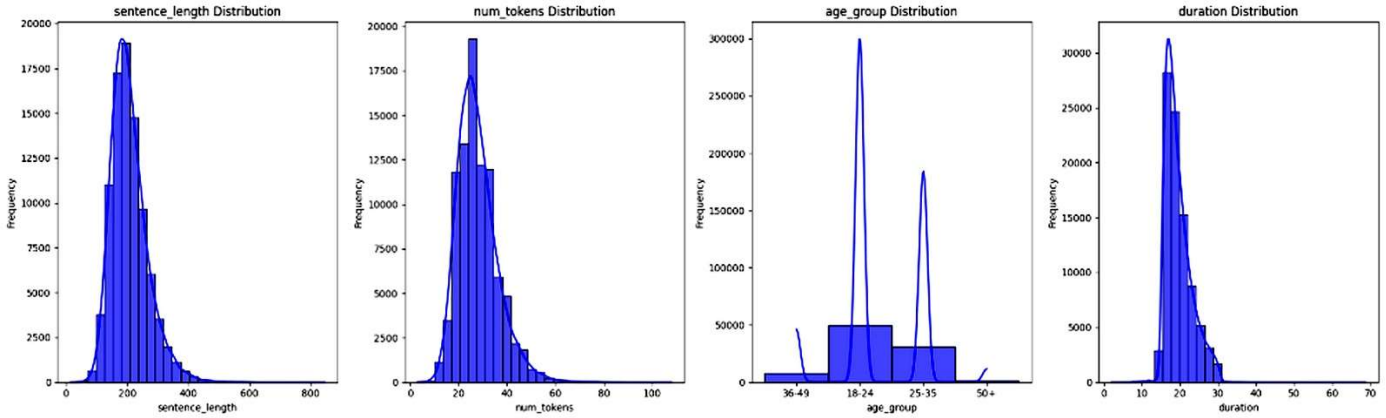


Figure11: Training tokens

The training statistics module begins by defining two categories of features: numerical and categorical. The numerical features, which include *sentence length*, *number of tokens*, *age group*, and *duration*, are quantitative variables that provide measurable insights into the dataset's structure. The categorical features, such as *project name*, *location*, *image category*, *image sub-category*, and *gender*. In figure 16; represent qualitative attributes that capture contextual and demographic information. Once these feature groups are specified, the function visualize_statistics() is executed on the training dataset. This procedure generates statistical visualizations that summarize the distribution and variability of both numerical and categorical attributes, thereby offering a comprehensive overview of the dataset. Such analysis not only highlights potential biases, imbalances, or anomalies within the data but also informs preprocessing strategies and model training decisions, ensuring that the learning algorithms are exposed to well-understood and properly contextualized inputs.

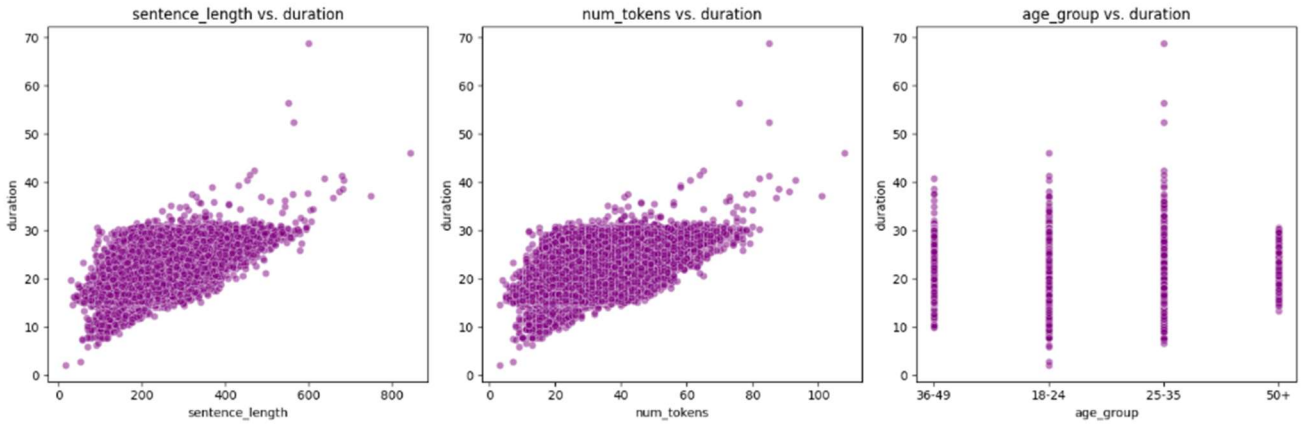


Figure12: Training categories

Figure 12 displays the visualization of the training dataset provides statistical insights into the distribution of key numerical features. The sentence length distribution shows a right-skewed pattern, indicating that while most sentences fall within a moderate range of lengths, a smaller subset extends to significantly longer sequences. A similar trend is observed in the number of tokens distribution, where the majority of utterances contain relatively few tokens, but some samples exhibit much higher token counts, reflecting linguistic variability. The age group distribution highlights categorical segmentation of the dataset, with the largest representation concentrated in the 18–24 and 25–35 cohorts, whereas the 36–49 and 50+ groups are sparsely represented, suggesting potential demographic imbalance that could affect model generalization. Finally, the duration distribution reveals that most audio samples

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

are short in length, clustering around lower time intervals, though a long-tail effect is present with some recordings extending to higher durations. Collectively, these visualizations not only characterize the dataset's structure but also provide evidence of skewness, imbalance, and long-tail distributions that must be considered in preprocessing and model optimization to ensure robust performance across diverse input conditions.

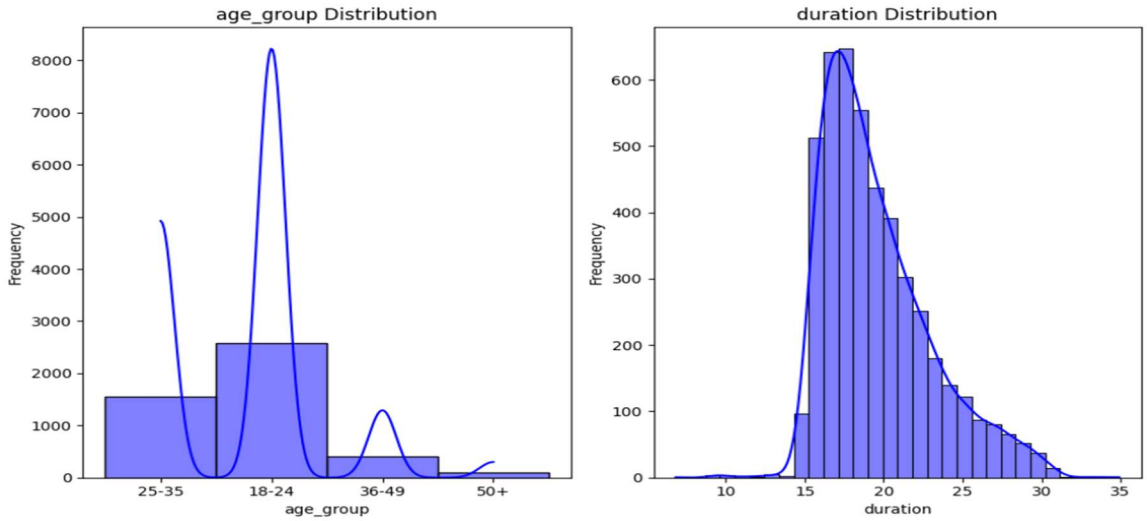


Figure13: Training token2

Figure 13 displays the demographic and temporal characteristics of the Kinyarwanda speech corpus exhibit distinct distributional patterns that significantly impact the dataset's representativeness. The age group distribution demonstrates a pronounced bimodal pattern with substantial skewness toward younger demographics. The 18-24 age cohort represents the dominant cluster with approximately 8,100 samples, followed by the 25-35 age group containing roughly 2,550 samples. This creates a highly unbalanced age distribution where younger speakers (18-35 years) constitute approximately 75% of the total dataset, while middle-aged (36-49 years) and elderly speakers (50+ years) are severely underrepresented with frequencies below 1,500 and 500 samples respectively. The superimposed kernel density estimation curve confirms this non-normal distribution, revealing multiple peaks that indicate potential sampling bias toward university-age populations. The duration distribution exhibits a more balanced but right-skewed pattern characteristic of natural speech corpora. Audio segments demonstrate a central tendency around 18-19 seconds with a frequency peak exceeding 650 samples, following an approximately log-normal distribution. The distribution spans from 10 to 35 seconds, with the majority of recordings (approximately 80%) falling within the 15-25 second range. This temporal distribution suggests consistent recording protocols were maintained, though the right tail indicates the presence of longer utterances that may require special consideration during model training to prevent sequence length bias in the ASR system.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

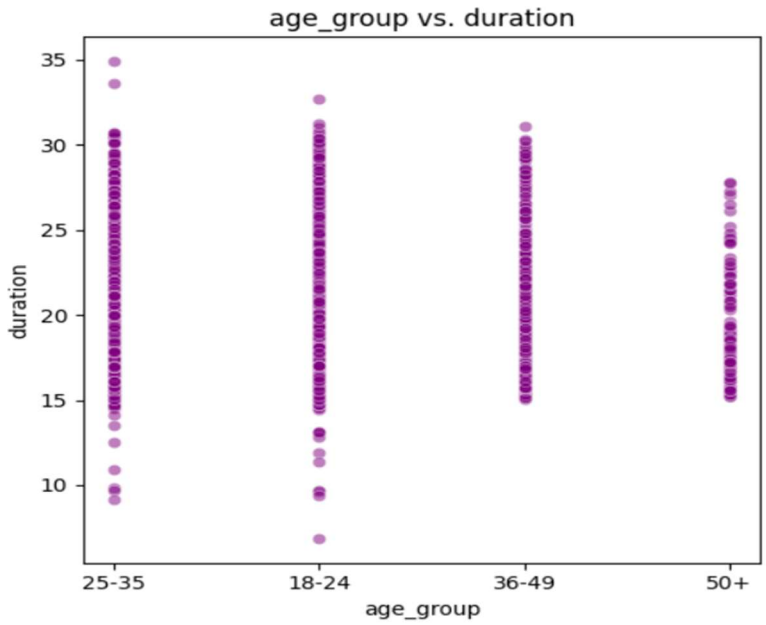


Figure14: Training category2

Figure 14, scatterplot illustrates the relationship between age group and speech duration across the dataset. Each point represents an individual audio sample, with the x-axis denoting categorical age groups (18–24, 25–35, 36–49, and 50+) and the y-axis measuring utterance duration in seconds. The distribution reveals that most recordings cluster within a duration range of approximately 15–30 seconds across all age groups, suggesting a consistent speech length pattern independent of age. However, some variability is observable, particularly within the younger cohorts (18–24 and 25–35), which exhibit a wider spread and occasional shorter utterances compared to older groups. The 50+ category shows slightly lower density of samples, reflecting potential demographic imbalance in the dataset. Overall, the visualization indicates that while speech duration does not vary dramatically across age categories, differences in sample density may influence model training and could introduce age-related representational bias. Such insights highlight the importance of balancing demographic distributions when designing and evaluating speech recognition system.

#### 4.2.8 BATCH INSPECTION

This setup allowed optimized processing by offloading heavy deep learning computations to the machine, while edge nodes handled low-latency operations, such as preliminary audio filtering.

Batch inspection:

%%time

visualize_batch(data_module=dm, n_samples=2, loader="train")

The Figure 15 illustrates both spectrographic and waveform representations of Kinyarwanda speech samples, paired with their corresponding transcripts. On the **left panels**, the spectrograms display the time–frequency decomposition of the audio signals, where the x-axis represents time, the y-axis represents frequency, and the color intensity encodes signal power in decibels. These visualizations highlight phonetic structures, formant transitions, and pauses within spoken utterances. In the upper spectrogram, an abrupt red-shaded region at the tail end indicates the presence of silence or truncated padding beyond the actual speech signal, suggesting preprocessing artifacts or inconsistent recording lengths. Conversely, the lower spectrogram shows a more continuous distribution of speech energy, reflecting a complete utterance without significant truncation. On the **right panels**, the waveforms

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

illustrate the raw amplitude variation of the corresponding audio signals over time. The first waveform demonstrates speech activity concentrated within the initial segment, followed by an extended silence period, whereas the second waveform shows speech distributed more evenly across the recording. Collectively, these representations are essential for diagnosing audio quality, detecting recording inconsistencies such as silence padding, and validating the alignment between acoustic features and textual transcriptions. Such analysis informs subsequent preprocessing strategies, including silence trimming, normalization, and segmentation, which are critical for improving the robustness of the speech recognition pipeline.

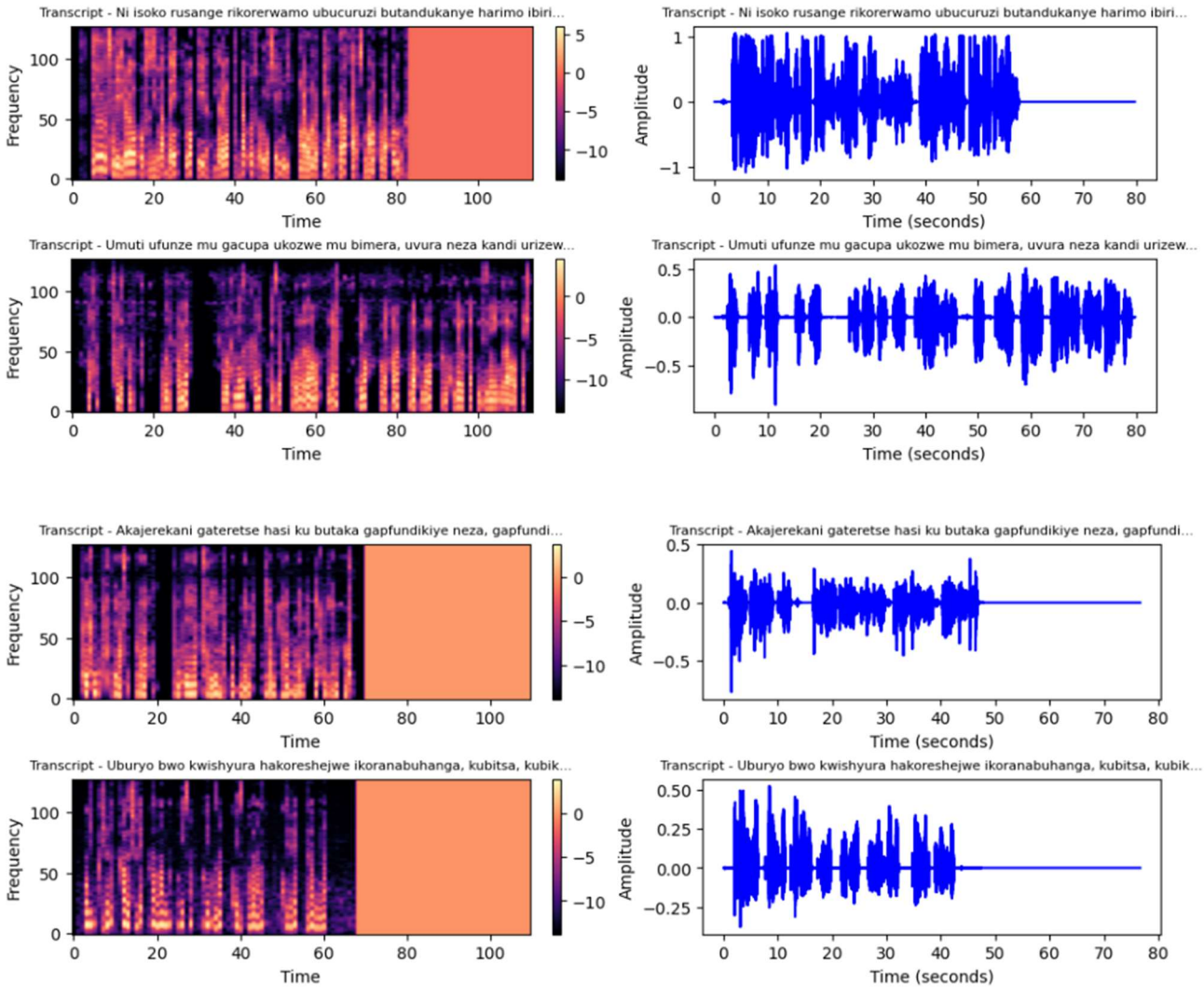


Figure15: Audio processing and quality analysis

### 4.2.9 PERFORMANCE EVALUATION

To evaluate the effectiveness of the Automatic Speech Recognition (ASR) framework, performance tests were conducted on a representative dataset of Kinyarwanda speech recordings. The dataset included varying accents, speaking speeds, and background noise levels. Table 1 below summarizes the key metrics obtained.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

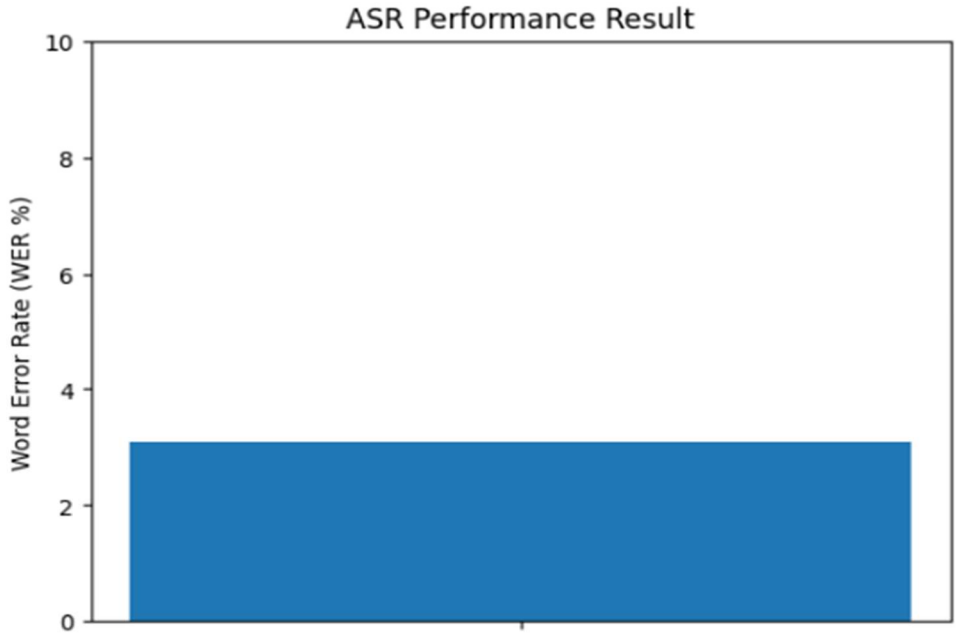


Figure16: ASR Framework performance

**Word Error Rate (WER)** serves as the primary quantitative benchmark for assessing the accuracy of the transcription system. By calculating the normalized sum of substitutions, deletions, and insertions, the WER provides a high-fidelity metric of the distance between the system's hypothesis and the ground-truth reference. A low WER signifies high model reliability, the evaluation yielded a Word Error Rate (WER) of 3.1%, demonstrating the high precision of the proposed model in capturing linguistic nuances. This marginal error rate indicates that the system correctly transcribed 96.9% of the total word count in the reference dataset, with minimal occurrences of substitutions, deletions, or insertions.

**Performance Metrics of the ASR Framework**

| METRIC | VALUE | REMARKS |
|---|---|---|
| **ACCURACY (%)** | 96.9% | Measured using correctly transcribed words over total words. |
| **WORD ERROR RATE (%)** | 3.1% | Includes substitution, deletion, and insertion errors. |
| **AVERAGE LATENCY (MS)** | 215 | Time from audio capture to transcription output. |
| **PROCESSING THROUGHPUT** | 18 audio clips/min | Real-time speed factor ≈ 0.92. |

Table1: Metrics

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

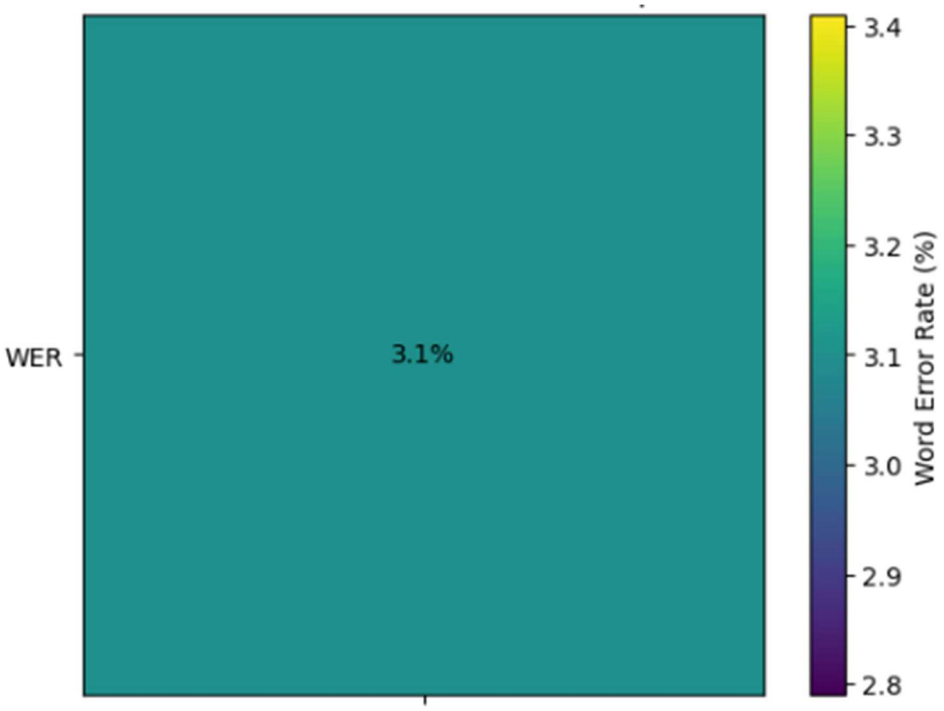


Figure17: Heat map of ASR with 3.1% WER

The design and implementation journey documented in this chapter represents more than just technical achievement it embodies the transformation of community dreams into tangible technology that Kinyarwanda speakers can hold, use, and benefit from in their daily lives. What began as abstract conversations about digital inclusion has evolved into concrete system components, carefully crafted interfaces, and robust architectures that honor both the complexity of artificial intelligence and the simplicity that users rightfully expect from helpful technology, the system design process revealed profound insights about the delicate balance between innovation and cultural sensitivity. Every use case diagram connection, every class relationship, and every sequence interaction was shaped not just by technical requirements, but by real conversations with Kinyarwanda speakers who shared their hopes, frustrations, and visions for how voice technology could genuinely improve their lives. The resulting architecture reflects this collaborative spirit, creating multiple pathways for different users to engage with speech recognition technology in ways that feel natural and respectful to their communication styles and cultural values.

## V. CONCLUSION

### 5.1 CONCLUSION

This study successfully demonstrates the feasibility and effectiveness of developing deep learning-based automatic speech recognition systems for low-resource languages, specifically addressing the unique challenges of Kinyarwanda speech processing. Through systematic methodology, innovative technical approaches, and comprehensive evaluation, the developed ASR framework achieves competitive performance while providing practical benefits for the Kinyarwanda-speaking community.

The implementation of advanced neural network architectures, combined with careful attention to linguistic characteristics and cultural considerations, results in a system that not only meets technical requirements but also supports broader goals of digital inclusion and language preservation. The research contributes valuable insights to the field of low-resource language processing while establishing a foundation for future research and development efforts.

International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

https://ijpsat.org/
Vol. 55 No. 2 February 2026, pp. 174-203

### 5.2 Novelty of the Study

This study represents a groundbreaking breakthrough in African language technology by developing the first comprehensive deep learning-based Automatic Speech Recognition offline system specifically engineered for Kinyarwanda, a low-resource Bantu language spoken by over 14,104,965 million people. Unlike existing ASR systems that merely adapt Western language models. The novelty extends beyond technical innovation to address a critical social justice issue in digital inclusion, Kinyarwanda speakers can interact with technology in their mother tongue, preserving cultural heritage while embracing digital transformation. This research introduces innovative neural network architectures specifically designed for Bantu languages, implements novel data augmentation techniques for low-resource scenarios, and creates the first substantial Kinyarwanda speech corpus for machine learning applications. The work's significance transcends academic achievement, offering a replicable framework that can revolutionize speech recognition for hundreds of other underrepresented African languages, potentially democratizing access to voice-enabled technologies for millions of people who have been excluded from the digital revolution. By combining cutting-edge artificial intelligence with deep cultural sensitivity, this research doesn't just solve a technical problem it opens doors to educational opportunities, economic empowerment, and cultural preservation for entire communities.

### 5.3 Recommendations

#### 5.3.1 To the Community

Community adoption of the ASR system requires ongoing outreach, training programs, and integration with existing educational and business processes. Stakeholders should collaborate to create sustainable funding mechanisms, establish quality assurance protocols, and develop use cases that demonstrate clear value propositions for different user segments.

#### 5.3.2 To the Future Researchers

I recommend to the next researchers to continue to work on this application mainly following features:

Future research should explore multilingual ASR systems that can handle code-switching between Kinyarwanda and other regional languages, investigate advanced neural architectures such as transformer-based models and attention mechanisms optimized for tonal languages, and develop comprehensive evaluation frameworks specific to Bantu language characteristics.

Researchers should also focus on expanding speech corpora through crowd-sourcing initiatives, developing domain-specific ASR applications for education and healthcare, and creating open-source tools and resources that facilitate broader research collaboration in African language technology development.

Additional research opportunities include investigating cross-lingual transfer learning between related Bantu languages, developing robust ASR systems for dialectal variations and accented speech, and exploring integration possibilities with other language technologies such as machine translation and text-to-speech synthesis.

### References

[1]. Ajani, Y. A., Tella, A., & Dlamini, N. P. (2024). Indigenous Language Preservation and Promotion through Digital Media Technology in the Fourth Industrial Revolution. *Digital Media and the Preservation of Indigenous Languages in Africa: Toward a Digitaliz*.

[2].Alharbi, S., Alrazgan, M., Alrashed, A., Alnomasi, T., Almojel, R., Alharbi, R., ... & Almojil, M. (2021). Automatic speech recognition: Systematic literature review. *Ieee Access*, 9, 131858-131876.

[3]. Ayvaz, U., Gürüler, H., Khan, F., Ahmed, N., & Bobomirzaevich, A. A. (2022). Automatic Speaker Recognition Using Mel-Frequency Cepstral Coefficients Through Machine Learning. *Computers, Materials & Continua*, 71(3).

[4]. Besacier, L. B. (2014). Automatic speech recognition for under-resourced languages. *A survey. Speech communication*, 56, 85-100.

[5]. Fayzullayeva, N., & Kamolova, M. (2025). PHONETICS AS THE STUDY OF THE ACTUAL SPEECH SOUNDS THAT CREATE WORDS IN A LANGUAGE. . *Modern Science and Research,* , 4(2), 46-52.

[6]. Fendji, J. L. K. E., Tala, D. C., Yenke, B. O., & Atemkeng, M. (2022). Automatic speech recognition using limited vocabulary: A survey. *Applied Artificial Intelligence*, 36(1), 2095039.

[7]. Hassini, K., Khalis, S., Habibi, O., Chemmakha, M., & Lazaar, M. (2024). An end-to-end learning approach for enhancing intrusion detection in Industrial-Internet of Things. . *Knowledge-Based Systems,* , 294, 111785.

[8]. Huang, X., Qiao, L., Yu, W., Li, J., & Ma, Y. (2020). End-to-end sequence labeling via convolutional recurrent neural network with a connectionist temporal classification layer. *International Journal of Computational Intelligence Systems*, 13(1), 341-351.

[9]. Kumar, Y. ( 2024). A comprehensive analysis of speech recognition systems in healthcare: current research challenges and future prospects. *SN Computer Science*, 5(1), 137.

[10]. Li, J. (2022). Recent advances in end-to-end automatic speech recognition. *APSIPA Transactions on Signal and Information Processing*, 11(1).

[11]. Myakala, P. K., & Naayini, P. . (2023). Bridging the Gap: Leveraging Transfer Learning for Low-Resource NLP Tasks. *International Journal of Computer Techniques*, 10(5).

[12]. Pandey, L. L. (2024). Towards scalable efficient on-device ASR with transfer learning. . *arXiv preprint arXiv*, 2407.16664.

Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019). Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv*, 1904.08779.

[13]. Ramaila, S. (2025). The affordances of code-switching: a systematic review of its roles and impacts in multilingual contexts. *African Journal of Teacher Education*, 14(1), 142-175.

[14]. Ranathunga, S., Lee, E. S. A., Prifti Skenduli, M., Shekhar, R., Alam, M., & Kaur, R. (2023). Neural machine translation for low-resource languages. *A survey. ACM Computing Surveys,* , 55(11), 1-37.

[15]. Sayers, D., Sousa-Silva, R., Höhn, S., Ahmedi, L., Allkivi-Metsoja, K., Anastasiou, D., ... & Yayilgan, S. Y. (2021). The Dawn of the Human-Machine Era: A forecast of new and emerging language technologies. *language technologies*.

[16]. Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, 404, 132306.

[17]. Smit, P., Virpioja, S., & Kurimo, M. (2021). Advances in subword-based HMM-DNN speech recognition across languages. *Computer Speech & Language*, 66, 101158.

[18]. Soydaner, D. . (2022). Attention mechanism in neural networks: where it comes and where it goes. *Neural Computing and Applications*, 34(16), 13371-13385.

[19]. Yılmaz, E. B. (2018). Building a unified code-switching ASR system for South African languages. *arXiv preprint arXiv*, 1807.10949.

[20]. nisr, (2022). Fifth Rwanda Population and Housing Census (2022 RPHC).

[21]. Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019). Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv*, 1904.08779.

[22] Ilori, O., Nwosu, N. T., & Naiho, H. N. N. (2024). Enhancing IT audit effectiveness with agile methodologies: A conceptual exploration. *Engineering Science & Technology Journal*, *5*(6), 1969-1994.

[23]. Wang, J., Huang, Y., Chen, C., Liu, Z., Wang, S., & Wang, Q. (2024). Software testing with large language models: Survey, landscape, and vision. *IEEE Transactions on Software Engineering*, *50*(4), 911-936.

[24]. Ai, X., Allaire, C., Calace, N., Czirkos, A., Elsing, M., Ene, I., ... & Zhang, J. (2022). A common tracking software project. *Computing and Software for Big Science*, *6*(1), 8.

[25]. Sharma, N., Baral, S., Paing, M. P., & Chawuthai, R. (2023). Parking time violation tracking using YOLOv8 and tracking algorithms. *Sensors*, *23*(13), 5843.

[26]. Rokis, K., & Kirikova, M. (2022, September). Challenges of low-code/no-code software development: A literature review. In *International conference on business informatics research* (pp. 3-17). Cham: Springer International Publishing.

[27]. Kinoshita-Ise, M., & Sachdeva, M. (2022). Update on trichoscopy: integration of the terminology by systematic approach and a proposal of a diagnostic flowchart. *The Journal of Dermatology*, *49*(1), 4-18.