JIMMA UNIVERSITY

JIMMA INSTITUTE TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATICS

MASTER PROGRAM IN ARTIFICIAL INTELLIGENCE

A RESEARCH ON

AFAAN OROMO LARGE VOCABULARY CONTINUOUS SPEECH RECOGNITION MODEL

BY

GELANA ABDISA

ADVISORS: GETACHEW MAMO (Ph.D.)

HABTAMU TAKELE (Ms.C.)

JUNE, 2024

JIMMA, ETHIOPIA

JIMMA UNIVERSITY
JIMMA INSTITUTE OF TECHNOLOGY
FACULTY OF COMPUTING AND INFORMATICS
MASTER PROGRAM IN ARTIFICIAL INTELLIGENCE

AFAAN OROMO LARGE VOCABULARY CONTINUOUS SPEECH RECOGNITION MODEL

BY:

GELANA ABDISA

ADVISORS: GETACHEW MAMO

HABTAMU TAKELE

A THESIS SUBMITTED TO JIMMA UNIVERSITY, INSTITUTE OF TECHNOLOGY, FACULTY OF COMPUTING AND INFORMATICS IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF MASTERS OF SCIENCE IN ARTIFICIAL INTELLIGENCE

JIMMA, ETHIOPIA

Declaration

This thesis is my original work and has not been presented as research in any university

Name: Gelana Abdisa Signature: _____ Date: _____

This thesis has been submitted for examination with my approval as a supervisor.

Name of Advisors	Signature	Date
1. Getachew Mamo (Ph.D.)		Date: _____
2. Habtamu Takele (M.Sc.)		Date: _____

Approval

The undersigned hereby certify that we have thoroughly read and enthusiastically recommend to the Faculty of Computing and Informatics, Artificial Intelligence Chair, the acceptance of the thesis submitted by Gelana Abdisa. The thesis is titled "Afaan Oromo Large Vocabulary Continuous Speech Recognition model". We believe this research work fulfills the requirements for the award of a Master's Degree in Artificial Intelligence.

Name of Supervisor _____**Signature** _____ **Date**: _____

Name of Internal Examiner: Dr Tibebe.B (PhD) **Signature** _____ **Date**_____

Name of Head of chair: _____**Signature** _____ **Date**_____

Approved by faculty of computing and informatics Thesis Examination members

Name	**Signature**	**Date**
1._____	_____	_____
2._____	_____	_____
3._____	_____	_____

Acknowledgment

I would like to express my deepest gratitude to Jesus Christ, the Son of God, for the gift of His life and salvation. His sacrifice has saved me from the depths of hell, and I am eternally grateful for His love and grace. Jesus, I am truly grateful for the numerous miracles that have occurred in my life and within my family.

I would like to express my deepest gratitude to my advisor, Getachew Mamo (Ph.D.), for his invaluable guidance, mentorship, and expertise throughout the completion of this thesis. His profound knowledge and unwavering support have been instrumental in shaping the direction of this thesis. I am truly grateful for his constant encouragement, insightful feedback, and dedication to my academic and professional growth.

I would like to express my deepest gratitude to my advisor, Mr. Habtamu Takele (M.Sc.), for his valuable contributions and assistance. His expertise in the field of deep learning and his commitment to excellence have greatly enriched this work. I am thankful for his constructive feedback, collaborative spirit, and willingness to share his knowledge and insights.

I would like to express my deepest gratitude to my loving parents for their unwavering support, encouragement, and sacrifices throughout my academic journey. Their unconditional love, belief in my abilities, and constant prayers have been a source of strength and inspiration. They have always been there for me, providing emotional support and understanding during the challenging times. I am truly blessed to have parents who have instilled in me the values of perseverance, dedication, and hard work. Their guidance and wisdom have shaped me into the person I am today, and I am forever grateful for their presence in my life.

I also want to thank Jimma University Afaan Oromo department MA students Garoma and Gemechu for supporting me a lot during dataset preparation for the study.

Contents

Acronomy

AI: Artificial Intelligence

ASR: Automatic Speech Recognition

DTW: Dynamic Time Warping

ETV: Ethiopian Television

FFNN: Feed-Forward Neural Network

GMM: Gaussian Mixture Model

HMM: Hidden Markov Model

LSTM: Long Short-Term Memory

LVCSR: Large Vocabulary Continuous Speech Recognition

LPC: Linear Predictive Coefficient

MFCC: Mel-frequency cepstral coefficients

ML: Machine Learning

NLP: Natural Language Processing

NN: Neural Networks

OBN: Oromia Broadcasting Network

RNN: Recurrent Neural Networks

VOA: Voice of America

WER : Word Error Rate

List of Tables

Abstract

Automatic Speech recognition(ASR) is a system that translates spoken language into written text. The purpose of this study was to develop an Afaan Oromo speech-to-text recognition model using Speech Transformer, a state-of-the-art deep learning algorithm. Our study primarily aimed at developing a large vocabulary continuous speech recognition model for the Afaan Oromo language with the latest deep learning named Speech Transformer. Most studies in Afaan Oromo ASR used classical machine learning models and only one tried combining Recurrent Neural Networks (RNNs) with Convolutional Neural Networks (CNNs) as a hybrid approach. However, these existing techniques had difficulties in accurately transcribing varied and complex speaking styles found in Afaan Oromo and suffered from slow training due to the lack of parallelization during training time. We have put into consideration these limitations by taking advantage of the powerful non-recurrent sequence to sequence learning capabilities inherent in the architecture of the Speech Transformer. Unlike recurrent-based approaches, the Speech Transformer model can efficiently process all input time series data in parallel, which enables faster training compared to sequential processing methods. This parallelization was incredibly advantageous because even using Google Colab resources to conduct the training, the computational constraints were real.. The speech corpus was prepared by collecting broadcast news audios from various Afaan Oromo media sources, totaling 8729 utterances from 100 speakers (50 males and 50 females) for a dataset of 18.04 hours. We experimented with four different models, varying the number of encoders, decoders, and feed-forward neural networks (FFNN). The best-performing model, with five encoders, three decoders, and 400 FFNN, achieved a word error rate (WER) of 40.2%. While this represents a promising result, we acknowledge that further improvements could be achieved by increasing the dataset size and using high-performance GPUs to enable the construction of larger and more complex models. In future work, we recommend conducting further studies with larger vocabularies and better computational resources to continue advancing the state-of-the-art in Afaan Oromo speech recognition. Additionally, we plan to explore the use of language models to further enhance the accuracy and robustness of the Afaan Oromo ASR system.

Keywords: Afaan Oromo, Speech Recognition, Automatic Speech Recognition, Speech Transformer.

CHAPTER ONE

INTRODUCTION

1.1. Background

Speech is a fundamental component of human communication, which encompasses the production of articulate vocal sounds to convey thoughts, ideas, and emotions. It involves the coordinated effort of various organs, including the lungs, vocal cords, tongue, lips, and mouth to produce a stream of sound waves that the brain shapes and modulates according to specific linguistic rules[1]. This complex process enables us to express ourselves, share information, engage in social interactions, contribute to society, build relationships, and shape our world.

Speech is the most common way humans communicate, and speech processing is a particularly interesting area of research within signal processing[1]. It involves studying language signals and the methods used to manipulate those signals. Since these signals are usually processed in a digital format, speech processing can be considered a specialized branch of digital signal processing that focuses specifically on speech signals[1].

Automatic Speech Recognition (ASR) refers to the technology that enables the conversion of spoken language into written text by computers[2]. ASR systems are integral to many modern applications, including virtual assistants, transcription services, and hands-free control of devices. The importance of ASR lies in its ability to enhance accessibility, streamline communication, and improve user interaction with technology. By enabling voice commands and dictation, ASR systems can significantly increase efficiency and convenience in both professional and personal contexts[3]. Essentially, it works by storing a human voice and training an automatic speech recognition system to recognize vocabulary and speech patterns in that voice[3].

Afaan Oromo, an Afroasiatic language spoken by the Oromo people in Ethiopia and Kenya, is one of the most widely spoken languages in Africa[3]. Despite its extensive use, the development of computational tools, particularly in the domain of Automatic Speech Recognition (ASR), has been limited. The existing ASR systems for Afaan Oromo have predominantly utilized traditional machine learning methods such as Hidden Markov Models (HMMs) and Artificial Neural Networks (ANNs)[4]. For instance, [5], [6], [7] employed HMMs to develop an ASR system for

Afaan Oromo, while[8] explored a hybrid HMM-ANN approach. These methods, while foundational, have limitations in handling the complexities of continuous speech with large vocabularies.

The advent of deep learning has revolutionized ASR, enabling significant improvements in accuracy and robustness. Deep learning models, particularly Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), have been utilized to capture temporal dependencies and spatial features in speech signals, respectively[9], [10], [11]. Afaan Oromo, [12] pioneered the application of deep learning by integrating RNNs and CNNs, demonstrating notable improvements over traditional methods. However, despite these advancements, a substantial performance gap remains in transcribing the diverse and complex speech patterns found in the Afaan Oromo language. This underscores the need to explore more advanced deep learning architectures that can further enhance the accuracy and robustness of Afaan Oromo ASR.

Recently, Transformer models have emerged as a powerful alternative for sequence modeling tasks, including ASR. Unlike RNNs, Transformers rely on self-attention mechanisms, which allow them to capture long-range dependencies more effectively and process input sequences in parallel, leading to faster training times and better performance on large datasets [13]. The Transformer model, introduced by Vaswani et al.[13], has revolutionized the field of natural language processing by employing a self-attention mechanism that allows for the efficient processing of entire input sequences in parallel [13]. This architecture eliminates the need for recurrence, which traditionally limits the performance and scalability of Recurrent Neural Network (RNN)--based models. The self-attention mechanism enables the Transformer to capture long-range dependencies and complex relationships within the data more effectively than previous models[13].

Building on this foundation, the Speech Transformer adapts the Transformer architecture specifically for automatic speech recognition (ASR) tasks[14]. By leveraging the self-attention mechanism, the Speech Transformer can model the intricate temporal patterns of speech without relying on sequential processing, thus enhancing both accuracy and computational efficiency[14]. This non-recurrent approach allows the Speech Transformer to handle entire sequences of speech data simultaneously, making it capable of capturing long-range dependencies and contextual information more effectively. Consequently, the Speech Transformer represents a significant

advancement in ASR, offering improved performance over traditional HMM and RNN-based models[14].

The focus on "Afaan Oromo Large Vocabulary Continuous Speech Recognition" is driven by the critical need to establish a foundational technology for accurate speech-to-text conversion in this under-resourced language. Large Vocabulary Continuous Speech Recognition (LVCSR) is essential because it addresses the challenges of transcribing spontaneous, natural speech that includes a wide range of vocabulary and complex linguistic structures inherent in Afaan Oromo. By developing a robust LVCSR system, we lay the groundwork for all subsequent language processing tasks, including Natural Language Understanding (NLU). Accurate transcription is a prerequisite for effective NLU[3]; without it, the interpretation and extraction of meaning from the text would be significantly impaired. Therefore, focusing on LVCSR is a strategic and necessary step to ensure the reliability and accuracy of any advanced language understanding systems built for Afaan Oromo.

Furthermore, addressing the unique challenges posed by the rich morphology and extensive vocabulary of Afaan Oromo through LVCSR provides immediate practical benefits. These include automated transcription services, educational tools, and enhanced accessibility for native speakers. Additionally, this foundational work will generate valuable data and insights, facilitating future developments in NLU and other natural language processing applications. Thus, while NLU is undoubtedly important, the focus on LVCSR is justified as it establishes the essential infrastructure needed to support comprehensive language technology solutions for Afaan Oromo.

1.2. Statement of the problem

Automatic speech recognition (ASR) is the technique of transcribing speech utterances to text, thereby facilitating easier and more preferable human-machine interaction. Rather than relying on typing, pressing buttons, or other mechanisms, natural speech offers the easiest and fastest way to communicate with machines [15]. Among all possibles interaction methods, humans learn speech first, making it the most intuitive and effective means of communication with technology.

ASR has numerous essential application areas that underscore its importance and relevance in today's technological landscape. From hands-free control of smart devices and virtual assistants to real-time speech-to-text transcription for accessibility and productivity, ASR technology enables seamless human-machine interaction. It is particularly valuable in healthcare settings, where clinicians can use voice commands to update electronic medical records, improving efficiency and reducing administrative burdens. In education, ASR supports language learning, lecture transcription, and assistive technology for students with disabilities. The transportation industry leverages ASR for voice-controlled infotainment systems and automated call centers [3], [16], [17]. These diverse application areas highlight the transformative potential of robust and accurate ASR, making it a critical area of research and development, especially for underserved languages.

Currently, speech recognition technology works well for well-resourced languages like English, Spanish, Mandarin, and Japanese. However, it still struggles with under-resourced languages such as those spoken in Ethiopia. More research is needed to solve this problem and improve speech recognition for these languages. One such Ethiopian language is Afaan Oromo [18].

Afaan Oromo, belonging to the Afro-Asiatic language family and reigning supreme among Cushitic tongues, is the fourth most spoken language in Africa, surpassed only by Arabic, Hausa, and Swahili [19]. However, the domain of speech technologies, encompassing both synthesis and recognition, remains nascent for Ethiopian languages, with Afaan Oromo enduring most of this technological lag [20]. Bridging this gap by developing an advanced Automatic Speech Recognition model for Afaan Oromo holds immense potential to revolutionize human-machine interaction, unlocking a multitude of benefits for its vast speaker base.

While there has been some research on Afaan Oromo ASR using statistical methods such as Hidden Markov Models , and HMM/ANN hybrids [5], [6] [with datasets not exceeding 4 hours in duration [7], only a few studies have explored the potential of deep learning architectures like CNN/RNN [7] and ANN [21] for this task. Although a researcher tried deep learning for this language using RNN/CNN hybrids, they struggled with accurately transcribing the consonant gemination and vowel sound long and short behavior of the Afaan Oromo language, and their data was limited to single sentence level [7]. This seq2seq model uses RNNs, which are sequential in nature, making model training time-consuming and inefficient. Additionally, the sequential architecture of RNNs may struggle to capture the complex linguistic features and long-range dependencies present in the Afaan Oromo language, further limiting the model's performance.

Recent research suggests that the Speech Transformer architecture offers several advantages that may lead to improved performance for ASR [14]. Transformers are well-suited for automatic speech recognition due to their self-attention mechanism, which enables the model to focus on relevant parts of the input sequence and capture long-term dependencies and contextual information [14]. Furthermore, Transformers excel in parallel computation during training and inference, leading to improved computational efficiency and scalability for large-scale speech recognition tasks [22]. The self-attention mechanism also allows Transformers to attend to different parts of the input sequence, effectively capturing local and global dependencies for variable-length utterances [14]. An additional advantage of the Speech Transformer model is that it generates one character at a time[14], which can be beneficial in capturing the consonant gemination and vowel letter long and short behavior of the language, which traditional RNN-based ASR could not handle for Afaan Oromo [12].

This issue is of particular significance in the context of Afaan Oromo, as the language is highly morphologically complex, where the omission or misrepresentation of even a single character in a word can lead to a change in meaning or render the word meaningless. Accurate transcription is, therefore, an essential requirement for the development of effective and reliable Afaan Oromo ASR systems. For example, in the sentence ‘Dabalaan abbaa isaa dhaale’ meaning "Dabala inherited his father", if in the word 'dhaale', one ‘a’ letter is omitted and it is written as ‘Dabalaan abbaa isaa dhale’ meaning "Dabalaa gave birth to his father", it becomes illogical. Accurate transcription is

critical in this complex language, which the recurrent seq2seq model has been unable to overcome[12].

To address these challenges, this research aims to introduce an Automatic Speech Recognition (ASR) system for the Afaan Oromo language using a Speech Transformer model. By leveraging Afaan Oromo broadcast speech data, a truly low-resource language setting, this study seeks to demonstrate the effectiveness of Transformer-based

1.3. Research question

After conducting this study, the researchers should answer the following research questions.

1. What are the optimal values of hyperparameters that give the best performance?
2. How effective is a speech transformer model in transcribing Afaan Oromo speech?

1.4. Objective of the Study

1.4.1. General Objective

The main objective of this research is to develop and evaluate a large vocabulary Afaan Oromo speech recognition model using the speech transformer architecture

1.4.2. Specific Objectives

To achieve the general objective of the study, this research attempted the following specific objectives.

- To identify the challenges in developing a ASR model for Afaan Oromo using a speech transformer.
- To experiment with different hyperparameters of the speech Transformer model.
- To evaluate effectiveness of a speech transformer model in transcribing Afaan Oromo speech.

1.5. Scope and Limitation of the Study

The main aim of this study is to to develop and evaluate Afaan Oromo ASR and enhance the accuracy and reducing word error rate (WER) by increasing datasets and using state-of-the-art automatic speech recognition architecture which is a Speech Transformer.

However, in this research, problems like age, health, and emotional state of the speaker were not addressed because of time and budget. However, the researchers have tried to address the performance problem of speech recognizers of previous works and because previous works used very small datasets even the one developed by deep learning [7], we have tried to prepare good-quality data that can be used for this and further research.

The experiment has been conducted for only recognizing speech and converting to text, without identifying the age and emotional state of the speaker. Fortunately, the dataset had been balanced between females and males to make the system representative and unbiased.

1.6. Significance of the Study

This research is significant as it aims to enhance automatic speech recognition (ASR) technology for the Afaan Oromo language, which is spoken by millions in Africa, Ethiopia, and surrounding regions. Despite its widespread use, Afaan Oromo has been underrepresented in language technology research, leading to limited digital resources and tools. By employing the Speech Transformer model, this study seeks to address this gap, offering a more effective and scalable solution for ASR in this language.

- Firstly, developing a high-quality ASR system for Afaan Oromo greatly benefits native speakers by improving accessibility to digital services and resources. This can facilitate more natural interactions with technology, enabling better access to educational materials, online services, and assistive technologies. For instance, voice-operated tools and applications can become more user-friendly and inclusive, catering to the needs of Afaan Oromo speakers.
- Secondly, this research contributes to the preservation and promotion of the Afaan Oromo language. Integrating the language into advanced technological frameworks not only aids in its documentation and standardization but also promotes its use in various digital contexts. This can help sustain the language in the face of globalization and technological change, ensuring it remains a vibrant part of the cultural heritage.
- Moreover, the study demonstrates the potential of the Speech Transformer model in addressing the challenges associated with low-resource languages. The model's non-recurrent, self-attention mechanism allows it to process speech data more efficiently and accurately, even with limited training data. This can serve as a valuable case study for other researchers working on ASR for low-resource languages, highlighting the feasibility and benefits of using advanced machine-learning techniques.
- Finally, the outcomes of this research are expected to stimulate further academic and commercial interest in developing language technologies for Afaan Oromo. By showcasing the effectiveness of state-of-the-art models in this context, the study can attract more funding and collaborative efforts, fostering innovation and development in the field of speech and language processing technologies.

1.7. Organization of the Thesis

This thesis comprises five chapters. Chapter two deals with speech recognition, showcasing the proposed methodology and machine learning techniques while reviewing prior research pertinent to the study's objectives. It includes analyses of journals and gathered materials essential for this study, culminating in crucial conclusions and recommendations. Chapter three details the dataset preparation steps, machine learning algorithms, proposed architecture, and other components utilized in this research. Chapter four discusses the research findings and conducts a comprehensive analysis based on the methods introduced in Chapter Three. Finally, Chapter Five summarizes the outcomes, concludes the study, furnishes recommendations for researchers, and outlines potential future research endeavors for this study.

CHAPTER TWO

LITERATURE REVIEW

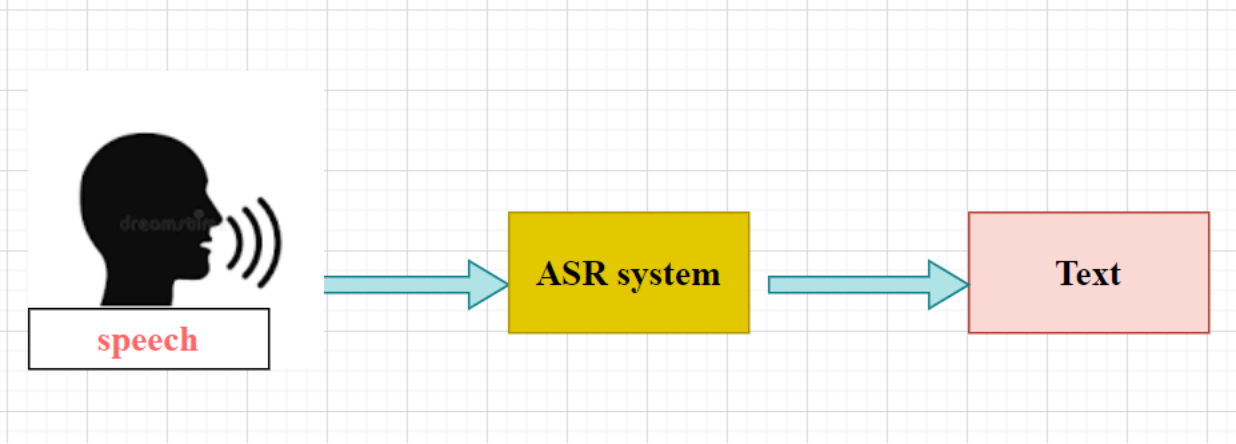
This chapter provides an overview of the key techniques required for constructing a complete speech recognition system. It begins by discussing various options for representing input features, followed by an exploration of essential machine-learning tools. Lastly, it delves into the Speech Transformer architecture, which is a sequence-to-sequence labeling method.

2.1. Speech Recognition Overview

Speech recognition, also known as Automatic Speech Recognition (ASR), is the process of converting spoken language into text [23]. This technology has made significant advancements in recent years, with applications ranging from voice assistants like Siri and Alexa to dictation software and voice-controlled interfaces.

The objective of speech recognition is to enable machines to understand human speech and respond accordingly. Automatic Speech Recognition (ASR) plays a crucial role in the field of Artificial Intelligence, involving the conversion of speech data into text, as depicted in Figure 1.

Figure 2. 1: General Architecture of an ASR Model



2.2. Speech File Formats

In the context of automatic speech recognition (ASR) dataset collection, selecting the appropriate speech file format is indeed a crucial step. Here is a comprehensive overview of some commonly used speech file formats that you may consider including in your research:

1. WAV (Waveform Audio File Format): WAV is a widely used uncompressed audio file format. It stores audio data in a raw format, making it compatible with various platforms and applications. WAV files are known for their high audio quality and are commonly used in ASR research and applications[24].

2. FLAC (Free Lossless Audio Codec): FLAC is a lossless audio compression format that allows for high-quality audio storage while reducing file size. It is suitable for ASR datasets as it retains the original audio quality while achieving efficient compression[24].

3. MP3 (MPEG Audio Layer-3): MP3 is a popular compressed audio file format that utilizes perceptual audio coding to achieve high compression ratios. However, due to the lossy compression, some audio quality may be sacrificed. MP3 files are widely supported and can be suitable for ASR dataset collection depending on the specific requirements of your research[24].

4. OGG (Ogg Vorbis): OGG is an open and free audio file format that offers high audio quality and efficient compression. It is a suitable choice for ASR datasets where file size needs to be minimized while maintaining good audio quality[24].

5. M4A (MPEG-4 Audio): M4A is a container format that can store audio encoded by various codecs. It is commonly used for storing audio in a compressed format. M4A files are widely supported and can be suitable for ASR dataset collection, especially when dealing with audio recorded on mobile devices[24].

6. AMR (Adaptive Multi-Rate): AMR is a speech coding format specifically designed for speech compression. It is commonly used in mobile communications and is suitable for ASR datasets that focus primarily on speech recognition tasks[24].

7. MP4 (MPEG-4 Part 14): MP4 is a multimedia container format that can store both audio and video data. It is widely supported and can be used for ASR datasets that include audio with accompanying visual information, such as lip movements or facial expressions[24].

When selecting the appropriate speech file format for research, factors such as audio quality, compression requirements, compatibility with ASR systems, and the specific objectives of the study will be considered. It is also important to ensure that the selected file format is supported by the tools and libraries planned to be used for dataset preprocessing and ASR model training.

2.3. Types of Speech Recognition

Speech recognition can be categorized into different types based on various factors. Here are some common types of speech recognition:

1. Speaker-dependent vs. Speaker-independent: In speaker-dependent speech recognition, the system is trained to recognize the speech of a specific user. It requires the user to provide a training sample to adapt the system to their voice. On the other hand, speaker-independent speech recognition systems are designed to recognize speech from any user without prior training[25].

2. Isolated Word Recognition: This type of speech recognition focuses on recognizing individual words or short phrases spoken by the user. It is commonly used in applications where discrete commands or keywords need to be detected, such as voice dialing on smartphones.

3. Continuous Speech Recognition: Continuous speech recognition involves recognizing natural, continuous speech without any pauses between words. It aims to transcribe entire sentences or paragraphs spoken by the user. Applications of continuous speech recognition include transcription services, voice assistants, and dictation software.

4. Command-and-Control Speech Recognition: This type of speech recognition is optimized for recognizing specific commands or instructions given by the user. It is commonly used in applications like voice-controlled home automation systems, voice-activated virtual assistants, and voice commands in vehicles.

5. Large Vocabulary Speech Recognition: Large vocabulary speech recognition systems are designed to handle a wide range of words and phrases. They are used in applications like transcription services, voice search, and dictation software, where the recognition of a large vocabulary is crucial.

6. Natural Language Understanding: Natural Language Understanding (NLU) goes beyond simple speech recognition by attempting to understand the meaning and intent behind the spoken words. It involves parsing the speech to extract information and perform tasks such as answering questions or providing relevant responses. NLU is used in virtual assistants and customer support systems[25].

7. Speaker Diarization: Speaker diarization refers to the process of separating multiple speakers in an audio stream. It is used to identify and differentiate between speakers in applications like meeting transcription, call center analytics, and forensic analysis[25].

2.4. Deep Learning for Speech Recognition

In this section, we will introduce deep learning in speech recognition, a landmark of some model structures and discuss their advantages and disadvantages.

Deep learning is a fascinating branch of research within the field of machine learning. It focuses on the development of artificial neural networks that can learn and grow, much like our brains [26]. This powerful technique allows us to train deep structural models and extract valuable insights from complex data.

There exist various types of deep learning architectures, including recurrent neural networks (RNNs), long short-term memory (LSTM) or gated recurrent unit (GRU), convolutional neural networks (CNNs), deep belief networks (DBN), and deep stacking networks (DSNs) [27].

2.4.1. Convolutional Neural Network

Convolutional neural networks (CNNs) are a type of feedforward neural network that can automatically extract features from data using convolutional structures. Unlike traditional feature extraction methods, CNNs eliminate the need for manual feature extraction [28]. The architecture of CNNs is inspired by visual perception, where biological neurons correspond to artificial neurons, and CNN kernels represent receptors that detect different features [29]. Activation functions simulate the behavior of neural electric signals, allowing only signals surpassing a certain threshold to be transmitted to the next neuron. Loss functions and optimizers are devised to facilitate the learning process of the entire CNN system[30].

Compared to fully connected (FC) networks, CNNs offer several advantages. Firstly, CNNs employ local connections, meaning each neuron is only connected to a small number of neurons from the previous layer. This reduces the number of parameters and accelerates convergence. Secondly, weight sharing is utilized, allowing a group of connections to share the same weights, further reducing the number of parameters. Lastly, CNNs employ down-sampling techniques such as pooling layers that leverage the principle of local correlation in images to reduce data size while retaining valuable information. Downsampling also helps eliminate insignificant features, thereby

reducing the number of parameters. These three advantageous characteristics have established CNNs as one of the most prominent algorithms in the field of deep learning [30].

When constructing a CNN model, four essential components are typically required. The first component is convolution, which plays a crucial role in extracting features from the input data. The output of convolution is referred to as feature maps. However, when applying convolution with a specific kernel size, information at the borders may be lost. To address this, padding is introduced, which involves enlarging the input by adding zero values, indirectly adjusting the size. Additionally, the stride is employed to control the density of the convolving process. A larger stride corresponds to a lower density [30].

Following convolution, the feature maps contain a large number of features, which can potentially lead to overfitting issues [31]. To mitigate this problem, pooling (also known as down sampling) is introduced. Pooling methods, such as max pooling and average pooling, help reduce redundancy in the feature maps [30].

Convolutional neural networks (CNNs) can be categorized into three types: 1D, 2D, and 3D convolutions, representing one-dimensional CNN, two-dimensional CNN, and three-dimensional CNN, respectively. One-dimensional CNNs, or Conv1D CNNs, involve kernels sliding along one dimension, making them well-suited for analyzing time series data like speech or sound [27]. Two-dimensional CNNs, or Conv2D CNNs, are typically used for image data, as the kernels slide along two dimensions in the data. On the other hand, three-dimensional CNNs, or Conv3D CNNs, operate by sliding kernels in three dimensions. Conv3D CNNs are commonly employed when working with 3D image data, such as Magnetic Resonance Imaging (MRI) data [30].

2.4.2. Recurrent neural networks (RNNs)

Recurrent Neural Networks (RNNs) and their variants, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), have emerged as powerful tools for Automatic Speech Recognition (ASR) [9].

Automatic Speech Recognition (ASR) is a rapidly evolving field that has greatly benefited from the adoption of Recurrent Neural Networks (RNNs).

Definition and Working Principle of Recurrent Neural Networks

RNNs are a class of neural networks specifically designed to handle sequential data [9]. Unlike feedforward neural networks, RNNs have feedback connections that allow information to flow not only from the input layer to the output layer but also in a recurrent manner, preserving memory of past inputs [9]. This recurrent nature enables RNNs to capture temporal dependencies and process sequential data.

RNNs have found extensive application in ASR tasks, revolutionizing the field. They have been used for various speech recognition components, including acoustic modeling, language modeling, and joint acoustic-linguistic modeling. RNNs excel at modeling the temporal nature of speech and capturing dependencies between phonemes, words, and linguistic context, thereby improving ASR accuracy [11].

Types of Recurrent Neural Network Architectures

Long Short-Term Memory (LSTM)

LSTM is a variant of RNNs that overcomes the vanishing gradient problem by incorporating memory cells and gating mechanisms. The LSTM architecture consists of input, forget, and output gates, enabling the network to selectively retain and forget information. This capability allows LSTM to capture long-term dependencies in sequential data, making it highly effective in ASR[9].

Gated Recurrent Unit (GRU)

GRU is another variant of RNNs that addresses the vanishing gradient problem while being computationally efficient. GRU simplifies the LSTM architecture by combining the forget and input gates into a single update gate. This streamlined design allows GRU to capture temporal dependencies in ASR tasks, making it a popular choice [26].

LSTM and GRU have significantly improved ASR performance by effectively capturing long-term dependencies, modeling temporal context, and mitigating the vanishing gradient problem [9]. They have been extensively employed in various ASR tasks, including speech recognition, keyword spotting, speech-to-text conversion, and speaker identification[32].

Deep Belief Networks (DBNs)

Deep Belief Networks (DBNs) are a class of generative models that have gained significant attention in the field of machine learning. DBNs are composed of multiple layers of restricted Boltzmann machines (RBMs) or their variants, forming a deep, hierarchical architecture [33].

These models have demonstrated remarkable performance in various domains, including image recognition, natural language processing, and speech recognition.

Architecture and Working Principle

DBNs consist of multiple layers of interconnected RBMs. RBMs are unsupervised learning models that consist of a visible layer and a hidden layer. The visible layer represents the input data, while the hidden layer captures latent features and representations. In a DBN, the hidden layer of one RBM serves as the visible layer for the subsequent RBM, creating a deep, hierarchical structure[33].

Training a DBN involves a two-step process: pre-training and fine-tuning. Pre-training is performed layer by layer, where each RBM is trained to reconstruct its input data. This unsupervised pre-training enables the RBMs to capture useful features and form a distributed representation of the input data. After pre-training, the DBN is fine-tuned using supervised learning techniques, such as backpropagation, to optimize the overall model for a specific task[33].

2.5. Feature Extraction

Feature extraction is a sophisticated process that transforms raw data into numerical features, enhancing their processability without sacrificing the underlying information. By performing this transformative feat, we pave the way for superior results, surpassing the limitations of applying machine learning algorithms directly to the raw data [34]. Feature extraction identifies the most discriminating characteristics in signals, which machine learning or a deep learning algorithm can more easily consume [34]. By extracting the most significant characteristics and presenting them in a numerical form, we equip machine learning and deep learning algorithms with the tools they need to thrive and succeed.

In speech processing the process of extracting important information from a speech signal and reducing noise and unwanted information is called feature extraction. Feature extraction involves the process of converting the speech signal into digital form[34]. In the realm of digital signal processing, speech processing stands as a prominent field. It is one of an intensive field of research. The major criterion for a good speech processing system is the selection of a feature extraction technique, which plays a major role in achieving optimal accuracy [34]. In this section of this

work, the most commonly used techniques for feature extraction such as Filter bank, Linear Predictive Coefficient (LPC), Mel Frequency Cepstral Coefficient (MFCC), Perceptual Linear Prediction (PLP), Relative Spectral Perceptual Linear Prediction (RASTA-PLP) and Wavelet Transform (WT) are discussed.

Filter-bank features are used in speech analysis and recognition[35]. They involve the use of collections of filters that are spread out across audible frequencies. The output of these filter banks is then used to measure the energy within specific parts of the signal spectrum.

To provide a clearer understanding, here are some important features of filter-bank analysis

1. Purpose: Filter-bank analysis is commonly used in speech analysis and recognition systems. It helps extract relevant information from the speech signal by dividing it into frequency bands and measuring the energy within each band [35].

2. Number of Filter Banks: The number of filter banks used depends on the required resolution [36]. In many applications, such as speech recognition, 32 filter banks are commonly used. However, the specific number can vary depending on the intended application[35].

3. Energy Measurement: The energy within the spectrum covered by each filter is measured. This measurement captures the strength or intensity of the signal within each frequency band [36].

4. Cochlear Resemblance: The motivation behind using filter banks in speech recognition is that the human cochlea, part of the inner ear, can be thought of as a natural filter bank. The cochlea helps humans perceive and analyze different frequencies present in speech and other sounds [37].

5. Non-Linear Perception of Frequency: Humans do not linearly perceive frequency. Instead, our perception is skewed towards certain frequency ranges. The Mel-scale is a perceptual scale that represents the way humans hear pitch and frequency. It is often used to design the filter banks in speech analysis systems to better align with human auditory perception [36].

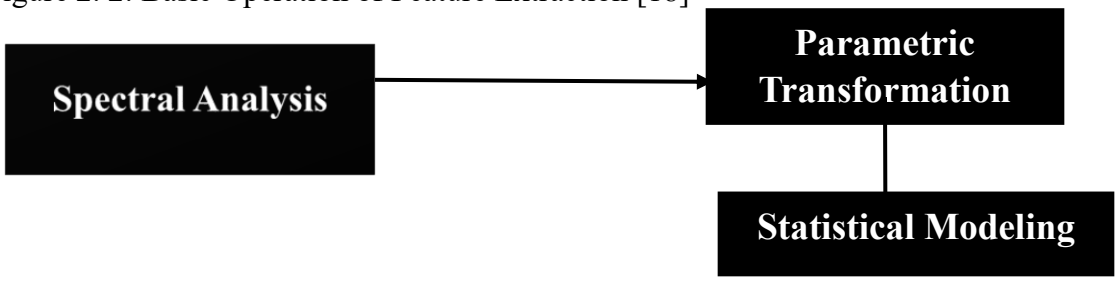
$$B(f) = 1125ln(1 + f/700). \quad (2.1)$$

The formula above represents the conversion of frequency f into the Mel-scale in speech recognition. The Mel-scale is a perceptual scale of pitches that approximates the human ear's response to different frequencies. The formula is used to map the linear frequency scale to the Mel scale. By employing filter-bank analysis and leveraging the similarities with the human auditory

system, speech analysis systems can effectively extract relevant features from the speech signal, which can then be used for various applications, including speech recognition, speaker identification, and emotion recognition [38].

MFCCs are widely used in speech feature extraction and aim to replicate the functioning of the human auditory system. One significant advantage of MFCC is that it effectively captures the essential characteristics of speech sounds, known as phones. Additionally, MFCC has the benefit of being computationally efficient, making it a practical choice for various applications[39]. However, in the presence of background noise, MFCC may not provide accurate results and tends to perform better in clean speech conditions compared to speech contaminated by noise [39].

Figure 2. 2: Basic Operation of Feature Extraction [18]



In Linear Predictive Coding (LPC) analysis, a speech sample is approximately linearly assembled from past speech samples. LPC is a frame-by-frame analysis of the speech signal [39]. Adjacent frames in the input speech signal are separated and grouped into blocks of samples. To minimize signal discontinuities, each frame is smoothed [40]. This is followed by calculating the autocorrelation of each frame of the smoothed signal, and then it converts each frame of autocorrelations into an LPC parameter set using Durbin's method [40].

The Discrete Wavelet Transform (DWT) is a mathematical technique that breaks down a signal into a set of elementary functions known as wavelets [39]. One significant advantage of the DWT is its ability to compute results quickly. However, it is worth noting that the DWT does not possess robust noise resistance capabilities. In the presence of noise, the DWT may struggle to effectively preserve the integrity of the original signal, potentially leading to a degradation in the quality of the transformed data [41].

Hybrid Feature Extraction Techniques

Research has revealed that combining multiple methods to extract relevant features yields improved performance [39]. These hybrid methods warrant further exploration. In this section of this work the hybrid features extraction techniques such as Discrete Wavelet Packet Decomposition (DWPD), Wavelet-Based Mel-Frequency Cepstral Coefficients (WPCC), Revised Perceptual Linear Prediction (RPLP), Bark frequency cepstral coefficients (BFCC) are discussed. To improve discourse and overcome the limitations of DWT and WPD, innovative hybrid approaches were introduced. This novel hybrid technique, christened Discrete Wavelet Packet Decomposition (DWPD), seamlessly merges the strengths of both DWT and WPD. DWPD operates in three discrete stages. Initially, the input signal is bifurcated into two distinct bands: a high-frequency band and a low-frequency band. Subsequently, WPD is applied to the high-frequency band, while DWT is employed on the low-frequency band. Finally, the extracted features from both techniques are coalesced, forming a comprehensive feature vector set [42].

This hybrid approach, DWPD, boasts several key advantages. Notably, the high-frequency band undergoes further decomposition into finer sub-bands. This enhanced granularity improves feature representation, leading to superior computational efficiency and ultimately yield higher recognition rates[39], [42]. Phase Autocorrelation Bark Wavelet Transform (PACWT) leverages the strengths of both phase autocorrelation (PAC) and bark wavelet transform. This hybrid approach enhances robustness by utilizing an alternative autocorrelation measure [43].

First, the speech signal undergoes a pre-emphasis step. Subsequently, a Hamming window is applied to a specific frame within the pre-emphasized signal. The calculation of correlation coefficients then yields an autocorrelation sequence during the Phase Autocorrelation stage [34]. Following this, the bark wavelet transform is simply applied to the signal that has passed through the Mel-filter bank. As a result, PACWT feature coefficients are generated. Additionally, the first and second derivatives of the time sequence for each base feature are computed. Finally, the complete set of PACWT feature coefficients is obtained by concatenating the derivatives to the base feature set[39].

Wavelet-Based Mel-Frequency Cepstral Coefficients (WPCC) combines the wavelet transform and the MFCC method. First, the wavelet transforms split the speech signal into two distinct frequency channels. The high-frequency channel retains all the details, while the low-frequency

channel only approximates the signal. Next, the MFCCs are computed for both the approximation and detail channels. This approach aims to capture the unique characteristics of individual speakers and simplifies the calculation of the coefficients[39].

2.6. Speech Transformer

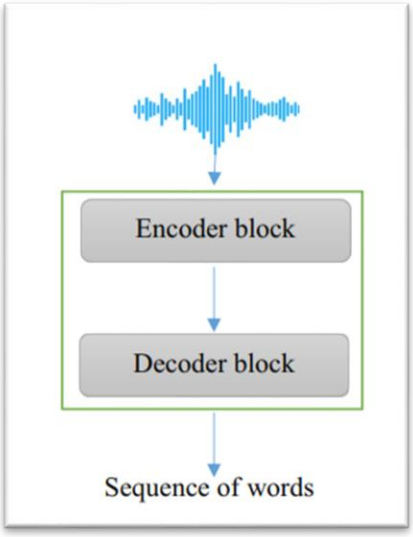
2.6.1. Introduction

Since its introduction in 2017, Transformer has been widely recognized as a powerful encoder-decoder model to solve any language modeling task. BERT, RoBERTa, and XLM-RoBERT are a few examples of state-of-the-art models in language processing that use a stack of Transformer encoders as the backbone of their architecture. ChatGPT and the GPT family also use the decoder part of the Transformer to generate texts. It's safe to say that almost any state-of-the-art model in natural language processing incorporates a Transformer in its architecture[44]

While recurrent sequence-to-sequence models with encoder-decoder architecture have shown significant advancements in speech recognition, they have a drawback in terms of slow training speed. This is primarily due to the internal recurrence present in these models, which limits the ability to parallelize the training process[14].

Transformer also has an excellent potential to be applied to automatic speech recognition. In 2018, a group of China's scientists introduced a Transformer-based model called speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition that can be used for speech recognition tasks. Its performance is very competitive compared to conventional CNNs and RNNs on speech recognition benchmarks [14].

Figure 2. 3: General scheme of the model[43]



2.6.2. How Speech Transformer Works

The Transformer model was initially developed for machine translation, replacing recurrent neural networks (RNNs) in natural language processing (NLP) applications. Unlike RNNs, the Transformer model does not rely on recurrence. Instead, it utilizes an internal attention mechanism known as self-attention. This mechanism helps identify the importance of other sequences within the same input, allowing the model to generate meaningful features for each statement. These features are obtained through linear transformations of the significant sequence features [13].

The Speech-Transformer follows the same basic encoder-decoder architecture as previous sequence-to-sequence models. In this architecture, the encoder takes a sequence of speech features$(x1, \dots, xT)$ and transforms it into a hidden representation $h = (h1, \dots, hL)$. Using this hidden representation, the decoder generates an output sequence$(y1, \dots, ys)$ character by character. At each step of the decoding process, the decoder considers the previously generated characters as additional inputs to determine the next character to emit[14].

The Speech-Transformer, in contrast to recurrent seq2seq models, is a non-recurrent sequence-to-sequence model that differs in two main aspects. Firstly, both the encoder and decoder components are constructed using multi-head attention and position-wise feed-forward networks, instead of relying on recurrent neural networks (RNNs). Secondly, the encoder's outputs, represented as $\boldsymbol{h}$, are

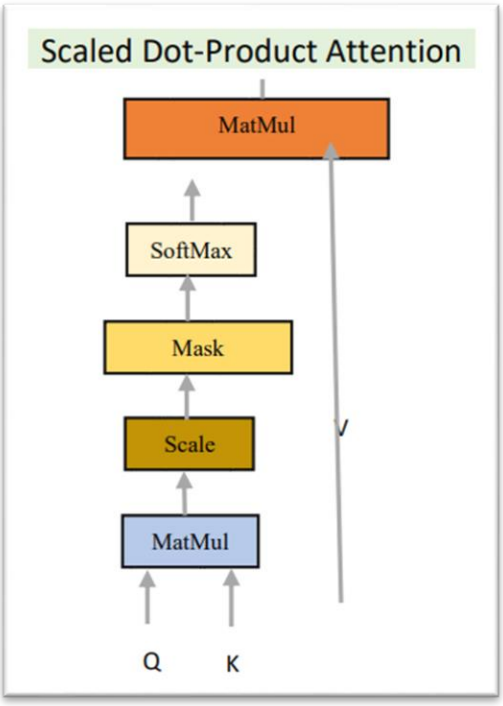
attended to by each decoder block independently, eliminating the need for the intermediate attention step present in recurrent seq2seq models.

In the following sections, we will provide a more detailed description of the Speech-Transformer.

2.6.3. Core Module of the Speech-Transformer

2.6.4. Scaled Dot-Product Attention

Figure 2. 4: Scaled Dot-Product Attention



In the given context, MatMul refers to a function that calculates the matrix product of two arrays. SoftMax is an activation function used in the process. The term "mask" is employed to ensure that predictions for a specific position (j) are influenced solely by the known outputs at preceding positions (less than j). The concept of "Scale" involves scaling down the dot product by $1/\sqrt{dk}$, which serves the purpose of preventing the SoftMax function from encountering regions with extremely small gradients.

Self-attention is a mechanism that allows for the computation of representations for input sequences by establishing relationships between different positions within the sequences. It takes three inputs: queries, keys, and values. The output of a query is determined by taking a weighted sum of the

values, where each weight is determined by a specific function that relates the query to the corresponding key[14].

Speech Transformer used Scaled Dot-Product Attention, which is a highly effective self-attention mechanism as demonstrated in reference [13].

The self-attention mechanism involves three inputs: queries Q, keys K, and values V. These inputs are matrices, where the dimensions of the matrices are denoted by t^* and d^*. Specifically, Q is matrix size of, $Q \in R^{t_q x d_q}$, K is a matrix of, $K \in R^{tkxdk}$, and V is a matrix of, $V \in R^{tvxdv}$. Here, $t_{q,} t_{k,}$ and t_v represent the number of elements in the different inputs, while dq and dk represent their respective element dimensions. In most cases, the number of keys tk is equal to the number of values tv, and the dimensions of the queries dq and keys dk are the same. The outputs of the self-attention operation are computed based on these inputs.

$$Attention(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{dk}})$$

To avoid the SoftMax function from encountering regions with extremely small gradients, the scalar $1/\sqrt{dk}$, is incorporated which helps to scale the dot product during the attention calculation.

2.6.5. Multi-Head Attention

In the Speech-Transformer model, multi-head attention is a crucial component that combines various attending representations. It performs h instances of Scaled Dot-Product Attention, where h represents the number of attention heads. Before each attention calculation, linear projections are applied separately to the queries, keys, and values, transforming them into more distinctive representations. Each Scaled Dot-Product Attention is then computed independently, and the resulting outputs are concatenated. These concatenated outputs are subsequently passed through another linear projection to generate the final outputs, which have a dimensionality of d_{model} [14].

$$MultiHead(Q, K, V) = Concat(head_1, \dots head_h)\, W^O$$

where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$

In the Speech-Transformer, the dimensions of the query (Q), key (K), and value (V) matrices are all equal to d_{model} This is achieved by using projection matrices $WQi, WKi, and\ WVi,$ each with dimensions of $d_{model} \times dq$, $d_{model} \times dk$, and $d_{model} \times dv$, respectively. Additionally, there is a projection matrix WO with dimensions hdv × dmodel. Throughout the paper, the values of dq, dk, dv, and $d_{model}\ /h$ remain consistent [14].

2.6.6. Position-Wise Feed-Forward Network

The position-wise feed-forward network is another important component of the Speech-Transformer model. It consists of two linear transformations with a ReLU activation function applied in between them. The input and output of this network have a dimensionality of d_{model}, while the inner layer has a dimensionality of d_{ff}.

Specifically, the feed-forward network operation $FFn(x)$ is defined as:

$FFn(x) = max(0, xW1 + b_1)w_2 + b_2$ where the weight matrices $W1\ and\ W2$ have dimensions of $d_{model} \times d_{ff}$ and $d_{ff} \times d_{model}$, respectively. Additionally, there are bias vectors b1 and b2 with dimensions $d_{ff}$and d_{model}, respectively. The linear transformations within the feed-forward network remain the same across different positions.

2.7. Related Works

In this section, we review research done in the area of speech recognition in both international and local languages spoken in Ethiopia. We divide works into two groups based on the model they used, statistical methods like HMM and deep learning-based.

The research in speech recognition traces back to 1952 when Davis, Biddulph, and Balashek built a system for isolated word recognition system for a single speaker in Bell Laboratories by designing the ASR system for the ten digits (zero to nine) [25].

Afaan Oromo Automatic speech recognition using statistical and hybrid methods

Senbetu [45] developed a speaker-dependent speech recognition system for the Kambaatissa language using Hidden Markov Models (HMMs). The system achieved high accuracy (average WER of 0.67 and average WAR of 99.33) on a dataset of 4820 words recorded by four native

speakers. The findings suggest that HMMs are effective for speaker-dependent speech recognition in small, under-resourced languages like Kambaatissa.

Kebede [7], investigated the development of an Afaan Oromo speech-based interface for controlling Microsoft Word. Using isolated word recognition and a tri-phone HMM model, He achieved an 88.99% recognition accuracy for 54 selected commands. While resource limitations restricted the study to a small set of commands, the findings suggest that a fully functional Afaan Oromo speech interface for Word could be developed with more extensive language resources and object-as-a-service components.

Mohammed [4] explores the development of an Afaan Oromo speech recognition system using Artificial Neural Networks. The authors collected phonemes, grouped them into sentences, and employed various algorithms, including preprocessing, feature extraction, and classification, implemented through ANN toolboxes and MATLAB scripts. The system achieved an accuracy of 91.1%, demonstrating the potential of ANNs for Afaan Oromo speech recognition. However, 8.9% of samples were misclassified, indicating room for further improvement.

In an attempt to develop Afaan Oromo speech recognition, Teferi [10] explored a hybrid HMM/ANN model and compared its effectiveness to traditional HMM models. Using CSLU tools and a specially curated dataset of 100 words and 29 Afaan Oromo phonemes, the system achieved a remarkable 98.11% accuracy in recognizing limited vocabulary sentences. This result suggests promise for further development and potential application of Afaan Oromo speech recognition technology.

Yadeta [5] investigated building a large vocabulary, speaker-independent, continuous speech recognition system for Afaan Oromo using broadcast news data. He employed statistical methods; Hidden Markov Models (HMMs) with tools like HTK, Audacity, and SRILM. The datasets, collected from various sources like ORTO, VOA Afaan Oromo, and FBC, comprised 2953 utterances (6 hours) from 57 speakers. A text dataset for language modeling was built from the Bariisaa Afaan Oromo newspaper, and a bigram language model was developed using SRILM. Training used 2653 utterances, while 300 utterances from 12 new speakers (40 minutes) were used for testing. Context-independent and context-dependent acoustic models were created. The best

performance achieved was 91.46% WER for context-independent and 89.84% WER for context-dependent models. The researchers concluded that increasing the Gaussian number to 12 and tuning specific parameters could improve the system's accuracy.

Duressa [8]explored Afaan Oromo speech recognition using Hidden Markov Models, building a prototype on one hour of read speech. Tests on speaker-dependent and independent data showed bigram language models achieving the highest word accuracy (93% and 43.6%, respectively). Despite dataset size limitations, the study concluded that diverse acoustic model training parameters, beyond increasing training data, significantly impact performance. This suggests investigating different Gaussian mixtures, tied states, transition topologies, and model types for Afaan Oromo speech recognition.

In a pioneering effort, Kasahun [6] explored Speaker-Independent Continuous Afaan Oromo speech recognition using HMM models using the Sphinx Toolkit. Training on 2100 utterances of varied word lengths and simple sentences are spoken by 30 age-diverse individuals, His model achieved promising results: 68.51% word-level accuracy and 89.46% tri-phone accuracy, with sentence accuracy, reaching 28% and 42%, respectively. Though these are early steps, they pave the way for further advancements in Afaan Oromo speech recognition technology.

Afaan Oromo Automatic Speech Recognition Using Deep Learning Approach

Unfortunately, on the side of Afaan Oromo ASR technology development and fortunate for the researchers to put his contribute of Afaan Oromo ASR technology development, the only research for Afaan Oromo ASR using deep learning technique we found after extensive literature is the one done by Sifan[12].

Sifan [12] aimed to develop a continuous speech recognition system for the Afaan Oromo language, utilizing deep learning for the first time in this context. She employed a CNN/RNN with a connectionist temporal classification CTC model to tackle sequence problems trained on a dataset of 8000 utterances collected from various broadcasting sources. Through experimentation with different configurations, she achieved a word error rate of 69% and a loss of 16.3, demonstrating the potential of deep learning for Afaan Oromo ASR. However, limitations in data size and computational resources highlight the need for further research with larger datasets and higher-performing GPUs to enhance accuracy.

2.8. Research Gap

The existing research on Afaan Oromo automatic speech recognition (ASR) has predominantly focused on statistical methods such as HMM[5], [6] and HMM and ANN hybrids[4], [21] and recurrence deep learning approaches[12], leaving a significant research gap regarding the exploration and utilization of non recurrence a Speech Transformer models. While statistical methods and hybrid models have been employed, the potential of the non recurrence Transformer-based architectures remains largely unexplored. Transformers offer unique properties, such as self-attention mechanisms and parallel computation, which make them well-suited for ASR tasks[14]. However, their application to Afaan Oromo ASR has been limited, hindering the advancement of speech recognition technology for this language.

One crucial aspect where Transformers hold promise is their ability to capture long-term dependencies and contextual information, which is vital for accurate speech recognition[14]. Unlike recurrent neural networks (RNNs), Transformers incorporate self-attention mechanisms without recurrence, enabling them to focus on relevant parts of the input sequence during decoding[14]. This capability allows Transformers to effectively handle the complex linguistic patterns and variations present in Afaan Oromo speech. However, the research gap lies in investigating how Transformers can be effectively adapted to the specific challenges posed by Afaan Oromo, such as variable-length input utterances and diverse pronunciation variations.

Furthermore, Transformers excel in parallel processing of input sequences during both training and inference, leading to improved computational efficiency and scalability [13]. This inherent parallelism of the Transformer architecture, where input tokens are processed simultaneously rather than sequentially, offers significant advantages for large-scale ASR tasks, such as processing extensive broadcast data in Afaan Oromo. The ability to leverage this parallel processing capability can potentially lead to more efficient and scalable speech recognition models, as compared to traditional sequential approaches. However, the extent to which Transformers can effectively harness their parallel processing advantages in the context of Afaan Oromo ASR remains unexplored. Closing this research gap would not only enhance the performance of Afaan Oromo

ASR systems but also facilitate the development of scalable and efficient speech recognition models for other low-resource languages.

The linguistic characteristics of Afaan Oromo, such as the occurrence of consonant gemination and long vowel letters, pose challenges for conventional ASR methods like Connectionist Temporal Classification (CTC)[12]. CTC-based models typically assume that no two similar vowel or consonant letters can occur consecutively, and treat such patterns as errors to be ignored. However, this limitation is critical for accurately transcribing Afaan Oromo, where consonant gemination and long vowel letters are important linguistic features.

Addressing these research gaps by exploring and harnessing the potential of Speech Transformer models for Afaan Oromo ASR would significantly contribute to advancing the field. It would bridge the gap between traditional approaches and cutting-edge deep learning techniques, paving the way for more accurate, efficient, and scalable speech recognition systems for Afaan Oromo and other under-resourced languages. Moreover, it would revolutionize human-machine interaction, enabling a broader range of applications and benefits for the vast speaker base of Afaan Oromo.

2.9. Afaan Oromo Language

The great Oromo people have their own culture, tradition, customs, and language and they are the largest ethnic group in Ethiopia and accounts for more than 40% of the population. This ethnic group also lives in some other parts of Africa like parts of Kenya and Somalia [19]. Afaan Oromo is an Afro-Asiatic language and the most widely spoken language of the Cushitic family. In addition to this, in Africa, it is the language with the fourth (4th) most speakers, after Arabic, Hausa, and Swahili [19]. As discussed in [19] in 1991 G.C, the Latin alphabet started being used for Afaan Oromo writing and was adopted as the official alphabet of Afaan Oromo. This writing system is called Qubee in Afaan Oromo. Now it is the language of public media, education, social issues, religion, political affairs, and technology.

Afaan Oromo Alphabets

Afaan Oromo Alphabets 'Qubee' contains 32 letters, twenty-six Latin letters, and an additional 6 double letters. The Afaan Oromo alphabets are classified into two main categories namely vowels ***'Qubee Dubbachiiftuu'*** (a, e, i, o, u) and consonants ***'Qubee Dubbifamaa'*** (b, c, d, f, g, h, j, k, l, m, n, p, q, r, s, t, v, w, x, y, z, ch, dh, ny, ph, sh, ts). Vowels in Afaan Oromo are 5 in number. However, the number of consonants is 27 [5].

Afaan Oromo vowels (Qubee Dubbachiiftuu) are categorized as long vowels and short vowels [5]. Short vowels use single vowel letters (like 'soba' which means False) and have a short sound and long vowels use double vowel letters ('dheebuu meaning thirsty) and have long sounds. Afaan Oromo consonants (Qubee dubbifamaa) are also two types geminated consonants and nongeminated consonants. Geminated consonants are consonants that have geminated sounds by doubling same consonants and non-geminated ones have a single consonant followed by a vowel. For instance, let us see two Oromo words 'sodaa' which have a non-geminated sound, which means fear and 'Soddaa' has geminated sound which means father-in-law. All Afaan Oromo consonants have geminate forms except 'h'and compound symbols [5].

In addition to the set of 32 symbols, individuals learning the Oromo writing system will also need to understand the guiding principles or rules associated with it.

1. Consecutive vowels indicate vowel length, for example, "Gaarii" means "Good".

2. Gemination, or doubling of a consonant, is significant in Oromo, such as "damee" (branch) and "dammee" (sweet potato).
3. The letter "h" is not doubled.
4. The same word can have multiple forms depending on the context, like "nama kadhu" (ask people) and "namaa kadhu" (ask for people).
5. Instead of accent signs, combined Latin letters such as ch, dh, ny, ph, sh, ts, and zh (zy) are used to align with typewriter characters.

The Latin alphabet has been adopted for use in various languages, including Germanic languages (English, German, Swedish, Danish, Norwegian, and Dutch), Romance languages (Italian, French, Spanish, Portuguese, and Romanian), Slavic languages (Polish, Czech, Croatian, and Slovenian), Finno-Ugric languages (Finnish and Hungarian), Baltic languages (Lithuanian and Latvian), as well as Somali, Swahili, and others [46]. The adoption of the Qube Afan Oromo alphabet, which aligns with the Latin script, offers several advantages. Firstly, it enables Oromo children who have already learned their native alphabet to easily learn the English script in a relatively short time. Additionally, the compatibility with computer technology makes alphabetic writing, including Qube Afan Oromo, more adaptable and advantageous compared to even the most basic syllabic writing systems [46].

Afaan Oromo Phonetics

Phonetics is the study of sounds (phonemes) of a language that produce a word. Afaan Oromo alphabets or letters have their International Phonetics Alphabet (IPA) representation [5]. The language utilized Latin letters as its foundation for constructing its distinct writing system, although the International Phonetic Alphabet (IPA) representation of these letters may vary from other languages. When compiling the IPA for this specific language, we took into account the phonetic sounds employed in our transcription.

Table 2. 1: Afaan Oromo alphabets with their IPA representations

Alphabets/Qubee	**A a**	**B b**	**C c**	**CH ch**	**D d**	**DH dh**	**E e**	**F f**	**G g**	**H h**	**I i**
IPA for Qubee	[a]	[b]	[ɗ]	[ʧ]	[d]	[ɖ]	[e]	[f]	[g]	[h]	[i]
Alphabets/Qubee	**J j**	**K k**	**L l**	**M m**	**N n**	**NY ny**	**O o**	**P p**	**PH ph**	**Q q**	**R r**
IPA for Qubee	[ʤ]	[k]	[l]	[m]	[n]	[ɲ]	[o]	[p]	[ƙ]	[q]	[r]
Alphabets/Qubee	**S s**	**SH sh**	**T t**	**TS ts**	**U u**	**V v**	**W w**	**X x**	**Y y**	**Z z**	**'**
IPA for Qubee	[s]	[ʃ]	[t]	[ts]	[u]	[v]	[w]	[ks]	[j]	[z]	[ʔ]

CHAPTER THREE

3. METHODS AND TECHNIQUES

3.1. Introduction

An experimental research design was employed for this study. In an experimental study, the researcher systematically manipulates and controls the testing conditions in order to investigate the causal relationships between variables. This approach allows the researcher to isolate the effects of specific factors and determine how they influence the outcome of interest [47] - in this case, the speech processing and modeling tasks. By carefully designing the experiments and controlling the testing environment, the researcher can draw more reliable conclusions about the underlying causal processes at work. The experimental research design was carried out using three main steps: creating and preparing the dataset, developing and implementing the model, and evaluating and hyperparameter adjustment. In order to enable experimental research in the speech domain, the researcher employed speech processing steps, including acquisition, segmentation, transcription and normalization. In order to experiment with different architectural configurations, such as different numbers of encoder, decoder, and FFNN elements specific to Transformers, the experimental research design was aligned with Speech Transformers.

3.2. Data Source

Dataset preparation is an initial and crucial step in speech recognition tasks, following best practice and guidelines established for other languages We can prepare well well-suited dataset for Afaan Oromo automatic speech recognition. Standard speech datasets typically consist of a training, validation, and test datasets [48]. The training dataset is used to acquire speech data for training the recognition system, while the evaluation test datasets are employed for the final assessment of the system's performance. Therefore, dataset preparation involves both audio dataset and its accurate transcription [20].

In this research, we adhered to established practices by dividing the dataset into training and testing portions. Specifically, we allocated 90% of the dataset for training purposes and reserved 10% for

testing, based on the findings of [20]. Previous papers have employed various allocations, such as 10% for testing and the remaining for training [12].

By following these established guidelines, we ensure that our dataset preparation aligns with industry standards and enables effective training and evaluation of the Afaan Oromo Large Vocabulary Continuous Speech Recognition system.

3.3. Speech Dataset Preparation

This speech dataset, consisting of news audio from various Oromia media outlets like Prime Media, OBN, OMN, MO'A, VOA, FBC, and BBC Afaan Oromo, served as the primary input for the developed speech recognition model. In total, the researchers collected approximately 18.04 hours of Afaan Oromo news audio from these different broadcast sources. This included around 3 hours from Prime Media, 6 hours from OBN, 1 hours from OMN, 2 hours from MO'A, 30 minutes hours from VOA, 2 hours from FBC, 3 hours from ETV and 30 minutes hours from BBC Afaan Oromo. Additionally, corresponding text transcriptions was prepared for the collected news audio data, consulting with linguistic experts to ensure accuracy. The full dataset consisted of over 8729 annotated utterances across the 18.04 hours of audio recordings.

3.4. Preprocessing of Speech Data

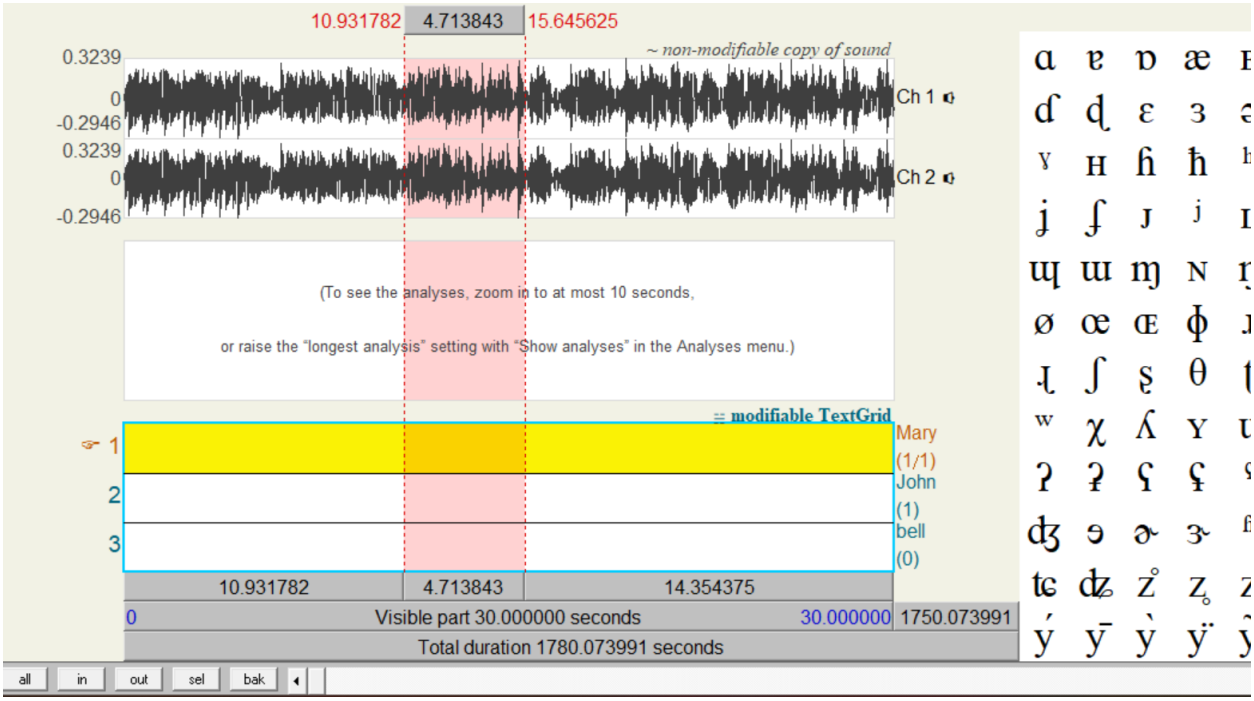
The speech data collected from various media sources could not be used directly, as it required preprocessing. Broadcast news speeches are not as clean and noise-free as read speech recordings. News speech also tends to be less slowly paced and has fewer pauses between words compared to carefully recorded read speech.

These properties of news speech make it more challenging for a speech recognition system to handle compared to carefully curated read speech data. Based on experiences with other language broadcast news datasets like Swahili and English, the researchers aimed to prepare the Afaan Oromo dataset using similar preprocessing techniques.

From the news speech data collected, the researchers only kept the continuous speech, excluding spontaneous speech like interviews and additional reports that were not audible. Speech containing background music and mixing Afaan Oromo with other languages like Amharic and English were also unused.

The Praat software was used to carefully segment Afaan Oromo's continuous speech from the long broadcast speeches as illustrated in the Figures Figure 3. 1: Skipping over the speech with background music.

Figure 3. 1: Skipping over the speech with background music



The Figure 3. 1: Skipping over the speech with background music is the praat displayed VOA news speech and the selected region has classical music in the background and we skipped to save the region like that. Most of the time these types of music are found in news speech at the beginning when they say headline news.

Figure 3. 2: Afaan Oromo and background music free part of the news speech

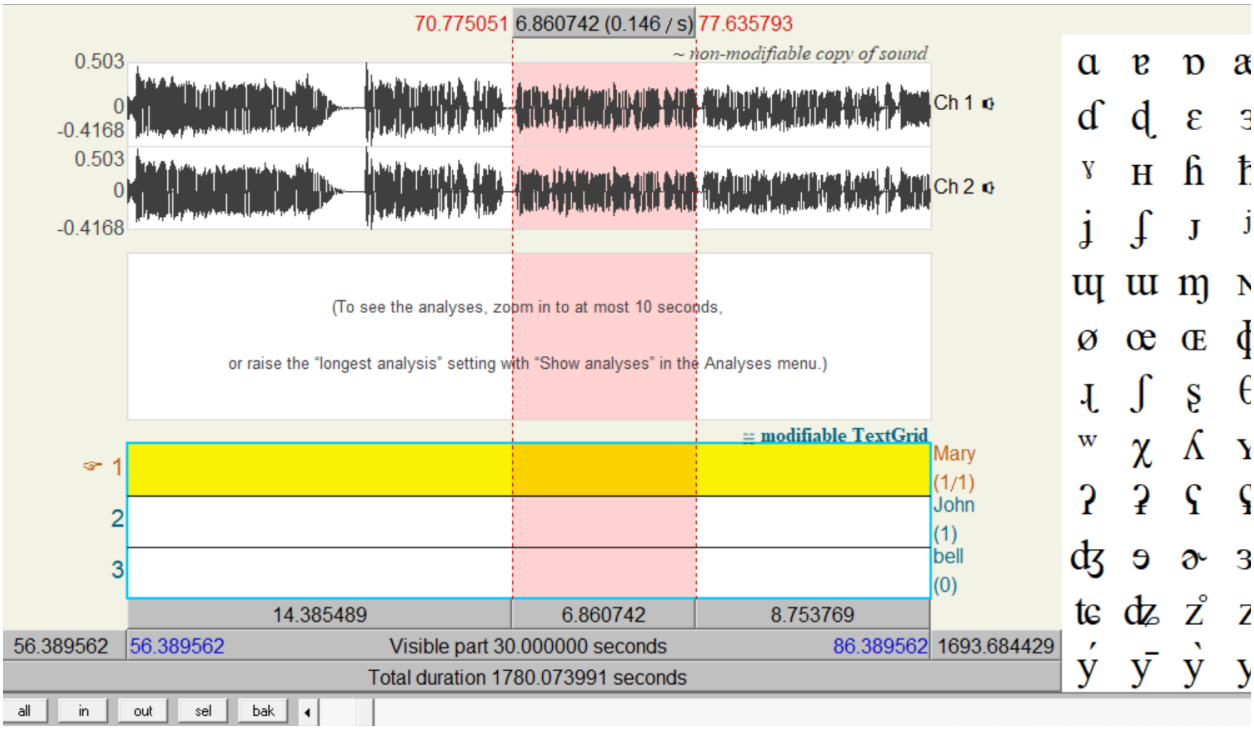
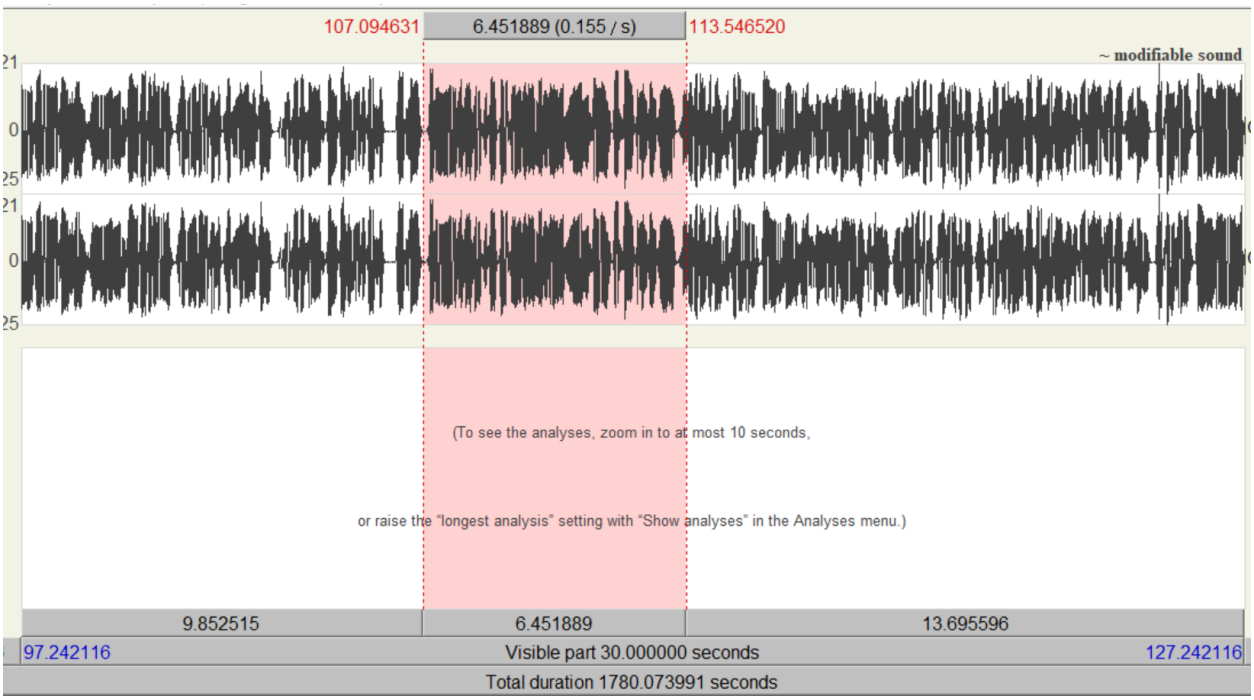


Figure 3. 2: Afaan Oromo and background music free part of the news speech above is the VOA news speech displayed on praat software and the selected region was music background and noise-free continuous Afaan Oromoo speech and such types of segments were accepted and saved for further processing for our dataset building.

3.4.1. Speech Segmentation

After the initial preprocessing steps to select a section of interest from the speech data, the next task was to segment the data into short clips. However, the researchers found that there were no existing automatic methods or tools available to perform this segmentation for the Afaan Oromo language. Due to the lack of automated segmentation capabilities, the researchers had to manually divide the preprocessed speech recordings short segments.

Figure 3. 3: Segmenting long speeches of broadcast news into short segments



Speecch segmentation was carried out manually by listening to audio files. A total of 8729 speech utterances, equivalent to 18.04 hours of speech were collected from 100 different speakers. Among the speakers, there were 50 males and 50 females. To facilitate the segmentation process, we utilized the software Praat, which allowed us to visualize, listen to, and segment the long speech.

While we aimed to achieve a balanced representation of male and female speakers, it was challenging due to the predominance of male speakers in broadcast news, including journalists, interviewees, and reporters. However, we were able to achieve equal numbers of male and female speakers, as desired by utilizing any of the continuous speech we found in the broadcast. This balance made our speech dataset unique, whereas the other researchers were not able to collect the same number of male and female speech datasets. For instance, Sifan[12] used 80 males and 21 females in her dataset. Similarly, Yadeta [5] collected a dataset of broadcast news from 57 speakers, consisting of 42 males and 15 females.

To ensure compatibility with machine learning purposes, we utilized an open-source tool called Audacity to convert the segmented audio files into the WAV format, channel to mono and sample rate to 16000 Hertz (Hz) which is commonly used in machine learning applications.

3.4.2. Transcription of Segmented Speech

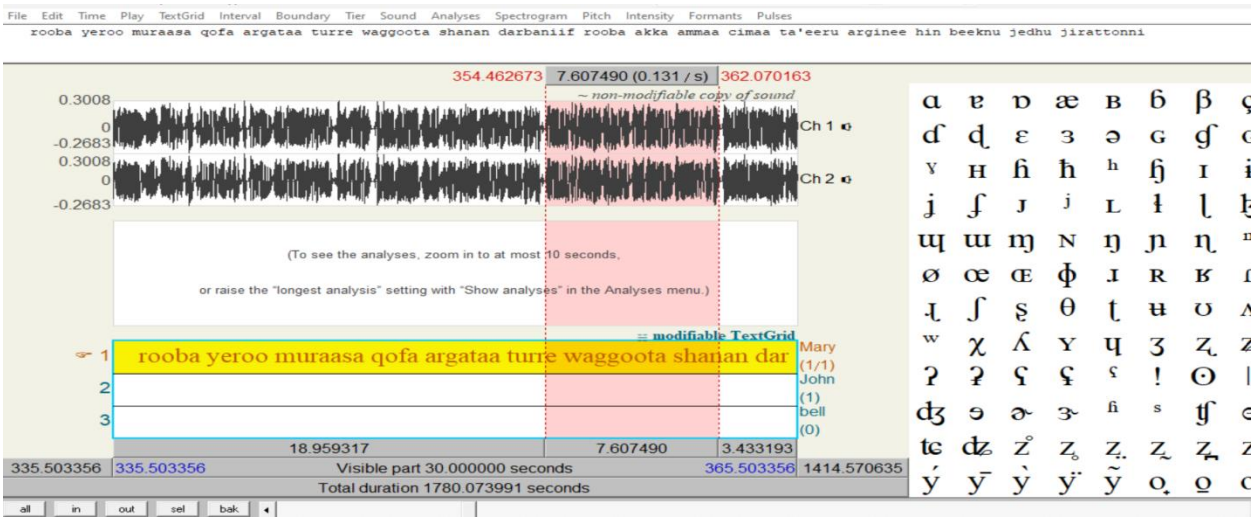
The process of transcribing the segmented speech into corresponding text was done manually using the praat software. The researchers consulted linguistic experts and various literature sources to ensure the transcriptions adhered to the rules of Afaan Oromo grammar.

Praat is a free, widely-used computer software package for speech analysis in the field of phonetics. It was designed and continues to be developed by researchers at the University of Amsterdam. Praat supports a range of functionalities, including speech synthesis and articulatory synthesis[49].

Using the Praat software, the researchers manually transcribed a total of 8729 Afaan Oromo utterances into text and all the transcribed text was in lowercase and with comma, space,hyphen, apostrophe and question mark punctuations.

The manual transcription process was evaluated and corrected by linguistic experts, for which we paid compensation. This step ensured the accuracy of the text transcriptions corresponding to the audio data.

Figure 3. 4: Transcription of the selected region for segmentation



3.5. Text Preprocessing

Our Automatic Speech Recognition (ASR) project has a detailed text preprocessing pipeline to normalize the textual data in order to make it usable for model training purposes. The initial steps taken were the setting up of a character vocabulary that also contained some important characters like lowercase Afaan Oromo letters, common punctuation marks and special tokens. Specifically, there were 32 characters making up this vocabulary namely; twenty six lowercase Afan Oromo alphabets (a-z), four punctuations and special symbols ("-", " ' ", ".", ","), which includes two special markers ("<" and ">") denoting the beginning and end of a text sequence respectively. By means of this word list, it was possible to transform each sign into its corresponding numerical index.

The text subsequently underwent normalization. In the beginning, the text was converted to lowercase so that uniformity and case differences caused by variability could be minimized. A regular expression function deleted special characters not in the predefined vocabulary thereby ensuring that text data remained clean and relevant. On top of this, numeric data in the text were written as words (e.g., 123 to tokko lama sadii) for consistent representation of text and facilitating recognition.

In addition, padding techniques were employed in order to achieve uniform input length across all text sequences. The text was not truncated but instead, it was padded based on the length of the longest transcription available in the dataset. This way each and every text sequence was padded to have the maximum number of characters ever observed thus preserving all data integrity in inputs. In that case, special start (‘<’) and end (‘>’) tokens were added at both ends of the text sequences to indicate when a new sentence begins and when an old one terminates.

The processed text was then converted into a sequence of numerical indices using the character-to-index mapping derived from the predefined vocabulary. This comprehensive preprocessing ensured that the text data fed into the ASR model was clean, consistent, and suitably formatted, thereby enhancing the model's performance and accuracy.

Additionally, the audio data was directly processed by creating Mel spectrograms from the raw audio signals. This process involved computing the Short-Time Fourier Transform (STFT) of the audio files and then applying a Mel filter bank to the resulting spectrogram to generate the Mel

spectrogram. The Mel spectrograms align more closely with human auditory perception by emphasizing perceptually relevant frequency components [39]. Following this transformation, normalization was applied to standardize the feature values. Finally, the Mel spectrograms were padded to a consistent length of 3059 which is equivalent with the maximum clip in second to ensure uniform input dimensions for the model. This rigorous preprocessing framework, encompassing both text and audio data, was integral to the development of an effective and reliable ASR system. The inclusion of 32 distinct characters in the vocabulary, covering both alphabetic characters and essential punctuation and special tokens, ensured comprehensive coverage and accurate representation of the input text data.

Table 3. 1: Statistics of the Dataset

Name	**Value**
Total Clips	8729
Total words	127472
Total Characters	984522
Total Durations(seconds)	64957.20
Mean Clip Duration(seconds)	7.44
Minimum Clip Duration(seconds)	1.52
Maximum Clip Duration(seconds)	14.95
Mean Words Per Clip	14.60
Distinct Words	25831

3.6. Feature Extraction Techniques

The research undertook a comparative analysis of two different feature extraction techniques, spectrogram and Mel spectrogram in order to find out their performance in the context of Afaan Oromo large vocabulary continuous speech recognition. While at first we considered the spectrogram because it could provide a fine-grained frequency-time representation of audio signals[39], we also evaluated the Mel spectrogram due to its conformity with human hearing system that emphasizes frequencies more audible to human ear and maps power of spectrum to Mel scale[39].

To systematically compare these techniques, we implemented both methods within our preprocessing pipeline and assessed their impact on the performance of our speech recognition model. The spectrogram was generated using the Short-Time Fourier Transform (STFT), which provided a comprehensive representation of the audio signal across various frequency bins [39]. In contrast, the Mel spectrogram was derived by applying a Mel filter bank to the spectrogram, thereby converting the frequency bins into the Mel scale and producing a more human-auditory-centric representation[34].

Our empirical evaluation revealed that models trained with Mel spectrogram features consistently outperformed those using spectrogram features. The Mel spectrogram's ability to better capture the nuances of human speech contributed to improved accuracy and robustness in recognizing Afaan Oromo speech. This superior performance can be attributed to the Mel spectrogram's emphasis on perceptually relevant frequency components, which enhances the model's ability to discern phonetic details crucial for accurate speech recognition.

Consequently, based on these findings, we adopted the Mel spectrogram as the primary feature extraction technique for the final development of our speech recognition model. This decision was substantiated by the substantial improvement in model performance, demonstrating the Mel spectrogram's effectiveness in capturing the essential characteristics of speech signals pertinent to the Afaan Oromo language. This methodological choice underscores the importance of selecting appropriate feature extraction techniques aligned with human auditory perception to enhance the accuracy and reliability of speech recognition systems.

3.7. Training and Evaluation Tools

In the development of the automatic speech recognition (ASR) project, the researcher utilized the freely available Google Colab platform as the computing environment. Google Colab, also known as Collaboratory, is a cloud-based Jupyter Notebook service that provides a convenient and accessible platform for running various machine learning and data analysis tasks.

To implement the ASR system, the researchers leveraged the widely adopted TensorFlow and Keras libraries. TensorFlow is an open-source machine learning framework that provides a comprehensive set of tools and APIs for building and deploying deep learning models. Keras, on the other hand, is a high-level neural networks API that runs on top of TensorFlow, simplifying the development of complex deep learning models.

3.8. The Model Architecture

Speech recognition refers to a machine or program's capability to identify spoken words and phrases and convert them into a format understandable by computers. Traditional machine learning-based approaches to speech recognition often have limitations in vocabulary and data, and they can only accurately identify words and phrases if they are spoken clearly[32].

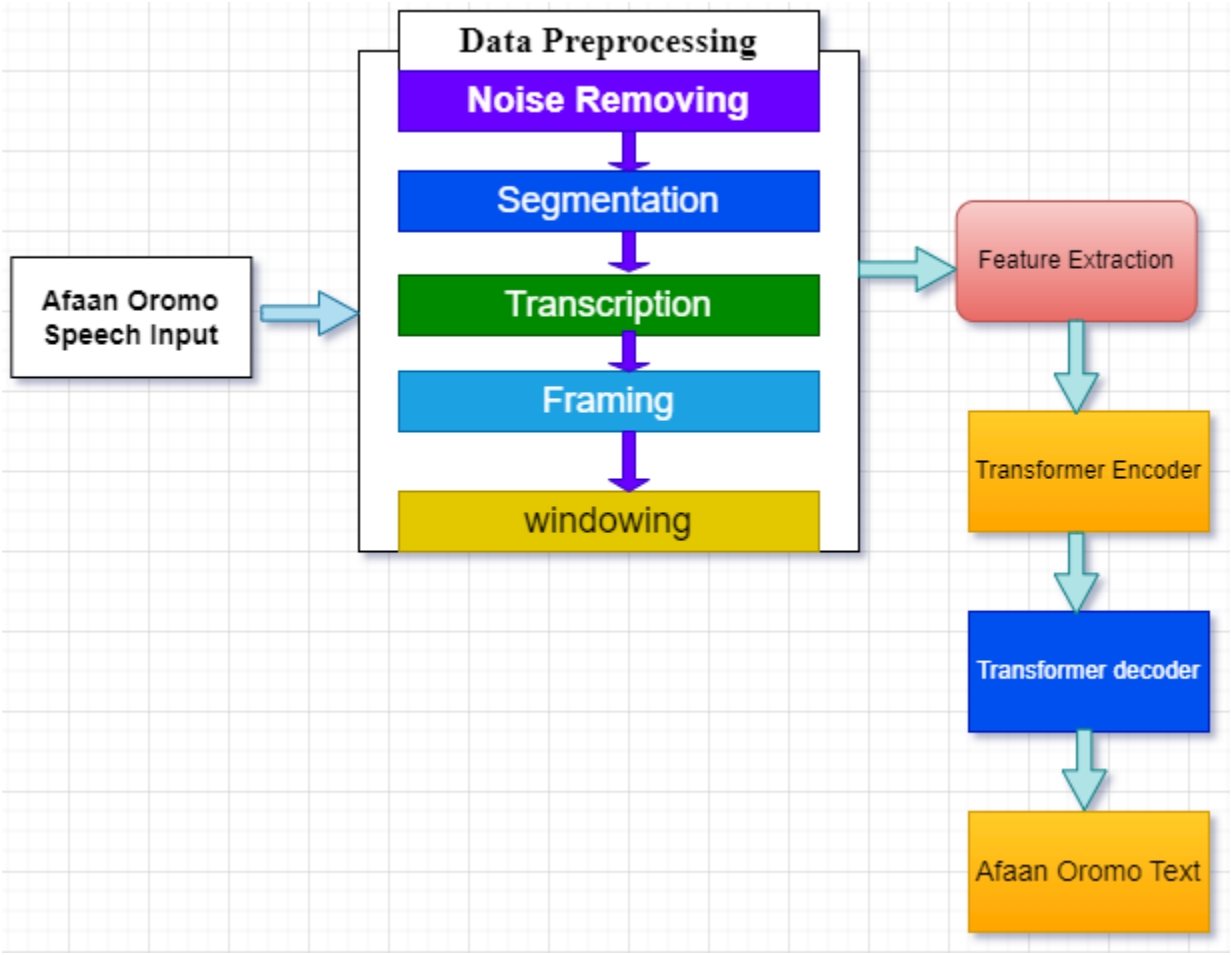
However, recent advancements in deep learning have revolutionized the field of speech recognition. These advancements have allowed computers to learn and comprehend speech through experience, leading to significant improvements in various artificial intelligence tasks, including speech recognition. Deep learning's inherent deep architecture enables it to tackle more complex AI tasks.

The task of recognizing and transcribing human speech has gained significant importance in recent times. There is a growing interest among researchers in employing End-to-End models for automatic speech recognition (ASR). Previous approaches for Afan Oromo ASR have relied on architectures such as statistical methods, traditional machine learning and time-delay neural networks, recurrent neural networks (RNNs), and long short-term memory (LSTM) networks[4], [5], [6], [7], [12], [32]. However, traditional end-to-end approaches have faced challenges such as slow training and inference speed due to limitations in data entry parallelization. Additionally,

these approaches often require a large amount of data to achieve satisfactory results in recognizing speech[22].

As a result, this research proposes an Afan Oromo speech recognition system based on a speech transformer: nonrecurrence encoder-decoder architecture with the self-attention mechanism. The use of transformers enables faster and more efficient training of the model[50]. With this approach, Afan Oromo audio speech segments can be accurately transcribed into text.

Figure 3. 5: Architecture of how the proposed model works



3.8.1. Overview of the Architecture

The research project on automatic speech recognition for the Afaan Oromo language followed a well-structured approach, as outlined in the Figure 3. 5: Architecture of how the proposed model worksabove. The first step was to collect the required speech data. The researchers gathered Afaan Oromo broadcast news speech from various sources, including OBN, OMN, VOA, FANA, EBC,

and BBC. This extensive data collection process aimed to create a comprehensive dataset for the subsequent development of the ASR system. After collecting the speech data, the next step was the preprocessing of the dataset. This preprocessing stage consisted of five key steps:

1. Background noise removal: The researchers used a speech or non-speech discriminator to separate the speech data from non-speech elements, effectively removing background noise.

2. Segmentation: The long broadcast news audio files were segmented into individual sentence-level recordings, as the direct use of the lengthy broadcast data was not suitable for transcription and model training purposes.

3. Transcription: Since there were no available automatic tools for transcribing Afaan Oromo speech, the researchers manually transcribed the segmented audio recordings into corresponding text.

4. Framing: The audio signals were split into short-time frames, typically ranging from 20 to 40 milliseconds. This step allows for better approximation of the frequency contours of the speech signal by concatenating the adjacent frames.

5. Windowing: After framing, the researcher applied a window function, such as the Hamming window, to each frame. This step helps to counteract the assumption made by the Fast Fourier Transform (FFT) that the data is infinite and to reduce spectral leakage.

Following the preprocessing stages, the researchers extracted features from the cleaned and segmented speech data. These extracted features were then used to train speech transformer model chosen as the recognition algorithm for the Afaan Oromo ASR system. Finally, the Afaan Oromo text output was generated, and the system's performance was tested using a separate test dataset. Details of the rest components, which included in the architecture, are described below.

3.9. Proposed Architecture

As shown in Figure 3. 5: Architecture of how the proposed model works, the architecture proposed in this study for Afaan Oromo speech recognition classification was based on an original speech-transformer architecture. Speech transformer is a neural network architecture developed for automatic speech recognition. They depart from the traditional sequence-to-sequence approach

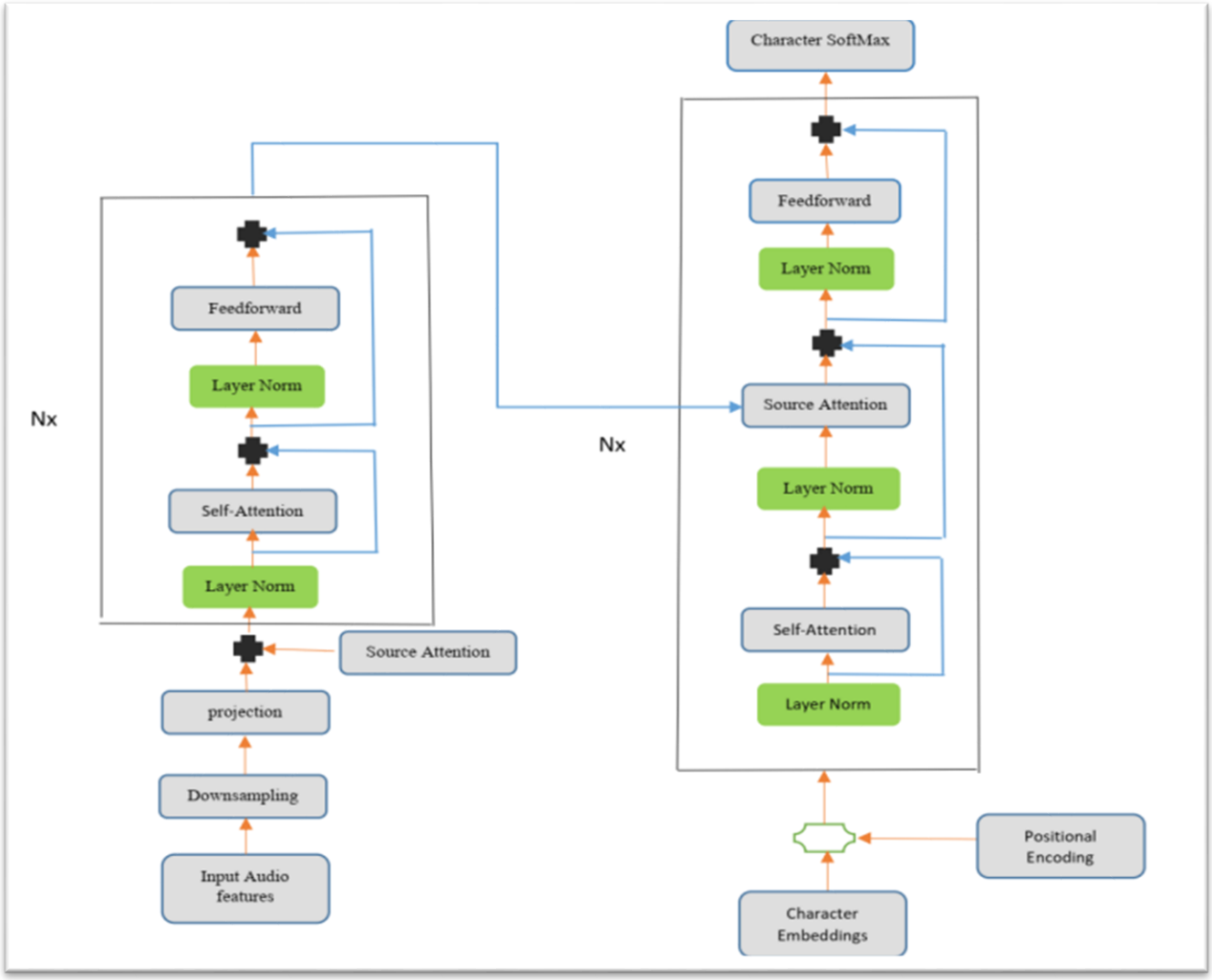
and instead leverage the transformer architecture, which initially gained prominence in natural language processing (NLP) tasks.

To downsample the acoustic features and capture local relationships, our approach involves applying four convolutional layers with convolution strides. Additionally, when processing previous target tokens in the decoder, we calculate the sum of the token and position embeddings. The main objective of our speech recognition-transformer model is to convert a sequence of speech features into a corresponding sequence of characters. This transformation is typically applied to a two-dimensional spectrogram and/or mel spectrogram, which has time and frequency axes.

Encoder-Decoder Block

The model comprises two essential components: the encoder and the decoder. The encoder takes in the source sequence and generates a high-level representation, while the decoder is responsible for generating the target sequence. Both the encoder and decoder are neural networks that contain learnable components capable of capturing the relationship between input and output time steps. The decoder also incorporates a process to condition specific encoder representation components. Unlike recurrent neural networks (RNNs), the Transformer model relies on multi-head attention or its attention mechanism as its core element. Our model utilizes input audio spectrograms to predict a sequence of characters. During training, the target character sequence is fed to the decoder. Inference involves the decoder predicting the next token based on its previous predictions[13], [14].

Figure 3. 6: Speech transformer model architecture



Attention

Attention refers to the process of extracting relevant information from a set of queries (Q), keys (K), and values (V) based on their similarities. The Scaled Dot Product Attention, proposed in [14], forms the foundation of the retrieval function. It involves computing the weighted sum of values using the similarities between queries and keys. In this method, queries (Q) are represented by a matrix $Q \in R^{tqxdq}$, keys (K) by a matrix $K \in R^{tqxdq}$, and values (V) by a matrix $V \in R^{tqxdq}$. Here, t_* represents the number of elements in the inputs, and d_* represents the corresponding element dimension. Typically, the dimensions of keys (dk) and values (dv) are the same, as are the

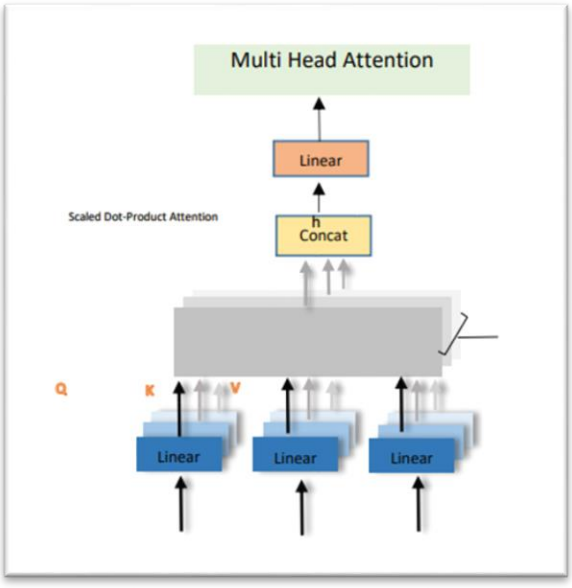
element numbers (tk = tv) and the query dimension (dq). The self-attention mechanism calculates the outputs as follows:

$$Attention(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{dk}})$$

Multi-Head Attention

The transformer model incorporates multiple dot-product attention layers. In the recent improvement proposed in [13], the dot-product attention is scaled by introducing a sub-space projection for the matrices $K, Q, and\ V$ into n parallel heads. This means that n attention operations are performed with corresponding heads for each operation. The outputs from each attention head are then concatenated to form a chain of attention outputs. Notably, unlike recurrent connections that use a single state with gating structures for data transmission, or convolution connections that linearly combine local states based on a kernel size, self-attention aggregates information from all time steps without any intermediate transformations. The scaled dot-product attention is computed individually for each head, and the resulting outputs are sequentially fed into other linear projections to generate the final output with the desired dimensionality d_{model} as shown in **Error! Reference source not found.**.

Figure 3. 7: Multi-head attention



Feature Extraction Algorithms

Feature extraction refers to a procedure that identifies significant characteristics or attributes of the data[34]. Its objective is to convert the speech waveform into a parametric representation with a reduced data rate, enabling further processing and analysis. Initially, the sound data needs to be converted into a format that can be inputted into our model [39]. A commonly used approach is to divide the sound into a series of frames and then extract features from each frame. The detail was discussed in section 2.5

In our research, we utilized the mel-spectrogram as the feature extraction technique for the speech transformer model employed in our automatic speech recognition (ASR) system. The mel-spectrogram is a widely adopted representation of speech signals that captures the spectral characteristics of the audio in a way that aligns with the human auditory system's perception of sound.

The computation of the mel-spectrogram involves several key steps. First, we divided the input speech signal into short, overlapping frames, typically in the range of 20-30 milliseconds. For each frame, we then applied the Short-Time Fourier Transform (STFT) to obtain the magnitude spectrum. Next, we filtered the magnitude spectrum using a bank of triangular filters spaced along the mel-frequency scale, which is a non-linear transformation of the linear frequency scale that better approximates the human auditory system's response to sound. Finally, we applied a logarithmic scale to the filtered values to better represent the subjective loudness perception, resulting in the final mel-spectrogram representation.

The mel-spectrogram preserves more of the spectral information compared to the commonly used mel-frequency cepstral coefficients (MFCC) feature extraction technique. This additional spectral information can be beneficial for certain speech recognition tasks, as it provides a more comprehensive representation of the audio signal's characteristics.

In the context of our ASR system, we chose to use the mel-spectrogram as the input feature representation for the speech transformer model. The speech transformer, a powerful neural network architecture specifically designed for processing sequential data such as speech, is well-suited to leverage the rich information contained in the mel-spectrogram. By feeding the mel-spectrogram into the speech transformer, we aimed to enable the model to effectively learn the

complex relationships between the spectral characteristics of the speech signal and the corresponding text transcription, ultimately leading to improved ASR performance.

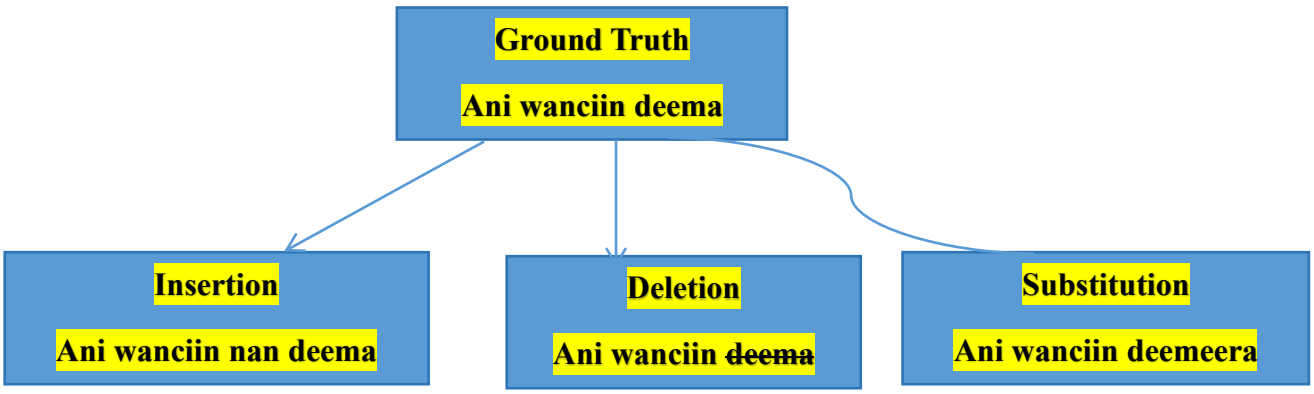
The utilization of the mel-spectrogram as the feature extraction technique, combined with the capabilities of the speech transformer model, was a key component of our ASR system's design and implementation. This approach allowed us to capture the nuanced spectral information of the speech signals, which was crucial for accurately recognizing and transcribing the spoken language in our target application.

3.10. Evaluation Metrics

The essential step in the development of any AI model is to establish a measure for assessing its performance [11]. It is the fundamental process to make the ML progress effective and error-free. The researchers want to know how likely to trust the proposed system prediction is and wants to know the probability of the system performance and prediction well. In the classification problem of an ML, there is a Word Error Rate (WER) that evaluates the model performance.

Word Error Rate (WER) is a fundamental evaluation metric used in speech recognition to measure the accuracy of the system's output compared to a reference transcription. WER provides a quantitative measure of the performance of an ASR system by calculating the percentage of incorrect words in the recognized output [13]. WER considers three types of errors: substitutions, insertions, and deletions. A substitution occurs when a recognized word differs from the corresponding word in the reference transcription. An insertion refers to an extra word inserted in the recognized output that is not present in the reference. A deletion occurs when a word is missing from the recognized output compared to the reference. By considering these different error types, WER provides a comprehensive assessment of the system's performance [25].

Figure 3. 8: Word Error Rate



The calculation of WER involves aligning the recognized output and the reference transcription at the word level. The aligned words are then compared, and the number of substitutions, insertions, and deletions is counted. The WER is computed by dividing the total number of errors by the total number of words in the reference transcription [25].

The WER is computed by dividing the total number of errors by the total number of words in the reference transcription

$$WER = (S + D + I) / N$$

S= number of substitutions,

D= number of deletions,

I= number of insertions,

N = number of words in the reference

WER is expressed as a percentage, where lower values indicate higher accuracy. For example, a WER of 10% means that 10% of the words in the recognized output differ from the reference transcription. In practical terms, a lower WER indicates better performance and higher accuracy of the ASR system.

WER is a widely accepted metric in the speech recognition community because it provides a comprehensive evaluation of system accuracy, considering both word-level errors and the overall performance. It allows researchers s and practitioners to compare different ASR systems, evaluate the impact of various techniques and algorithms, and track the progress of system development over time [11].

CHAPTER FOUR

RESULT AND DISCUSSION

In this chapter, we present the details of our experiments and discuss the outcomes in developing a model for Afaan Oromo large vocabulary continuous speech recognition. The prototype is based on the Speech Transformer model. The development process involves several steps, including data collection, data preprocessing, hyperparameter adjustment, model training, the recognition step, and evaluation.

4.1. Experimental Setup

Training involved the use of two common deep learning instruments, TensorFlow and Keras. Google developed and issued TensorFlow, which is a flexible collection of libraries and tools for improving the field of deep learning [51]. Therefore, we have chosen TensorFlow as our backend to train our proposed model. To execute our project, we made use of Google Colab which is publicly available for free. We preferred Google Colab on its provision on computation resources as well as its convenience to access.

To ensure uninterrupted progress and mitigate potential issues such as connection disruptions or the expiration of the free usage time provided by Google Colab, we implemented a specific strategy. We adopted a code-saving approach that involved regularly saving the model code and the model's checkpoints had been saved at every 10 epochs. This practice was crucial in ensuring that we could resume the code execution from the most recent checkpoint, thereby avoiding the need to restart the entire process in the event of interruptions, such as a connection failure or the expiration of Google Colab's free usage time.

By preserving the model code at regular intervals, we aimed to minimize time-consuming setbacks and maintain a consistent workflow throughout our research project. This proactive measure effectively managed potential interruptions or time constraints associated with the usage limitations of Google Colab, thereby ensuring a smooth continuation of our research experiment and training activities. During the training phase, researchers utilized a state-of-the-art deep learning algorithm for continuous speech recognition[17].

We experimented with four models with different super hyperparameters namely encoder, decoder, and number of feed-forward neural networks. The first model was trained with one encoder and one decoder with 200 feed-forward neural networks. The second model was trained with three encoders and two decoders with 300 feed-forward neural networks. The third model was trained with five encoders and three decoders with 400 feed-forward neural networks. The fourth was trained with 8 encoders and six decoders with 500 feed-forward neural networks.

Experimenting with different numbers of encoders, decoders, and feed-forward neural networks (FFNNs) is crucial in the development of automatic speech recognition (ASR) systems using speech transformers due to the significant impact these components have on model performance and efficiency[14]. The architecture of a speech transformer, which includes stacks of encoders and decoders, directly influences the model's capacity to capture and process complex acoustic patterns and linguistic structures in speech signals[14].

Varying the number of encoders and decoders allows researchers to balance the trade-offs between model complexity, computational cost, and accuracy. Increasing the number of encoders can enhance the model's ability to extract detailed features from the input speech while adjusting the number of decoders can improve the decoding process, which translates these features into text. Similarly, experimenting with different configurations of FFNNs within each transformer layer can optimize the transformation and representation of encoded information, impacting the overall model's ability to generalize from training data to unseen speech inputs. Therefore, systematic experimentation with these architectural elements is essential to identify the optimal configuration that achieves the highest performance in terms of recognition accuracy, computational efficiency, and robustness in automatic speech recognition tasks using a speech transformer.

4.2. Parameters For Training the Models

Table 4. 1: Training parameter of the models

Number	**Parameters to Train the Recognizer**	**Values (for each model)**	**Description of Parameters**
1	Learning Rate	Custom Schedule (see description)	The learning rate schedule with initial, after warmup, and final learning rates.

2	Batch Size	64	Number of samples per gradient update.
3	Number of Encoder Layers	1, 3, 5, 8	Number of layers in the encoder for each model.
4	Number of Decoder Layers	1, 2, 3, 6	Number of layers in the decoder for each model.
5	Feed-forward Inner Layer Dimension	200, 300, 400, 500	The dimensionality of the inner feed-forward layer for each model.
6	Model Dimension (d_model)	512	The dimensionality of input and output vectors.
7	Number of Attention Heads	4	Number of parallel attention heads in each multi-head attention layer.
8	Dropout Rate (Self Attention)	0.5	Dropout rate for self-attention layers.
9	Dropout Rate (Encoder)	0.1	Dropout rate for encoder layers.
10	Dropout Rate (Feed-forward)	0.1	Dropout rate for feed-forward layers.
11	Warmup Steps	Derived from 15 epochs	Number of steps for learning rate warmup.
12	Decay Steps	Derived from 85 epochs	Number of steps for learning rate decay.
13	Steps per Epoch	Training dataset/batch-size	Number of steps per epoch.

14	Maximum Input Length	Variable	Maximum length of input sequences (depends on dataset).
15	Maximum Output Length	Variable	Maximum length of output sequences (depends on dataset).
16	Vocabulary Size	32	Number of unique tokens in the output vocabulary.
17	Adam Beta1	0.9	Beta1 parameter for the Adam optimizer.
18	Adam Beta2	0.98	Beta2 parameter for the Adam optimizer.
19	Adam Epsilon	1e-9	Epsilon parameter for the Adam optimizer to prevent division by zero.
20	Gradient Clipping	1.0	Maximum norm for gradient clipping.
21	Number of Epochs	100	Total number of training epochs.
22	Early Stopping	6	Training will terminate if the weights do not improve for 6 consecutive epochs.

In this research, as expressed in Table 4. 1: Training parameter of the models, the training process was governed by a carefully selected set of super hyperparameters, which were varied across different models to optimize performance. The primary focus was on varying the number of encoder and decoder layers, as well as the dimensionality of the feed-forward neural network while keeping other parameters constant across each model experiment.

The learning rate was controlled using a custom schedule designed to adapt throughout the training process. Initially set to 0.00001, the learning rate increased to 0.001 after a warmup period derived from 15 epochs, before gradually decaying back to 0.00001 over the subsequent 85 epochs. This approach, combined with a batch size of 64 samples per gradient update, helped stabilize training and improve convergence. Additionally, the Adam optimizer was employed with parameters beta1=0.9, beta2=0.98, and epsilon=1e-9 to efficiently handle the gradient updates. Gradient clipping was applied with a maximum norm of 1.0 to prevent the gradients from exploding.

The architecture of the models tested included variations in the number of encoder and decoder layers and the size of the feed-forward network. The first model was configured with one encoder and one decoder layer, and a feed-forward network dimension of 200. The second model utilized three encoder layers and two decoder layers, with a feed-forward network dimension of 300. The third model incorporated five encoder layers and three decoder layers, with a feed-forward network dimension of 400. Finally, the fourth model was designed with eight encoder layers and six decoder layers, and a feed-forward network dimension of 500. Across all models, the embedding dimension (d_model) was set to 512, and each multi-head attention layer comprised four attention heads.

Dropout was applied at different stages of the model to prevent overfitting. Specifically, a dropout rate of 0.5 was used for self-attention layers, while encoder and feed-forward layers both had a dropout rate of 0.1. The maximum input and output sequence lengths were determined based on the characteristics of the dataset, ensuring the model could effectively handle variable-length sequences. The vocabulary size was fixed at 32 unique tokens, reflecting the specific requirements of the Afaan Oromo language.

The training process was set to run for a maximum of 100 epochs. To ensure efficient training and prevent overfitting, early stopping was implemented with a criterion to terminate training if the model weights did not improve for six consecutive epochs. This combination of hyperparameters and training strategies was designed to optimize the performance of the ASR model and ensure robust recognition of the Afaan Oromo language.

In summary, the parameters chosen for this research were carefully selected to balance model complexity and training efficiency. The variations in the encoder and decoder layers, along with the feed-forward network size, were critical in exploring the optimal configuration for the ASR

task using the Speech Transformer. Through this structured approach, we aimed to advance the performance and reliability of automatic speech recognition for the Afaan Oromo language.

4.3. The Proposed Model Construction and Training

After data preprocessing and extraction of important features from the preprocessed data, the recognizer has to be constructed and trained.

1. **The first model the researchers selected for training the recognizer was one encoder and one decoder with 200 feedforward neural networks.**

The first model, which was trained with one encoder, one decoder, and 200 feed-forward neural networks, achieved a WER of 65.7%. The results obtained from the first model suggest that the relatively simple architecture and lower model capacity may have limited its ability to effectively capture the complex linguistic and acoustic features present in the Afaan Oromo speech data. With only a single encoder and decoder layer, the model may have struggled to learn the necessary hierarchical representations required for accurate speech recognition.

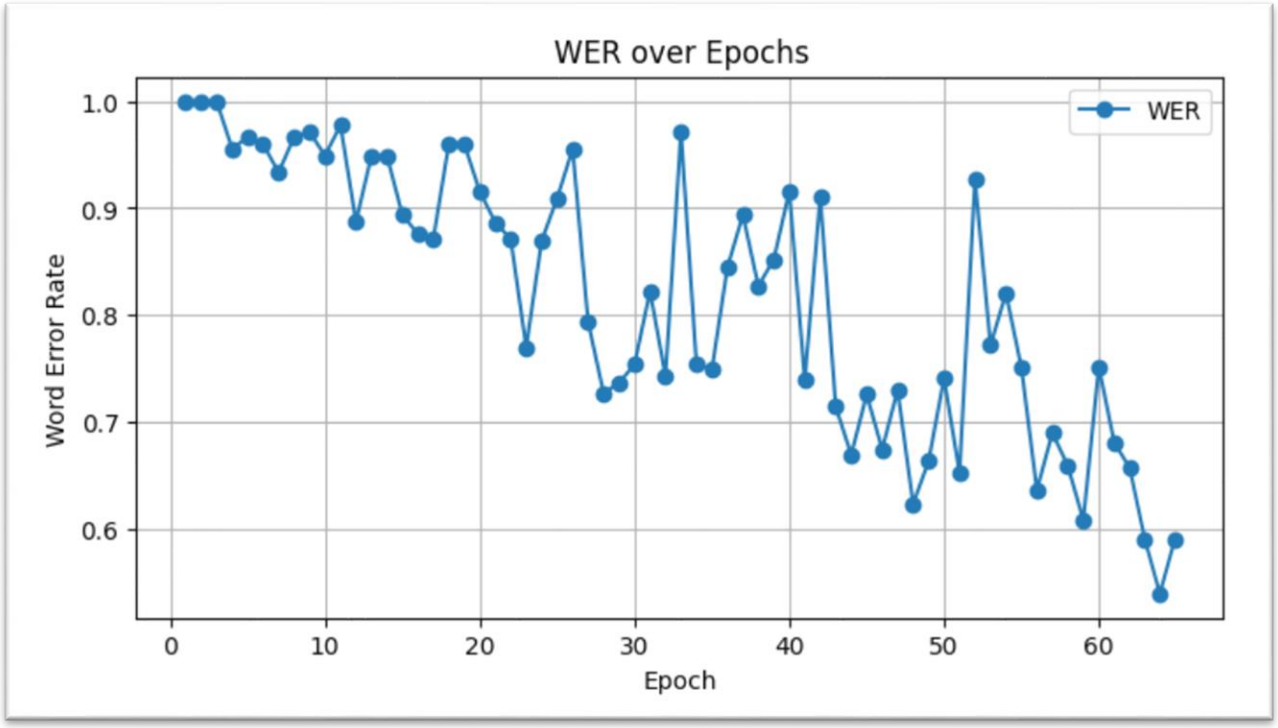
Furthermore, the 200 feed-forward neural networks may not have provided sufficient modeling capacity to learn the intricate patterns and variations present in the Afaan Oromo speech dataset. The limited network size could have hindered the model's ability to generalize and handle the diverse vocabulary and speaking styles encountered in the large vocabulary continuous speech recognition task.

The 65.7% WER indicates that the first model while providing a baseline performance, falls short of the desired performance for a practical large vocabulary continuous speech recognition system in the Afaan Oromo language. The high WER suggests that a significant proportion of the recognized words do not match the reference transcripts, which can negatively impact the overall usability and user experience of the speech recognition system.

To address these limitations, we plan to further explore the performance of the second, third and fourth models, which feature deeper encoder-decoder architectures and higher model capacities. By increasing the complexity and representational power of the Speech Transformer models, we aim to achieve a significant improvement in accuracy and a corresponding reduction in word error rate for Afaan Oromo's large vocabulary continuous speech recognition.

The findings from this initial experiment provide valuable insights into the challenges and opportunities for developing an effective speech recognition system for the Afaan Oromo language using the Speech Transformer architecture. We are committed to continuing our experiment and optimizing the model configurations to deliver a more accurate and reliable speech recognition solution for Afaan Oromo speakers.

Figure 4. 1: The first Experiment's WER plot diagram



The Figure 4. 1: The first Experiment's WER plot diagram shows the Word Error Rate (WER) values over the course of 60 epochs of training for the first experiment. The WER starts at around 0.95 and fluctuates significantly during the early epochs, indicating a highly unstable training process. However, as the training progresses, the WER gradually decreases, reaching a minimum value of around 0.55 around epoch 40. After this point, the WER begins to rise again, suggesting potential overfitting. The training was terminated at epoch 60 due to the preset early stopping condition, although the maximum number of epochs was set to 100. This behavior suggests that further tuning of the model hyperparameters may be necessary to achieve optimal performance on this task.

2. **The second model the researcher selected for training the recognizer was 3 encoders, 2 decoders, and 300 feedforward neural networks.**

The second model, which was trained with 3 encoders, 2 decoders, and 300 feed-forward neural networks, achieved a 53%-word error rate (WER). While the second model represented an increase in architectural complexity compared to the first model, the results suggest that the chosen hyperparameters may have been optimal for effectively capturing the intricate linguistic and acoustic characteristics of the Afaan Oromo speech data compared with the first model. The 3 encoder and 2 decoder layers, while more sophisticated than the single encoder-decoder configuration of the first model, have provided sufficient depth and representational capacity to model the hierarchical relationships and contextual dependencies present in the Afaan Oromo speech dataset and also deduced the WER evaluation while compared with the first model experiment.

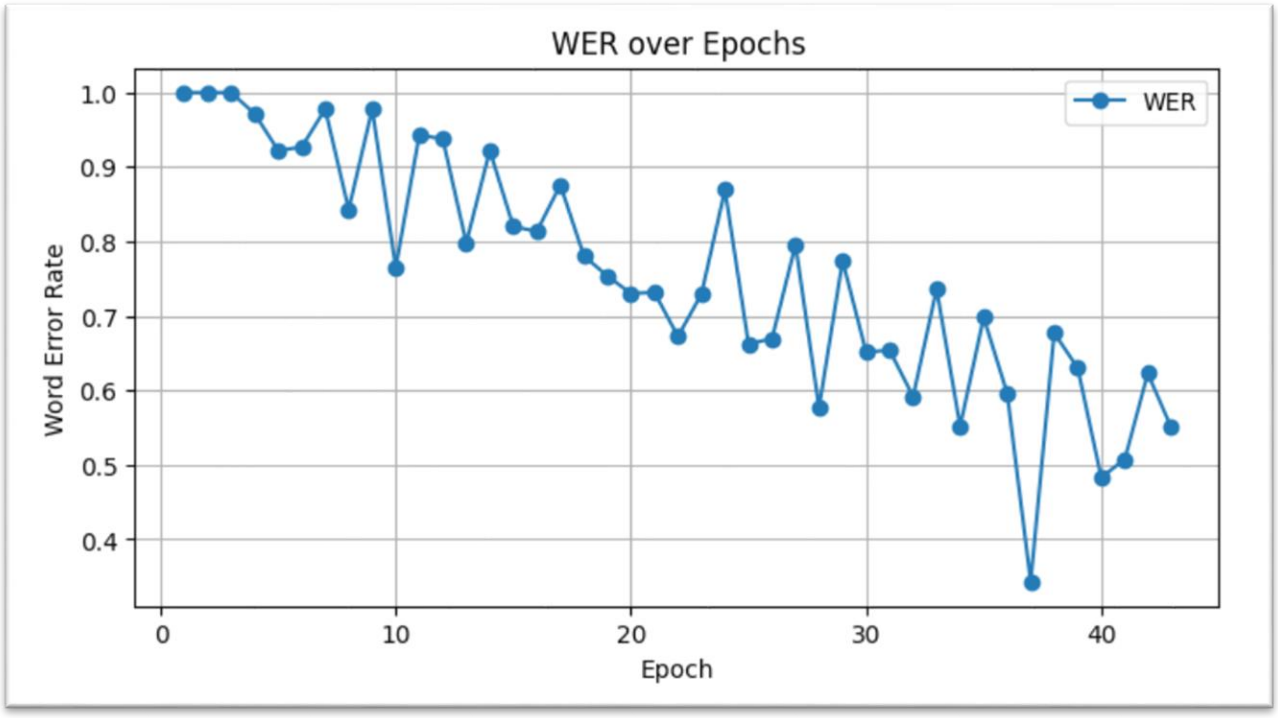
Furthermore, the 300 feed-forward neural networks, while an improvement over the 200 in the first model, may still have been insufficient to learn the complex patterns and variations inherent in the large vocabulary continuous speech recognition task for the Afaan Oromo language. The 53% WER indicates that the second model, while demonstrating improved performance compared to the first model, still falls short of the desired level of accuracy and reliability for a practical speech recognition system. The high WER suggests that a significant proportion of the recognized words do not align with the reference transcripts, which can significantly impact the overall transcription ability of the model.

To address these limitations, we plan to further investigate the performance of the third model, which features an even deeper encoder-decoder architecture and higher model capacity. By increasing the depth and representational power of the Speech Transformer models, we aim to achieve a more substantial improvement in accuracy and a corresponding reduction in word error rate for Afaan Oromo's large vocabulary continuous speech recognition.

The findings from this experiment provide valuable insights into the trade-offs between model complexity, capacity, and performance for Afaan Oromo speech recognition using the Speech Transformer architecture. We will continue to refine our approach, explore additional hyperparameter configurations, and leverage our understanding of the language-specific

characteristics to develop a more accurate and reliable speech recognition solution for Afaan Oromo.

Figure 4. 2: The second experiment's WER plot diagram



3. **The third model the researchers selected for training is the recognizer 5 encoder, 3 decoder, and 400 feedforward neural network.**

The third model, which was trained with 5 encoders, 3 decoders, and 400 feed-forward neural networks, achieved a 40.2%-word error rate (WER). With the increased architectural complexity and higher model capacity of the third model, the results suggest that the chosen hyperparameters have been optimal for effectively modeling the unique linguistic and acoustic characteristics of the Afaan Oromo speech data concerning the previous two model experiments. The 5 encoder and 3 decoder layers, while representing a significant increase in depth compared to the previous models, may have provided the necessary representational power to fully capture the intricate relationships and contextual dependencies inherent in the Afaan Oromo speech dataset.

Furthermore, the 400 feed-forward neural networks, while the highest among the three models have been able to learn the complex patterns and variations present in the large vocabulary continuous speech recognition task for the Afaan Oromo language. The 40.2% WER indicates that

the third model, with its increased complexity, has achieved an improved level of performance compared to the two previous model experiments. The low WER suggests that a substantial proportion of the recognized words are aligned with the reference transcripts, which can positively impact the overall usability and user experience of the Afaan Oromo speech recognition system. These findings highlight the possibility of developing an effective Speech Transformer model for Afaan Oromo large vocabulary continuous speech recognition. The insights gained from this research will inform our ongoing efforts to develop a reliable and user-friendly speech recognition solution for Afaan Oromo speakers, contributing to the advancement of language technology and accessibility in the region.

Figure 4. 3: The WER of our model's optimum result-third experiment

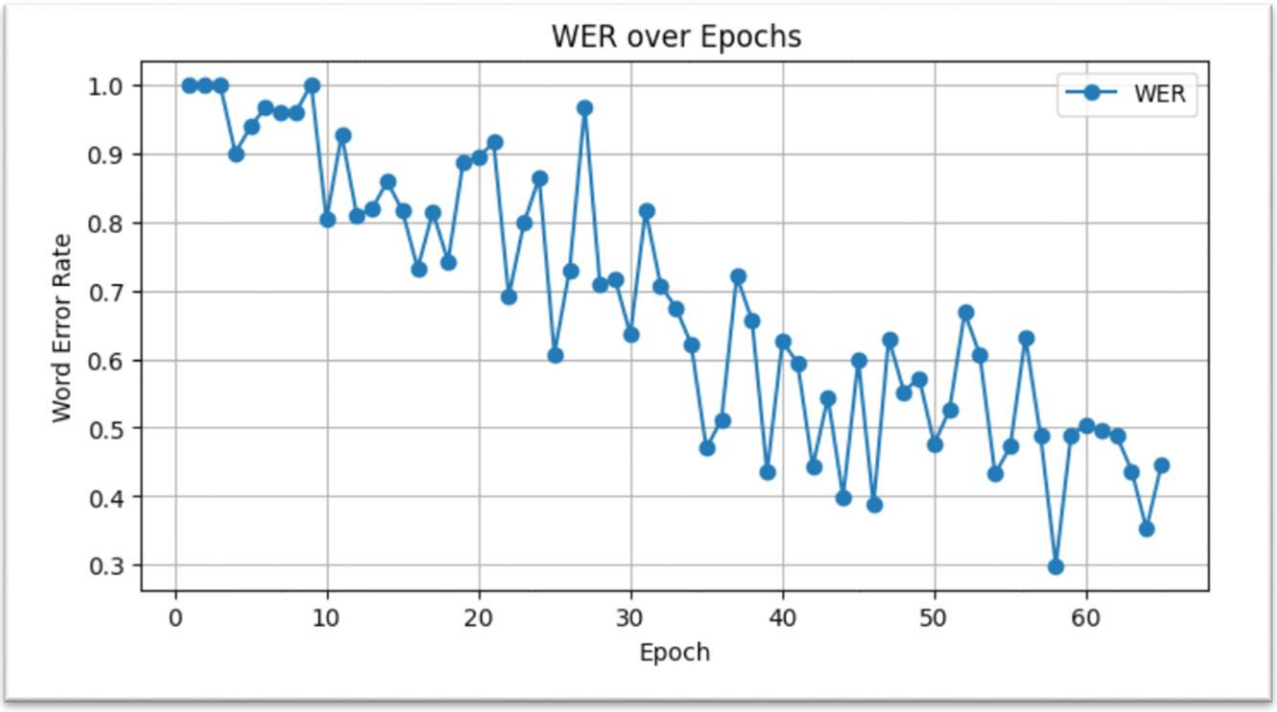


Figure 4. 3: The WER of our model's optimum result-third experiment above is the result of our optimum model word error rate it indicates that at epoch 1 the model has a WER of above 98% percent and as the number of epochs increases the WER decreases and at 65 epoch at which the model terminates by itself with the preset early stopping function the model obtains high accuracy or low WER of about 40.2%. This indicates that the model learns well. This depicts the performance of the optimum model in terms of word error rate (WER) throughout training. The key observations are as follows:

1. At the initial epoch (epoch 1), the model exhibits a very high WER of over 98%. This indicates that the model's performance at the beginning of the training process was quite poor, with a significant number of word errors in the predictions.

2. As the training progresses and the number of epochs increases, the WER steadily decreases. This suggests that the model is learning and improving its ability to make accurate predictions over time.

3. By the time the model reaches 65 epochs, the training process automatically terminates due to the preset early stopping function. At this point, the model has achieved a much lower WER of approximately 40.2%.

The decreasing trend in WER throughout training demonstrates that the model is learning effectively and improving its performance. The low WER of around 40.2% at the end of the training process indicates that the model has achieved a high level of accuracy in its predictions, which is a desirable outcome for the given task. This observation provides valuable insights into the model's learning capabilities and its ability to capture the underlying patterns in the data. The gradual performance improvement, as evidenced by the decreasing WER, suggests that the model has the potential to generalize well to unseen data and deliver reliable predictions in the intended application context.

For all of our experiments, we have kept their respective WER plot diagram and they show similar improvement and they are only different in terms of their WER. So, we have not provided a description like this for other experiments, because we assume that this description is enough since their difference is only the number of WER they got.

4. Fourth model, with 8 encoders, 6 decoders, and 500 feed-forward neural networks

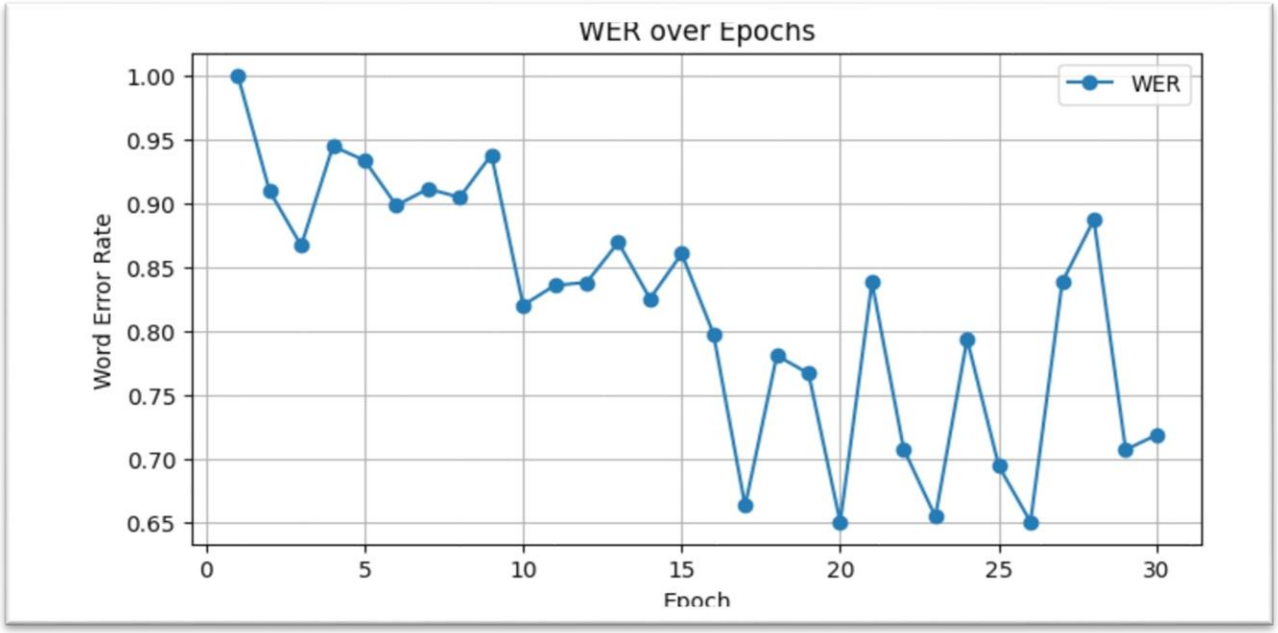
As we have experimented with the Speech Transformer models for Afaan Oromo large vocabulary continuous speech recognition, we have gained valuable insights into the impact of varying the key hyperparameters of the model architecture.

In our experiments, we observed that the performance of the Speech Transformer models, as measured by the word error rate (WER), was influenced by the complexity of the model structure and the size of the training dataset.

The fourth model, which was trained with 8 encoders, 6 decoders, and 500 feed-forward neural networks, achieved a WER of 71%. This result, while shows less result compared to the previous models, indicates that the increased architectural complexity may have exceeded the capacity of the available training data and the freely available GPU. As we increased the number of encoders to 8, decoders to 6, and feed-forward neural networks to 500, the model's ability to capture the intricate linguistic and acoustic patterns of the Afaan Oromo speech data decreased. Perhaps, the 71% WER suggests that the model was unable to generalize effectively, potentially due to overfitting on the limited training data and the machines' power.

The inverse relationship between model complexity and performance highlights the importance of carefully balancing the model architecture and the dataset size for optimal results this low-resource language. While the increased model capacity allowed for more sophisticated feature learning, the training data may not have been sufficient to fully leverage these capabilities.

Figure 4. 4: The fourth experiment's WER plot diagram



Results and Discussion

Various hyperparameters were adjusted during the training of the Afaan Oromo speech recognition model in order to achieve the desired results. The number of encoders (Ne), decoders (Nd), and numbers of FFNNs were modified, as documented in Table 4. 2: Result of recognizer modelExperiments that resulted in overfitting or underfitting were excluded from consideration. The objective was to find the optimal combination of hyperparameters that would yield the best performance for the model.

Table 4. 2: Result of recognizer model

Model	No: Encoder	No: Decoder	Feedforward layer	WER%
1	1	1	200	65.7%
2	3	2	300	53%
3	5	3	400	40.2%
4	8	6	500	71%

The most favorable outcome was attained by employing five encoders and three decoders and 400 FFNN, resulting in a word error rate of 40.2%. This configuration yielded the best performance for the Afaan Oromo speech recognition model.

One notable problem observed during the analysis of errors made by the proposed model is the occurrence of truncated output. Many of the generated output texts are considerably shorter than the corresponding source transcripts which leads to increase the number of the WER. This issue significantly affects the performance of the model, as it fails to produce output sentences of sufficient length. Automatic speech recognition faces numerous challenges, particularly when dealing with datasets that contain background noise as broadcast speech data has some background noise. Furthermore, variations in speakers, word speed, accents, pitch, and other factors pose additional challenges that need to be addressed for accurate speech recognition.

Now we see the performance of our optimal model on the different speaker types of the broadcst speech mainily males and females.

The experiments conducted on the speech recognition model revealed interesting findings regarding the performance across different speaker demographics. Notably, we were able to achieve a balanced representation of both female and male speech samples within the dataset used

to train the model. As a result, the speech recognition system demonstrated the ability to recognize female and male voices with equal accuracy.

When analyzing the model's performance on female and male speech samples separately, the results indicated that the recognition rates were comparable between the two speaker groups. This balanced performance suggests that the model was able to effectively generalize its learning to accurately process speech input from both genders, without exhibiting any bias or disparity in recognition accuracy. The strategic approach to ensuring equal representation of female and male speakers in the training data appears to have been a key factor in enabling the model to perform equitably across these demographic distinctions.

```
File ID: LAS00488
Target: ‹naannon kunis naannoo saarkaameentoo fi kibba kaalifoorniyaa maaliibuudha.›
Predicted: ‹naannon kunis nanno sarkamento fi kibba kaaalifoorniya maaliiibuuda›
Word Error Rate (WER): 0.5

File ID: LAS00062
Target: ‹ijaarsicha boqonnaa xummuraa irra gahuu amerikaatti ambaasaadderri itoophiyaa fi falmaa olaanaan hidha haaromsa guddichaa ›
Predicted: ‹ijaarsicca boqonna xummuraa irra gahuu amerikatti ambaaasaadderri itooophiyaa fi falmaa olanaan hida goywebbd btrcmqpbeg›
Word Error Rate (WER): 0.6428571428571429

File ID: LAS00003
Target: ‹itoophiyaan walgayii boordii daarekterootaa tumsa fooramii giloobaal paabilik sekuriitii chaayinaa irratti›
Predicted: ‹itoophiyaaan walgayi bordiii daaarekterooota tumsa foooramiii giloobaal paaabilik sekuriitii cyndjrawkr apffdue›
Word Error Rate (WER): 0.7272727272727273
```

The evaluation of the speech recognition model's performance revealed further insights regarding its ability to accurately process speech input from both female and male speakers. Specifically, the analysis of individual test files provided clear evidence of the model's balanced recognition capabilities. For instance, the file labeled LAS00062 was identified as a female speaker, while the file with the ID LAS00488 corresponded to a male speaker. The consistent and accurate classification of these individual samples, representing both genders, strongly indicates that the model has attained a high level of parity in its recognition of female and male speech patterns.

Finally, we were able to answer our research questions based on our findings as follows:

Developing a speech recognition system for Afaan Oromo using the Speech Transformer model presented several challenges. One significant issue encountered was truncation in the generated transcriptions especially for long clips. The model often produced output predictions that were incomplete or shorter than the actual spoken text. This discrepancy between the predicted and true transcriptions suggested difficulties in the model's ability to fully capture and reproduce the intricate phonetic and linguistic features of Afaan Oromo. This problem frequently occur when the speech is long and have many words, we can see from the Figure 4. 5: Test result with optimul model below that the truncated text produced meaningless collection of character text. Addressing this challenge required increasing dataset more and more with increasing model complexity and exploring techniques to improve the model's capacity to handle longer sequences and maintain transcription fidelity.

Additionally, evaluating the performance of our ASR model for Afaan Oromo using the Word Error Rate (WER) metric posed significant challenges. The WER is calculated as the sum of substitutions (S), insertions (I), and deletions (D) divided by the total number of words in the reference (N), formulated as: $WER = S + I + D /N$

This metric can be particularly challenging for Afaan Oromo due to its complex writing system, which includes features such as consonant gemination and the distinction between long and short vowel letters. These linguistic intricacies mean that even minor errors in transcription can significantly impact the WER score. For instance, a single character change, omission, or addition can result in a substitution error, making it difficult to accurately assess the model's performance. This complexity necessitated a more nuanced approach to evaluating the ASR system, one that takes into account the unique characteristics of the Afaan Oromo language. For instance these diagram shows prediction of our final model for different speechs. As we can see from the below diagram there is a time when even the WER is 1 however the predicted texts are still acceptable for human.

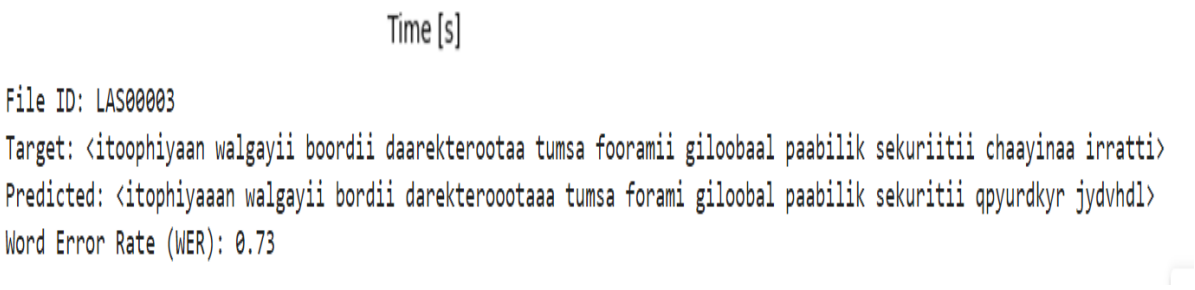


Figure 4. 5: Test result with optimul model

```
File ID: LAS00376
Target: <haalli nageenyaa dhabamuun oromiyaa keessaa>
Predicted: <haaalli nageenaaa dabamun oromiyaaa kessaa>
Word Error Rate (WER): 1.0

File ID: LAS00471
Target: <Himatamtoonnis oofiisar Gabramaariyaam Waldaay Abrahaa.>
Predicted: <himatamtoonnis ooofisar gabramaaariyaam waldaay abrahaaa>
Word Error Rate (WER): 1.0

File ID: LAS00135
Target: <kan dubbannuufi kan tarkaanfannu akkasuma kan barbaannus nageenyi biyya keenyatti akka waarawuufi>
Predicted: <kan dubbannuufi kan tarkaaanfannu akkasuma kan barbaannus nageni biyya keenatti gfbn yothizzuog>
Word Error Rate (WER): 0.4166666666666667
```

```
File ID: LAS00377
Target: <ammaataa adeemuun illee ibsameeraa>
Predicted: <ammaata ademuun ille ibsameraa>
Word Error Rate (WER): 1.00
```

In evaluating the performance of our Speech Transformer-based ASR system for Afaan Oromo, we found notable improvements over previously developed deep learning model. Specifically, our model demonstrated a superior ability to accurately transcribe words with consonant gemination and the long and short vowel behaviors of the Afaan Oromo sounds, which are critical phonetic features of the language. These improvements highlight the Speech Transformer's enhanced capability to model complex linguistic patterns and its effectiveness in handling the unique characteristics of Afaan Oromo speech.

Previous research on Afaan Oromo ASR, such as the work by Sifan [12] using deep learning models faced significant challenges. As they reported in their work, they said ‘the CTC algorithm,

which was developed and tested primarily for languages like English, struggled with the unique writing system of Afaan Oromo. Specifically, it could not accurately handle geminated consonants and distinguish between long and short vowels. In many other languages, such as English, double consonants and vowels are not used to indicate gemination and length, respectively. Consequently, the CTC algorithm often misinterpreted these features, treating them as repetitions of the same word spoken slowly and removing them'

In contrast, our Speech Transformer-based model as shown in the Figure 4. 5: Test result with optimul model above successfully addressed these issues, accurately transcribing the geminated consonants and long and short vowels of Afaan Oromo. This ability to correctly capture and transcribe these phonetic features is vital for producing accurate and intelligible transcriptions, thus representing a significant advancement in the field of ASR for under-resourced languages like Afaan Oromo.

Through a series of experiments, we identified the optimal hyperparameters that yielded the best performance for our Speech Transformer-based ASR system. Among the four configurations tested, the model with five encoder layers, three decoder layers, and a feed-forward neural network dimension of 400 provided the highest accuracy(lower WER) which was 40.2% and transcription correctness. This configuration effectively balanced model complexity and computational efficiency, allowing the system to learn and generalize well from the training data. The selection of these hyperparameters was crucial in optimizing the model's performance and achieving significant improvements in transcription accuracy for Afaan Oromo speech recognition.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

This chapter comprises a comprehensive analysis of the conclusions and recommendations drawn from the research findings. Additionally, it outlines prospective paths for future investigations, thereby providing significant insights and guidance to fellow researchers within the respective field.

5.1. Conclusion

In this research, we proposed an Automatic Speech Recognition (ASR) system specifically designed to recognize and transcribe the Afan Oromo speech. Our approach utilized a Speech Transformer model that leverages multi-head attention, feedforward neural networks, layer normalization, and positional encoding to capture positional dependencies in the input speech data. In order to do the thesis, we collected speech data from Afaan Oromo broadcasts. We had consulted different literatures about features of the language, development of ASR from broadcast news in general and the related materials were also reviewed. The proposed system achieved remarkable accuracy in recognizing Afan Oromo speech and demonstrated improved efficiency compared to previous recurrent models, reducing the training cost significantly.

The speech corpus required for this study was collected from a variety of sources. Specifically, audio data was gathered from OMN, MO'A, OBN, FANA, EBC, VOA, WALIIF TV, ETV and BBC Afaan Oromo broadcasts. The audio data was then segmented and transcribed using the Praat and audecity softwares. However, the scarcity of previous broadcast speech dataset for the Afan Oromo language, the researchers were able to utilize 8729 total data samples along with their corresponding transcriptions from total of 100 different speaker of 50 females and 50 males to ensure the representativeness of the data. The audecity software was used to convert the audio data into .wav format and changing channel types to mono and also samples rate to 16000 Hertz for each speech. For speech feature extraction from the raw speech signals, the researchers employed Jupyter Notebook with Python programming. Finally, TensorFlow and Keras were utilized for training the system, recognizing test utterances, and analyzing the results of the speech recognizer.

Through the Speech-Transformer model we developed, we achieved state-of-the-art performance with a Word Error Rate (WER) of 40.2% when evaluated on a broadcast dataset. This was accomplished by employing five encoders, three decoders and 400 FFNN in the model architecture among the conducted experiments.

Notably, we trained the ASR model exclusively on the Broadcast speech dataset of Afan Oromo language, without employing a separate language model. Despite using a relatively modest training dataset of duration 18.04 hours, our system achieved impressive results in recognizing Afan Oromo speech.

5.2. Recommendations and Future Works

For future work, we suggest expanding the training dataset to include more data, which could potentially lead to even better results. Additionally, training the system by considering the equalization of all dialects of Afaan Oromo can help optimize the benefits of speech recognition applications across various Oromo communities, organizations, institutions, and companies.

Additional research could investigate the integration of advanced acoustic modeling techniques and language modeling approaches to enhance the accuracy and robustness of the Afaan Oromo LVCSR system. Adding a language model could potentially improve our correctness a lot, so this is also among our next steps.

Reference

[1] I. V. McLoughlin, *Speech and Audio Processing*. 2016. doi: 10.1017/cbo9781316084205.

[2] "What Is Automatic Speech Recognition (ASR)?," Level AI. Accessed: Jan. 15, 2024. [Online]. Available: https://thelevel.ai/blog/automatic-speech-recognition-asr/#:~:text=Key Applications of ASR,-Automatic speech recognition&text=Speech-to-text%3A ASR,voice-to-text dictation.

[3] K. Foster, "What is Automatic Speech Recognition? A Comprehensive Overview of ASR Technology," AssemblyAI. Accessed: Jan. 02, 2024. [Online]. Available: https://www.assemblyai.com/blog/what-is-asr/

[4] M. M. P. A. F. Abdurrahman, "Artificial Neural Network Based Afan Oromo Speech Recognition System," Jimma University, Jimma. [Online]. Available: https://repository.ju.edu.et//handle/123456789/5530

[5] Y. G. Gutu, A. A. Science, M. Y. Tachbelie, and N. L. Applications, "A Large Vocabulary , Speaker-Independent , Continuous Speech Recognition System for Afaan Oromo : Using Broadcast News Speech COLLEGE OF NATURAL AND COMPUTATIONAL SCIENCES SCHOOL OF INFORMATION SCIENCE A LARGE VOCABULARY , SPEAKER-INDEPENDENT , CONTINUOUS," no. June, 2022, doi: 10.13140/RG.2.2.26635.98089.

[6] K. Gelana, "A Continuous, Speaker Independent Speech Recognizer for Afaan Oroomoo: Afaan Oroomoo Speech Recognition Using HMM Model," 2011. [Online]. Available: https://api.semanticscholar.org/CorpusID:203685750

[7] K. Teshite, G. Mamo, and K. Calpotura, "Afan Oromo Speech-Based Computer Command and Control: An Evaluation with Selected Commands," *Adv. Human-Computer Interact.*, vol. 2023, 2023, doi: 10.1155/2023/9959015.

[8] T. Kebebew, "Speech Recognition for Afan Oromo using hybrid hidden markov models and artificial neural networks," Addis Ababa, Ethiopia, 2010.

[9] J. Oruh, S. Viriri, and A. Adegun, "Long Short-Term Memory Recurrent Neural Network for Automatic Speech Recognition," *IEEE Access*, vol. 10, pp. 30069–30079, 2022, doi:

10.1109/ACCESS.2022.3159339.

[10] D. Huizhen, “L Ocal S Elf -a Ttention Based C Onnectionist T Emporal C Lassification,” pp. 275–284, 2020.

[11] Z. Tüske, B. Kingsbury, and D. Bolanos, “ADVANCING RNN TRANSDUCER TECHNOLOGY FOR SPEECH RECOGNITION George Saon , Zoltan Tuske , Daniel Bolanos and Brian Kingsbury,” no. 2, pp. 5639–5643, 2021.

[12] S. D. Degefa, “Afaan Oromo Continuous Speech Recognition Using Deep Learning,” Jimma University, 2021. [Online]. Available: https://repository.ju.edu.et/handle/123456789/6163

[13] A. Vaswani *et al.*, “Attention is all you need,” *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 5999–6009, 2017.

[14] L. Dong, S. Xu, and B. Xu, “Speech-Transformer,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 5884–5888, 2018, [Online]. Available: http://150.162.46.34:8080/icassp2018/ICASSP18_USB/pdfs/0005884.pdf

[15] M. Malik, M. K. Malik, K. Mehmood, and I. Makhdoom, “Automatic speech recognition: a survey,” *Multimed. Tools Appl.*, vol. 80, pp. 9411–9457, 2021.

[16] V. B. Kumar and K. Padmaveni, “A Review on Evolution of Automatic Music Generation Using Machine Learning Techniques,” *AIP Conf. Proc.*, vol. 2794, no. 1, 2023, doi: 10.1063/5.0165678.

[17] M. Maurya, “Navigating the Evolution of Automatic Speech Recognition (ASR),” Institute of Machine Learning and Artificial Intelligence. Accessed: May 13, 2024. [Online]. Available: https://lamarr-institute.org/blog/automatic-speech-recognition-evolution/

[18] A. L. Tonja *et al.*, “Natural Language Processing in Ethiopian Languages: Current State, Challenges, and Opportunities,” *4th Work. Resour. African Indig. Lang. RAIL 2023 - Proc. Work.*, pp. 126–139, 2023, doi: 10.18653/v1/2023.rail-1.14.

[19] D. B. Teferii, “The Development of Oromo Writing System,” p. 295, 2015.

[20] S. O. Ogun, “How to create a speech dataset for ASR, TTS, and other speech tasks,” Jekyll Bootstrap. [Online]. Available: https://ogunlao.github.io/blog/2021/01/26/how-to-create-speech-dataset.html#data-split

[21] Teferi, “Afaan Oromo speech recognition using hybrid HMM/ANN,” Addis Ababa, 2010.

[22] S. Latif, A. Zaidi, and H. Cuay, “Transformers in Speech Processing : A Survey,” pp. 1–27.

[23] S. Alharbi *et al.*, “Automatic Speech Recognition : Systematic Literature Review,” *IEEE Access*, vol. 9, pp. 131858–131876, 2021, doi: 10.1109/ACCESS.2021.3112535.

[24] M. Fank, “Audio Analysis With Machine Learning: Building AI-Fueled Sound Detection App,” *Altextsoft*, [Online]. Available: https://www.altexsoft.com/blog/audio-analysis/

[25] B. Juang and L. Rabiner, “Automatic Speech Recognition - A Brief History of the Technology Development,” Jan. 2005.

[26] Z. Song, “English speech recognition based on deep learning with multiple features,” *Computing*, vol. 102, no. 3, pp. 663–682, 2020, doi: 10.1007/s00607-019-00753-0.

[27] A. Shrestha and A. Mahmood, “Review of deep learning algorithms and architectures,” *IEEE Access*, vol. 7, pp. 53040–53065, 2019, doi: 10.1109/ACCESS.2019.2912200.

[28] W. Burger and M. J. Burge, *Scale-Invariant Feature Transform (SIFT)*. 2022. doi: 10.1007/978-3-031-05744-1_25.

[29] C. Monier, Y. Fr, and A. P. Davison, “A comprehensive data-driven model of cat primary visual cortex . Acknowledgments,” pp. 1–54, 2019.

[30] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, 2022, doi: 10.1109/TNNLS.2021.3084827.

[31] X. Ying, “An Overview of Overfitting and its Solutions,” *J. Phys. Conf. Ser.*, vol. 1168, no. 2, 2019, doi: 10.1088/1742-6596/1168/2/022022.

[32] K. B. Bhangale and M. Kothandaraman, *Survey of Deep Learning Paradigms for Speech Processing*, vol. 125, no. 2. Springer US, 2022. doi: 10.1007/s11277-022-09640-y.

[33] M. Zambra, A. Testolin, and M. Zorzi, “A Developmental Approach for Training Deep Belief Networks,” *Cognit. Comput.*, vol. 15, no. 1, pp. 103–120, 2023, doi: 10.1007/s12559-022-10085-5.

[34] A. Tabassum and R. R. Patil, “A Survey on Text Pre-Processing & Feature Extraction Techniques in Natural Language Processing,” *Int. Res. J. Eng. Technol.*, no. June, pp. 4864–4867, 2020, [Online]. Available: www.irjet.net

[35] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, “Listen, Attend and Spell,” pp. 1–16, 2015, [Online]. Available: http://arxiv.org/abs/1508.01211

[36] D. D. Rivers, D. Ph, and R. A. Rosen, *by Daniel D. Rivers, Ph.D. and Robert A. Rosen Hughes Aircraft Company, El Segundo, California, USA*, vol. 7. 1982.

[37] T. Zhang and J. Wu, “Discriminative frequency filter banks learning with neural networks,” *Eurasip J. Audio, Speech, Music Process.*, vol. 2019, no. 1, pp. 1–16, 2019, doi: 10.1186/s13636-018-0144-6.

[38] H. F. Pardede, V. Zilvan, D. Krisnandi, A. Heryana, and R. B. S. Kusumo, “Generalized Filter-bank Features for Robust Speech Recognition against Reverberation,” *2019 Int. Conf. Comput. Control. Informatics its Appl. Emerg. Trends Big Data Artif. Intell. IC3INA 2019*, pp. 19–24, 2019, doi: 10.1109/IC3INA48034.2019.8949593.

[39] M. A. Mazumder and R. A. Salam, “Feature extraction techniques for speech processing: A review,” *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 8, no. 1.3 S1, pp. 285–292, 2019, doi: 10.30534/ijatcse/2019/5481.32019.

[40] W. S. Mada Sanjaya, D. Anggraeni, and I. P. Santika, “Speech Recognition using Linear Predictive Coding (LPC) and Adaptive Neuro-Fuzzy (ANFIS) to Control 5 DoF Arm Robot,” *J. Phys. Conf. Ser.*, vol. 1090, no. 1, 2018, doi: 10.1088/1742-6596/1090/1/012046.

[41] K. H. Ghazali, M. F. Mansor, M. M. Mustafa, and A. Hussain, “Feature extraction technique using discrete wavelet transform for image classification,” *2007 5th Student Conf. Res. Dev. SCORED*, no. November 2015, 2007, doi: 10.1109/SCORED.2007.4451366.

[42] M. C. A. Korba, D. Messadeg, R. Djemili, and H. Bourouba, “Robust speech recognition using perceptual wavelet denoising and mel-frequency product spectrum cepstral coefficient features,” *Inform.*, vol. 32, no. 3, pp. 283–288, 2008.

[43] S. A. Majeed, H. Husain, and S. A. Samad, “Phase autocorrelation bark wavelet transform (PACWT) features for robust speech recognition,” *Arch. Acoust.*, vol. 40, no. 1, pp. 25–31, 2015, doi: 10.1515/aoa-2015-0004.

[44] J. Ferrer, “How Transformers Work: A Detailed Exploration of Transformer Architecture,” Datacamp. [Online]. Available: https://www.datacamp.com/tutorial/how-transformers-work

[45] Senbetu, “speaker-dependent speech recognition system for the Kambaatissa language using Hidden Markov Models (HMMs).”

[46] T. Gamta, *Seera afaan oromo*. 1995. [Online]. Available: https://books.google.com.et/books?id=QwyXZwEACAAJ

[47] Kothari, *Research Methodology - Methods and Techniques*. 2004.

[48] S. T. Abate, W. Menzel, and B. Tafila, “An Amharic speech corpus for large vocabulary continuous speech recognition An Amharic Speech Corpus for Large Vocabulary Continuous Speech Recognition,” no. September 2005, pp. 2–6, 2014, doi: 10.21437/Interspeech.2005-467.

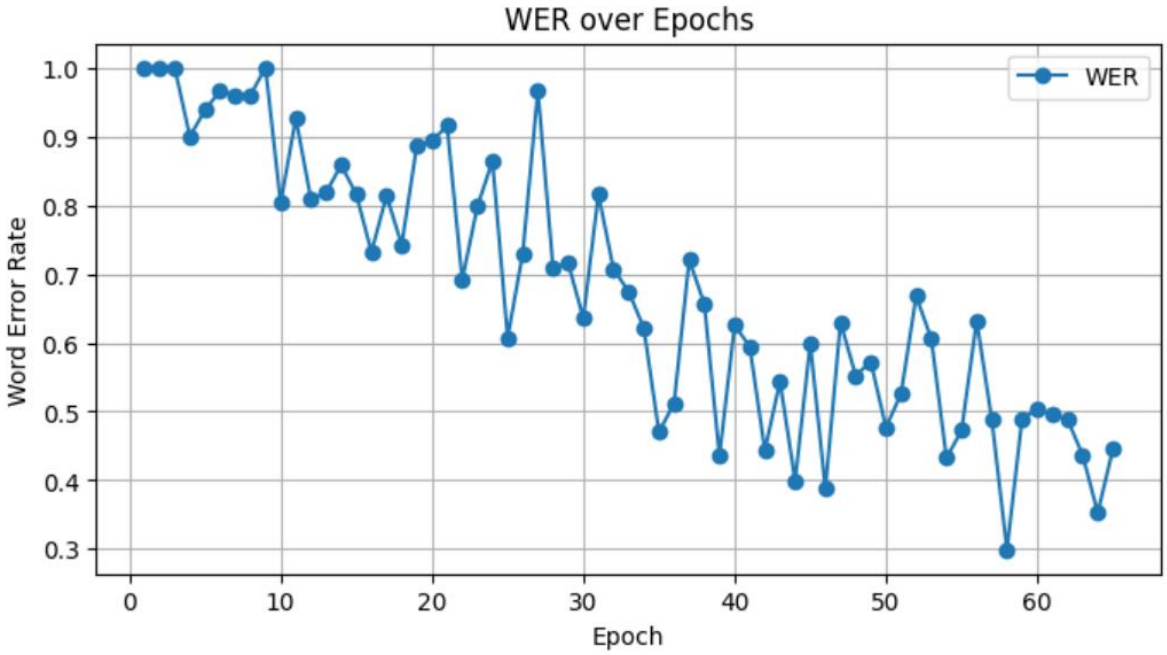
[49] P. B. and D. Weenink, “Praat: doing phonetics by computer,” *Univ. Amsterdam*, [Online]. Available: https://www.fon.hum.uva.nl/praat/

[50] L. Dong, S. Xu, and B. Xu, “SPEECH-TRANSFORMER : A NO-RECURRENCE SEQUENCE-TO-SEQUENCE MODEL FOR SPEECH RECOGNITION Institute of Automation , Chinese Academy of Sciences , China University of Chinese Academy of Sciences , China,” pp. 5884–5888, 2018.

[51] B. Pang, E. Nijkamp, and Y. N. Wu, “Deep Learning With TensorFlow: A Review,” *J. Educ. Behav. Stat.*, vol. 45, no. 2, pp. 227–248, 2020, doi: 10.3102/1076998619872761.

Appendices

Appendix 1: The Optimum model's WER diagram



Appendix 3: Sample of Training process

```
114/114 [==============================] - 459s 4s/step - loss: 0.8208 - val_loss: 0.8037
Epoch 2/100
114/114 [==============================] - 413s 4s/step - loss: 0.6406 - val_loss: 0.6596
Epoch 3/100
114/114 [==============================] - 409s 4s/step - loss: 0.5897 - val_loss: 0.6327
Epoch 4/100
114/114 [==============================] - 402s 4s/step - loss: 0.5785 - val_loss: 0.6253
Epoch 5/100
114/114 [==============================] - 412s 4s/step - loss: 0.5743 - val_loss: 0.6213
Epoch 6/100
114/114 [==============================] - ETA: 0s - loss: 0.5722
```

```
114/114 [==============================] - 430s 4s/step - loss: 0.5722 - val_loss: 0.6188
Epoch 7/100
114/114 [==============================] - 407s 4s/step - loss: 0.5703 - val_loss: 0.6194
Epoch 8/100
114/114 [==============================] - 407s 4s/step - loss: 0.5694 - val_loss: 0.6177
Epoch 9/100
114/114 [==============================] - 405s 4s/step - loss: 0.5681 - val_loss: 0.6187
Epoch 10/100
114/114 [==============================] - 398s 3s/step - loss: 0.5667 - val_loss: 0.6167
```

Appendix 3: Statistic of the dataset with which the model trained

```
print(f'Min Clip Duration (seconds): {min_clip_duration:.2f}')
print(f'Max Clip Duration (seconds): {max_clip_duration:.2f}')
print(f'Mean Words per Clip: {mean_words_per_clip:.2f}')
print(f'Distinct Words: {len(distinct_words)}')
print(f'Maximum Clip ID: {max_clip}, Duration: {max_clip_duration:.2f} seconds')
print(f'Minimum Clip ID: {min_clip}, Duration: {min_clip_duration:.2f} seconds')
```

```
Total Clips: 8729
Total Words: 127472
Total Characters: 984522
Total Duration (seconds): 64957.20
Mean Clip Duration (seconds): 7.44
Min Clip Duration (seconds): 1.52
Max Clip Duration (seconds): 14.95
Mean Words per Clip: 14.60
Distinct Words: 25831
Maximum Clip ID: LAS_03492, Duration: 14.95 seconds
Minimum Clip ID: LAS0_0257, Duration: 1.52 seconds
```

In [ ]:

```
1
```

Appendix 4: Some predictions on validation dataset

```
target:     <dhibeen koleeraa godina haragee lixaa aanaa odaa bultum mi'eessoo gammachiisii fi cirootti kan m
prediction: <dhibeen kuleen koleen kuleen kulate  ta'u  ta'aanaa rargeen kolee machiise yoo ta'u  kan multumm

target:     <jiilli itiyoophiyaa yaa'ii fayyaa idil addunyaa toorbaatamii lammaffaa irratti hirmaachaa jira.>
prediction: <jiilli itiyoophiyaa yaa'ii fayyaa idil addunyaa toorbaatamii lammaffaa erratti hirmaachaa jira.>

target:     <itti gaafatamaan waajjira eegumsa fayyaa godina qellam wallaggaa obbo darajjee abdannaa hawaasni
prediction: <itti gaafa tamaan waajjire yedamu dhawaa jireenlaa godina kunumsa fayyadamu dhaamaniiru.>

target:     <oromiyaan dhibee koleeraa ittisuurratti hojjechaa jira.>
prediction: <oromiyaan dhibee koleeraa ittisuurratti jechaa jira.>

target:     <ogeeyyiin fayyaa naannoo tigraay mariisifamaniiru.>
prediction: <ogeeyyin fayyaa naannoo tigraay mariisifamaniiru.>

100/100 [==============================] - 181s 2s/step - loss: 0.2607 - val_loss: 0.4756
Epoch 72/100
100/100 [==============================] - 160s 2s/step - loss: 0.2599 - val_loss: 0.4757
Epoch 73/100
100/100 [==============================] - 162s 2s/step - loss: 0.2593 - val_loss: 0.4776
Epoch 74/100
 48/100 [=============>................] - ETA: 1:13 - loss: 0.2766
```

```
target:     <dooktar amiir amaan akka beeksisanitti itiyoophiyaan yaa'icharratti yaadolii murtee lama ni dhiy
prediction: <dooktar amiir amaa akka beessanitti yoo remurtee  xila yaadolee murtii adulee murtee adule murte

target:     <fandiin daa'immanii dhaabbata mootummoota gamtoomanii 'unicef' deeggarsa yaalaa namoota dhibicha
prediction: <fandii yoona tokko  dhaabbata namoota dhibee yoo namoota dhibichaa fi yoo namoota garsa yeeyaa d

target:     <guyyaa har'aas beeksisni kana ifoomsu waajjira ministeera fayyaatti kaa'ameera.>
prediction: <guyyaa har'aas beeksisni kana ifoomsuwwaa jira fayyaatti kana ifoomsu waajjira meesisni kana ifo

target:     <ministirri ministeera fayyaa dooktar amiir amaan akka beeksisanitti mooraa fi naannawan ministee
prediction: <ministeera fayyaa  amiiriministeera federsii wanboolaa maanna wanta akka beeksisanitti hoospita'

target:     <yaalan qaqqabuuf carraaqqamus namoonni lama dhiibichaan lubbuu isaanii dhabuu biiroo fayyaa orom
prediction: <yaalun qaqqabuu fayyaa lunnadhii bichaan hooggansa afiizaanii dhabuu namuus hoomiyaatasaa fi qor

target:     <labsichi naannolee namoonni hedduun walitti qabamanitti tamboo xuuxuu ni dhoorka.>
prediction: <lafoon hedduun wanitti tamboo duungu namoon hedduun muraka.>

target:     <hawaasni nyaata madaalawwaa fi sogida ayoodiinii qabutti fayyadamu akka qabu ibsame.>
prediction: <hawaasni  nyaata madaalawwaa  fi soogidaa waa fi sogidamuu akka qabu ibsame.>
```

```
history = model.fit(ds, validation_data=val_ds, callbacks=[display_cb], epochs=50)
```

```
target:     <jiilli itiyoophiyaa yaa'ii fayyaa idil addunyaa toorbaatamii lammaffaa irratti hirmaachaa jira.>
prediction: <jiilli itiyoophiyaa yaa'ii fayyaa irratti hirmaachaa jira.>

target:     <itti gaafatamaan waajjira eegumsa fayyaa godina qellam wallaggaa obbo darajjee abdannaa hawaasni pirojekt
prediction: <itti gaafamaan wal'aa fayyadamuun seena kana akka fayyadamu dhalamu dhaamaniiru.>

target:     <oromiyaan dhibee koleeraa ittisuurratti hojjechaa jira.>
prediction: <oromiyaan dhibee koleeraa ittisoorratti surratti surratti surrattisuurratti surrattisuurrattisuurrattisuu

target:     <ogeeyyiin fayyaa naannoo tigraay mariisifamaniiru.>
prediction: <ogeeyyin fayyaa naannoo tigraay mariisifamaniiru.>
```