information

Article

Improving Amharic Speech Recognition System Using Connectionist Temporal Classification with Attention Model and Phoneme-Based Byte-Pair-Encodings

Eshete Derb Emiru [1,2], Shengwu Xiong [1,*], Yaxing Li [1], Awet Fesseha [1,3] and Moussa Diallo [1]

[1] School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, China; eshetede@whut.edu.cn (E.D.E.); yaxing.li@whut.edu.cn (Y.L.); awet.fesseha@mu.edu.et (A.F.); moussdiall@whut.edu.cn (M.D.)
[2] School of Computing, Debre Markos University, Debre Markos 269, Ethiopia
[3] School of Natural Science and Computing, Mekele University, Mek'ele 231, Ethiopia
* Correspondence: xiongsw@whut.edu.cn

Abstract: Out-of-vocabulary (OOV) words are the most challenging problem in automatic speech recognition (ASR), especially for morphologically rich languages. Most end-to-end speech recognition systems are performed at word and character levels of a language. Amharic is a poorly resourced but morphologically rich language. This paper proposes hybrid connectionist temporal classification with attention end-to-end architecture and a syllabification algorithm for Amharic automatic speech recognition system (AASR) using its phoneme-based subword units. This algorithm helps to insert the epithetic vowel ɨ[i], which is not included in our Grapheme-to-Phoneme (G2P) conversion algorithm developed using consonant–vowel (CV) representations of Amharic graphemes. The proposed end-to-end model was trained in various Amharic subwords, namely characters, phonemes, character-based subwords, and phoneme-based subwords generated by the byte-pair-encoding (BPE) segmentation algorithm. Experimental results showed that context-dependent phoneme-based subwords tend to result in more accurate speech recognition systems than the character-based, phoneme-based, and character-based subword counterparts. Further improvement was also obtained in proposed phoneme-based subwords with the syllabification algorithm and SpecAugment data augmentation technique. The word error rate (WER) reduction was 18.38% compared to character-based acoustic modeling with the word-based recurrent neural network language modeling (RNNLM) baseline. These phoneme-based subword models are also useful to improve machine and speech translation tasks.

Keywords: Amharic; automatic speech recognition; connectionist temporal classification with attention; natural language processing; low resource language; out-of-vocabulary

Citation: Emiru, E.D.; Xiong, S.; Li, Y.; Fesseha, A.; Diallo, M. Improving Amharic Speech Recognition System Using Connectionist Temporal Classification with Attention Model and Phoneme-Based Byte-Pair-Encodings. *Information* **2021**, *12*, 62. https://doi.org/10.3390/info12020062

Academic Editor: Yannis Korkontzelos
Received: 28 December 2020
Accepted: 31 January 2021
Published: 3 February 2021

1. Introduction

The use of conventional hidden Markov models (HMMs) and deep neural networks (DNNs) of automatic speech recognition (ASR) systems in the preparation of a lexicon, acoustic models, and language models results in complications [1]. These approaches also require linguistic resources, such as a pronunciation dictionary, tokenization, and phonetic context dependencies [2]. In contrast, end-to-end ASR has grown to be a popular alternative to simplify the conventional ASR model building process. End-to-end ASR methods depend on paired acoustic and language data, and train the acoustic model with a single end-to-end ASR algorithm [3]. As a result, the approach makes it feasible to construct ASR systems. The end-to-end ASR system directly transcribes an input sequence of acoustic features (F) to an output sequence of probabilities for tokens (p) such as phonemes and characters [4].

Various types of end-to-end architectures exist for ASR [5,6], such as connectionist temporal classification (CTC) [7], recurrent neural network (RNN) transducer [8], attention-based encoder-decoder [9], and their hybrid models [10,11]. The CTC method is used to train recurrent neural networks (RNNs) without knowledge of the prior alignment between input and output sequences of different lengths. The CTC model can also make a strong assumption between labels, and the attention-based model trains a decoder depending on the previous labels. End-to-end speech recognition systems are extensively used and studied for multiple tasks and languages, such as English, Mandarin, or Japanese. Considering a hybrid system is useful due to the advantage of the constrained CTC alignment in a hybrid CTC-attention end-to-end ASR system [6,12,13].

Most languages have insufficient data because obtaining speech data with their corresponding transcribed text is costly [14]. In contrast, automatic speech recognition (ASR) requires a large quantity of training data to perform recognition well [15]. In addition to low data resources, out-of-vocabulary words are also a significant challenge in automatic speech recognition systems. End-to-end ASR methods typically rely only on paired acoustic and language data. Without additional language data, they can suffer from data out-of-vocabulary (OOV) issues [16].

Our proposed CTC-attention end-to-end model is applied to the Amharic language, which is morphologically rich but poorly resourced [17]. It is the official working language of the Federal Democratic Republic of Ethiopia. This language is one of the Semitic languages [18] and one of the phonetic languages spoken in eastern Africa. It is also the second most widely spoken Semitic language in the world following Arabic [19]. Amharic scripts originate from the Ge'ez alphabet, which lacks capitalization [20]. These Amharic script graphemes are a combination of a consonant and a vowel. Hence, Amharic generally has 275 characters/graphemes composed of consonant–vowel (CV) syllables [21]; samples of Amharic graphemes are shown in Appendix A. The Amharic language mainly consists of seven vowels [18], namely ቨ[ə], ኡ[u], ኢ[i], ኣ[a], ኤ[e], እ[ɨ], ኦ[o]. This language has 32 consonants [18,22] that are categorized based on their articulation stops (14), fricatives (8), affricatives (3), nasals (3), liquids (2), and glides (2). These consonants are indicated in Appendix B with their corresponding International Phonetic Alphabet (IPA) representations.

Most Amharic speech recognition studies have been conducted using conventional HMM [22–24] and DNN [25] approaches. In the DNN approach, convolutional neural networks (CNNs) and RNNs with long-short term memory (LSTM) have been used [25]. A training speech corpus of 40.2 h collected by the Intelligence Advanced Research Projects Activity (IARPA) Babel project was used in both CNN and RNN techniques [25]. The OOV percentage was 11.7% and the minimum word error rate (WERs) registered were 42.1% and 42.0% using LSTM and CNN techniques, respectively.

Based on our reading, Amharic language speech recognition using the end-to-end system has not yet been reported. Among end-to-end ASR methods, considering a hybrid end-to-end ASR system is useful to our study due to the advantage of the constrained CTC alignment and attention mechanism trains based on the context of previous labels [6,12,13].

In this work, we focus on the OOV problem in speech recognition with different Amharic language units. CTC-attention end-to-end speech recognition is also proposed for modeling these units. Among language units, characters, phonemes, and subword units are directly used for acoustic modeling. Subword units are sequences of characters, phonemes, and phonemes with an epenthesis vowel inserted by a syllabification algorithm. These subwords are generated by a byte-pair-encoding (BPE) segmentation algorithm. Although a CTC-attention model with phoneme-based subword modeling is explicitly required for Amharic language, it has not yet been explored, and a syllabification algorithm could make a vital contribution to Amharic end-to-end speech recognition systems. The main contributions of this paper are summarized as follows:

1. In addition to Amharic reading speech of the ALFFA (African Languages in the Field: speech Fundamentals and Automation) dataset [22], we prepared additional speech

and text corpora, which cover various data sources. These data provide good coverage of the morphological behavior of Amharic language.

2. CTC-attention end-to-end ASR architecture with phoneme mapping algorithms is proposed to model subword level Amharic language units to resolve the problem of OOV words in Amharic automatic speech recognition (AASR).
3. We explored the effects of OOV words by considering the most frequently occurring words in different vocabulary sizes, namely, 6.5 k, 10 k, 15 k, and 20 k, in character-based and phoneme-based end-to-end models.
4. Evaluating and analyzing the speech recognition performance was performed using various meaningful Amharic language modeling units such as phoneme-recurrent neural network language modeling (RNNLM), character-RNNLM, and word-RNNLM. These language models help to explore the effects of context-dependent and independent RNNLMs in end-to-end speech recognition models.
5. The performance speech recognition results were compared and better results were found in phoneme-based subwords generated by the BPE segmentation algorithm. These phonemes include the Amharic epithetic vowel ɨ[ɨ] inserted by syllabification algorithms during preprocessing (phoneme mapping) of our dataset.

The remainder of the paper is organized as follows: Related studies of end-to-end ASR with various subword units are discussed in Section 2. A description of the dataset used in the current study and its preprocessing, and an overview of the proposed end-to-end speech recognition approaches, is provided in Section 3. Experiment parameter setups and results are noted in Section 4, and a discussion of the results is provided in Section 5. Conclusions and future work are presented in Section 6.

2. Related Work

To date, various studies of end-to-end speech recognition systems have been used in various languages and corpora. In this section, we review end-to-end ASR studies conducted based on character-based, phoneme-based, and subword-based models.

Inaguma et al. [26] used acoustic-to-word (A2W) and acoustic-to-character (A2C) end-to-end speech recognition systems for OOV detection. The A2C model was used to recover OOV words that are not covered by the A2W model through accurate detection of OOV words. To resolve OOV words, external RNNLM was developed in different standard vocabulary-sized Switchboard corpora (SBCs) and further improvement was achieved by recovering OOV words.

Boyer and Rouas [12] used CTC, location-based attention, and hybrid CTC-attention for character and sub-words generated by subword segmentation algorithms as acoustic modeling units. RNNLM was used in French language units, such as characters, subword units, and words of 50 k vocabulary size. Finally, minimum character error rates (CERs) and WERs were obtained in hybrid CTC-attention with character modeling units and subword modeling units, respectively.

Hori et al. [3] investigated an end-to-end speech recognition approach with different word-based RNNLM with vocabulary sizes, namely 20 K, 40 K, and 65 K of the LibriSpeech and Wall Street Journal (WSJ) corpora. Zeyer et al. [27] compared the grapheme-based and phoneme-based output labels via commonly used CTC and attention-based end-to-end models. Single phonemes and multiple phonemes without context and with BPE to obtain phoneme-based subwords were used, respectively. This experiment was conducted on a 300 h Switchboard corpus. The results showed that phoneme-based models and the grapheme-based model were competitive.

Wang et al. [28] used phoneme-based subwords found in byte-pair-encoding (BPE) for end-to-end speech recognition as modeling units. They used a pronunciation dictionary to convert transcriptions into phoneme sequences by maintaining the word boundaries and trained the hybrid CTC-attention acoustic model using phoneme BPE targets. Multi-level language model-based decoding algorithms were also developed based on a pronunciation dictionary. Experimental results show that phoneme-based BPEs tend to

yield more accurate recognition systems than the character-based counterpart on Switchboard corpora.

Xiao et al. [29] used a hybrid CTC-attention end-to-end ASR system to model subword units obtained by the byte-pair-encoding (BPE) compression algorithm. These subword models can model longer contexts and are better able to resolve the OOV problem on the LibriSpeech database than a character-based system. The subword-based CTC-attention showed a significant improvement of 12.8% WER relative reduction over the character-based hybrid CTC-attention system.

Yuan et al. [30] used a hybrid CTC-attention model for speech recognition purposes. A byte-pair-encoding (BPE) compression algorithm was also used for generating the subword units. Attention smoothing was used to acquire more context information during subword decoding. The subword-based models resolved the OOV problem on the LibriSpeech corpus.

Schuster and Nakajima [31] used closed dictionary and infinite dictionary models for Korean and Japanese voice searches with Google. They also addressed the challenges of scoring results in multiple script languages because of ambiguities due to the existence of many pronunciations per character, especially in the Japanese Unicode. Finally, the infinite vocabulary based on the word-piece segmenter resulted in a system with relatively low complexity to maintain and update.

Huang et al. [32] explored RNN-transducer, CTC, and attention-based end-to-end models. Word, character, and word-piece modeling units were used. All end-to-end experimental results showed that word-pieces achieved better results than words and characters. Label smoothing and data augmentation techniques were also used to improve the performance of the recognition on Switchboard/CallHome databases.

Das et al. [33] proposed a CTC end-to-end model combined with attention, self-attention, hybrid, and mixed-unit of word and letters to resolve the hard alignment and OOV problems of the word-based CTC model. These word-based CTC models only address frequently occurring words and the remaining words are tagged as OOV. These OOV problems are solved in a hybrid CTC by treating words as whole word units and OOVs are decomposed into a sequence of frequent words and multiple letters.

Zhang et al. [34] proposed a hybrid ASR system of CTC training and word-pieces. They simplified the conventional frame-based cross-entropy training using an engineering pipeline in addition to recognition accuracy. They also used word-piece modeling units to improve runtime efficiency because word-pieces were able to use a larger stride without losing accuracy.

In general, the above literature used word-pieces [32–34] or subwords to resolve OOV problems in each corpus, and most of these studies used CTC-attention ASR modeling. The subword units were words, characters, phonemes, and subwords generated by the BPE segmentation algorithm. The BPE segmentation algorithm was also used to obtain subwords of character and phoneme sequences as modeling units. The minimum word error rate was registered using phone-based subwords [27–30]. In some studies, various vocabulary sizes were also used to explore the OOV of words [3,26]. Studies on Amharic ASR systems were also conducted using conventional ASR systems [22,24,25] and with Amharic syllabification [23]. In the current research, we extended our Amharic ASR study in an end-to-end method with all subword units. In addition to subword unit modeling, we considered the epithetic vowel ɨ[ɨ], which is found in speech utterance but not in the transcribed Amharic text, during Amharic syllabification. Including the epithetic vowel ɨ[ɨ] in phoneme-based subwords has not yet been explored in CTC-attention end-to-end ASR modeling, thus making our study unique.

3. Dataset and Methods

3.1. Dataset and Data Pre-Processing

3.1.1. Dataset

We used the Amharic reading speech database collected for speech recognition purposes in the conventional ASR approaches [22], and an additional 2 h reading speech containing 999 sentences was used. These reading speech corpora were collected from different sources to maintain variety, such as political, economic, sport, and health news; the Bible; fictions; and Federal Negarit Gazeta and penal codes. Numbers and abbreviations were converted to their Amharic word representations. All contents of a corpus were set up using morphological concepts. Reading speech corpora of the corresponding texts were prepared using a 16 kHz sampling frequency, 16 bit sample size, and 256 kb bitrates with a mono channel. Its training speech was 22 h recorded by 104 speakers and a collection of 11,874 sentences. We used a text corpus prepared by [22], which consists of 120,262 sentences (2,348,150 tokens or 211,120 types). This text corpus was prepared to train the language model and to derive the vocabulary for the pronunciation dictionaries. The models' evaluations took place with a vocabulary size of 5 k, which contains 360 sentences arranged by [22].

Data augmentation is a method of increasing training data in poorly resourced languages [35]. It is used to address the scarcity of resources and to increase the performance of their ASR systems [36]. It is also one of the most effective means of making commutative end-to-end automatic speech recognition (ASR) with a conventional hybrid approach in low-resource tasks [37]. Data augmentation is a common strategy adopted to increase the quantity of training data. It is a key ingredient in state-of-the-art systems for speech recognition. Due to the widespread adoption of neural networks in speech recognition systems, large speech databases are required for training such a deep architecture, which is very useful for small data sets [36].

SpecAugment is a type of data augmentation in which data is augmented using three approaches, namely, time warping, frequency masking, and time masking [38]. In our proposed end-to-end ASR system, these three approaches were combined by increasing the size of FBank features [38] and delivered as inputs to the bi-directional long-short term memory (BLSTM) encoder. BLSTM can read inputs backward and forwards, which enables it to use future context to recognize speech more accurately

3.1.2. Text Corpus Pre-Processing

The grapheme-based text corpus was prepared as explained in Section 3.1.1. Amharic graphemes or characters are a combination of consonant and vowel phonemes [23]. These Amharic graphemes are directly used in character-based ASR, but text corpus pre-processing is required for phoneme-based speech recognition systems.

A phoneme-based text corpus was also prepared using the G2P conversion algorithm (Algorithm 1) and phoneme normalization list. Some Amharic graphemes have the same pronunciation representation, but the speech recognition task requires unique pronunciation. Phonemes that have the same pronunciation were used to normalize the text corpus at the phoneme-level. These phonemes are structured in the following groups: (ሀ, ሐ, ኀ, and ኸ) to ሀ, (አ and ዐ) to አ, (ሥ and ሰ) to ሰ, and (ፀ and ጸ) to ጸ. A phoneme normalization list was prepared and normalized based on their unique pronunciations.

In addition to the basic Amharic graphemes, labiovelar and labialized graphemes were also included in G2P conversion lists with their corresponding representation of phonemes. These graphemes were represented in two or three CV phonemes; for example, ቈ/qwa/ is a labialized grapheme represented as a combination of the basic phoneme ቀ/q/ with the rounded vowel ዉ አ/wa/. Ultimately, the normalized phoneme-based corpus was found through the normalization process, and used for phoneme-level language modeling during our experiment. The overall phoneme-based corpora preparations of our study are indicated in Figure 1.

Algorithm 1: Amharic Grapheme-to-Phoneme (G2P) Conversion algorithm

Input: Grapheme-based Amharic text corpus
Output: Phoneme-based Amharic text corpus
1: Index text file = 0; repeat
2: For each indexed file f in text corpora normalize the text corpus using unique phonemes
3: If grapheme is not unique phonemes
4: Replace each Grapheme G by its CV phoneme representation using G2P conversion list:
Grapheme G ← phoneme with CV phoneme pattern
5: Else
6: Keep its phoneme representation
7: End if
8: End for
9: index ++;
10: until the process is applied to all files

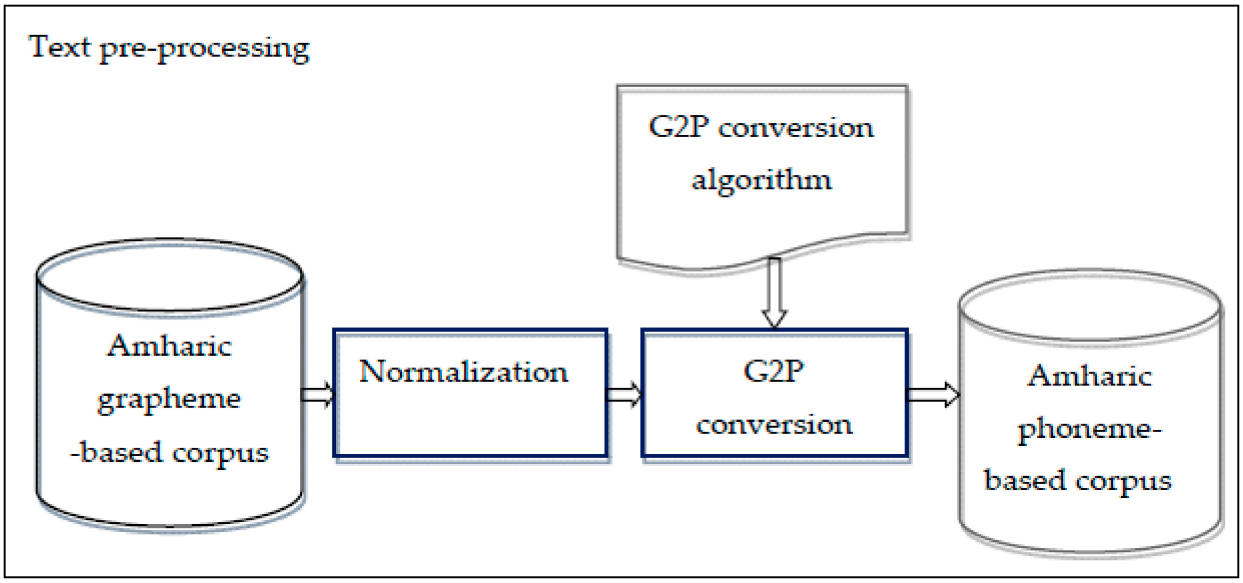

Figure 1. The overall text pre-processing using a phoneme-mapping algorithm.

Extending to the G2P conversion algorithm, the Amharic syllabication algorithm considered for epenthetic vowel ɨ[i] insertion and overall text pre-processing with the syllabification algorithm is presented in Figure 2. The two significant challenges of G2P conversion are epenthesis and gemination due to the failure of Amharic orthography to show epenthetic vowels and geminated consonants. Amharic is a syllabic language in which graphemes are represented by consonant–vowel (CV) combinations [23]. However, all Amharic graphemes are not represented in CV sequences. In Amharic, all graphemes (231 in total) and seven labiodental graphemes are represented in CV syllables, but twenty Labiovelar and eighteen labialized graphemes are represented as either two or three combinations of CV syllables. Amharic has different syllable patterns, such as V, VC, CV, and CVC, and these are considered syllabification rules. These templates/rules embrace gemination and consonant clusters.

A syllabification algorithm developed for the Amharic language was proposed in [39] and all syllabification algorithm templates or rules were included for epenthetic vowel insertion. Rules of Amharic syllables are V, CV, VC, CVC, VCC, and CVCC [40]. This syllabification used principles of sonority hierarchy and maximum onset to develop a rule-based syllabification algorithm. This algorithm considers the epenthesis vowel ɨ[i] and gemination of the Amharic language. The summary of observed sequences of consonants

with their corresponding epenthesis is indicated in the following six epenthetic vowel ɨ[i] insertion rules:

1. # CC #CiC
2. CCC CCiC
3. C_1C_1C (CC:) C_1C_1iC (C:iC)
4. CC_1C_1(CC:) CiC_1C_1 (CiC:)
5. $C_1C_1C_2C_2$ (C:C:) $C_1C_1iC_2C_2$ (C:iC:)
6. CC# CiC#

where # indicates the final position, otherwise medial or initial positions, and subscripts 1 and 2 indicate the germination sequences.

The syllabification algorithm was implemented in our study using these epenthetic vowel insertion rules. An epenthetic vowel insertion algorithm or procedure was adopted from [39] with minor modification and improvement. Finally, a phoneme-based text corpus with an epenthetic vowel was prepared using the G2P algorithm for our further experiment.

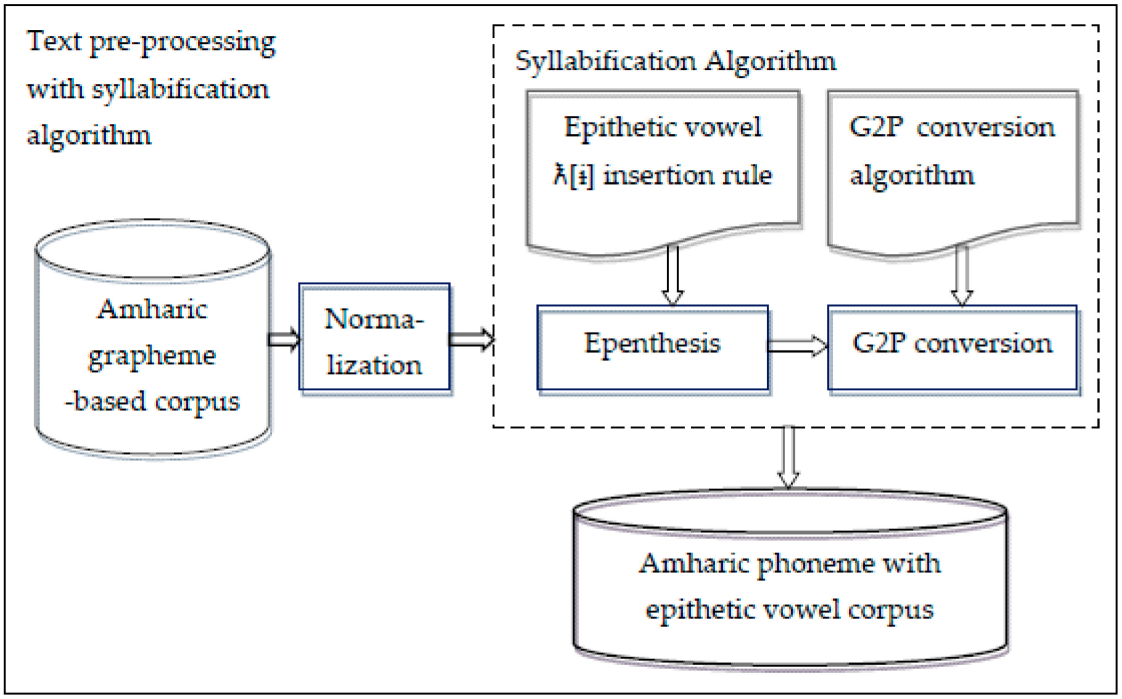

Figure 2. Overall text pre-processing using a syllabification algorithm.

3.2. Methods

This section briefly overviews the CTC, attention, and hybrid CTC-attention end-to-end ASR models [10,41]. Because these models are proposed for training/modeling, decoding, and evaluating [42] Amharic language units, an explicit overview of their functionalities is required.

3.2.1. CTC Model

The CTC model is used to map speech input frames into corresponding output labels [43], which is used to optimize the prediction of a transcription sequence. When the length of the output labels is shorter than the length of the input speech frames, a CTC path is introduced to have an identical length as that of the input speech frames by adding a "blank" symbol as an additional label and allowing repetition of labels to map the label sequence into the CTC path. This forces the output and input sequences to have identical lengths [44].

Let F be the input feature vectors of a given acoustic input sequence and $F = F_1, F_2, \ldots, F_T$ in the given time of T. In the end-to-end speech recognition system, Pr $(P|F)$ = Pr $(P_1|F)$, ... , Pr $(P_L|F)$ are the output labels of posterior probability sequence vectors. The posterior sequence length's output labels are represented as Pr $(P_i|F)$ at the given position in [45]. Pr$(P_i|F)$ is the posterior probability vector dimension in the N number of target labels. In our paper, we used phonemes, characters, and subwords as output labels which can be generated from speech waveforms. In end-to-end ASR, the length (L) of the output labels is shorter than that of the input speech frames (T), and is the most typical challenge of speech recognition systems. For this typical purpose, a special blank was introduced during CTC training and inserted between two consecutive labels. This label also allows for the repetition of labels. P is a label sequence that is expanded to $\Psi(p)$, with the same input sequence length. The label sequence posterior probability (p) is calculated and the sums of posterior probabilities are the possible paths $\Psi(p)$. There are limitations of input sequences in which each label's posterior probabilities in the output sequence are independent of each other. Based on these concepts, the CTC loss can be calculated as:

$$\Pr(P/F) = \sum_{\pi \in \Psi(p\ )} \Pr(\pi/F) = \sum_{\pi \in \Psi(\mathrm{p}\ )} \prod_{i=1}^{T} \Pr(\pi_i/F) \quad (1)$$

where Pr $(\pi_i|F)$ represents posterior probabilities calculated with a multi-layer bi-direction-linebreak al RNN. CTC loss is calculated efficiently with the forward-backward algorithm by configuring its gradient network parameters. In CTC-based end-to-end models, each label's probability is independent because their relationship is not learned explicitly during training. The loss function can be decomposed further using a conditional independence assumption, using the product of the posteriors of each frame, as [33]:

$$\Pr(p/F) = \prod_{i=1}^{T} (\pi_i/F) \quad (2)$$

The shared BLSTM encoder networks are used in CTC model architecture. The monotonic alignment of speech and label sequences is forced by the forward-backward algorithm of the CTC model. This forward-backward algorithm helps to speed up the alignment of language units.

During decoding, it is straightforward to generate the decoded sequence using greedy decoding by simply concatenating the labels corresponding to the highest posteriors and merging the duplicate labels, and then removing the "blank' labels. Thus, there is neither a language model nor any complex graph search in greedy decoding.

3.2.2. Attention-Based Model

The attention-based end-to-end ASR approach is the other method of mapping speech utterances into their corresponding label sequences [46–48]. This approach has encoder-decoder subnetworks, such as in Listen, Attend, and Spell (LAS), where a neural network learns to transcribe speech utterances to characters [49]. The encoder transforms the acoustic feature sequences of a speech to the length (T) of the sequence representation. The decoder transcribes high-level features (H) generated by the shared encoder into a p output label sequence with the attention-based model. Based on the conditional probability of the label p_u given the input feature H and the previous labels $p_{1:u-1}$, the decoder calculates the likelihood of the label sequence using the chain rule [8,50]:

$$Pr(p/F) = \prod_{u} Pr(p_u/H, p_{1:u-1}) \quad (3)$$

In every step u, the decoder generates a context vector c_u based on all input features H and attention weight $b_{u\sum l}$:

$$c_u = \sum_{l} b_{u,l} H_l \quad (4)$$

Attention mechanisms can be divided into various types. Among these attention types, location-aware attention has a good score record in [12,45]. The attention weight $b_u = (b_{u,1}, b_{u,2}, \ldots, b_{u,1})$ is obtained from location-based attention energies $e_{u,1}$ as follows:

$$b_{u,l} = softmax\ (e_{u,l}) \tag{5}$$

$$e_{u,l} = \omega^T tanh(Wq_{u-1} + V\ h_l + Mf_{u,l} + b) \tag{6}$$

$$f_u = F * b_{u-1} \tag{7}$$

where ω, V, W, b, M are trainable parameters, q_{u-1} is the state of the RNN decoder. * denotes the one-dimensional convolution along the frame axis, l, with the convolution parameter, F, to produce the features $f_u = (f_{u,1}, f_{u,2}, \ldots, f_{u,l})$. We can predict the RNN hidden state q_u and the next output p_u with the context vector c_u in Equations (8) and (9), respectively.

$$q_u = LSTM(q_{u-1}, p_{u-1}, c_u) \tag{8}$$

$$p_u = FullyConnected(q_u, c_u) \tag{9}$$

where the LSTM function here is implemented as a unidirectional LSTM layer and the fully connected function indicates a feed-forward fully-connected network. In attention-based end-to-end speech recognition, special symbols sos and eos are added to the decoder module, and denote start-of-sequence and end-of-sequence, respectively. The decoder stops the generation of new output labels when eos is emitted.

3.2.3. CTC-Attention Model

The CTC model contains conditional independent assumptions between labels and the attention mechanism yields an output by a weighted sum of all inputs without the guidance provided by alignments. The CTC model can learn a monotonic alignment between acoustic features and sequence of labels using its forward-backward algorithm, which helps the encoder to converge more quickly. An attention-based decoder has also helped to learn dependencies among targeted sequences [29]. Hence, the CTC-attention model has advantages of both CTC and attention-based models, and was used for our study. The overall architecture of the CTC-attention model is presented in Figure 3.

In the CTC-attention model, the advantages of both CTC and attention-based models are utilized, namely the better alignment of input-output sequences and consideration of the context prior sequences, respectively. It combines both the CTC loss and the cross-entropy loss of the attention modeling mechanism calculated between the predicted label and targeted correct label sequences [10]. Assuming $F = (F_0, F_1, F_2, F_3, F_4, F_5, F_6, F_7, \ldots, F_T)$ is the input sequence of the acoustic feature and $p = (p_1, p_2, \ldots, p_u)$ is the corresponding sequence of the output symbol, the transcription between F and p is modeled by the CTC-attention end-to-end ASR approach.

Assume that $P_u \in \{1, \ldots, N\}$, where represents the number of different label units. In end-to-end speech recognition approaches, the feature sequence (i.e., u < T) is longer than the sequence length of the output label. The CTC-attention model uses a shared recurrent neural network (usually LSTM) encoder to produce a high-level representation hidden layer $H = (H_0, H_1, H_2, H_3, \ldots, H_L)$ of the input sequence F, and L represents the index of a downsampled frame.

$$H = Encoder(F) \tag{10}$$

In the CTC-attention end-to-end model, the objective function is used to train a locatio n-based attention model [6]. In the CTC-attention architecture overall diagram, the shared BLSTM with CTC and attention encoder networks are used. The monotonic alignment of speech and label sequences is forced by the attention model and forward-backward algorithm of the CTC model rather than the attention model only. This forward-backward algorithm helps to speed up the language units' alignment compared to solely using data-driven attention methods. Because this CTC-attention model contains both CTC and at-

tention models, it is also known as multi-task learning (MTL). The CTC-attention model equation with λ CTC-weight is presented in Equation (11) and its logarithmic linear combinations are presented in Equation (12).

$$L_{MTL} = L_{CTC} + (1 - \lambda)L_{att} \tag{11}$$

$$L_{MTL} = \lambda \log \Pr_{CTC} + (1 - \lambda) \log \Pr_{att}(p/F) \tag{12}$$

where λ values: $0 \leq \lambda \leq 1$ and att represents attention.

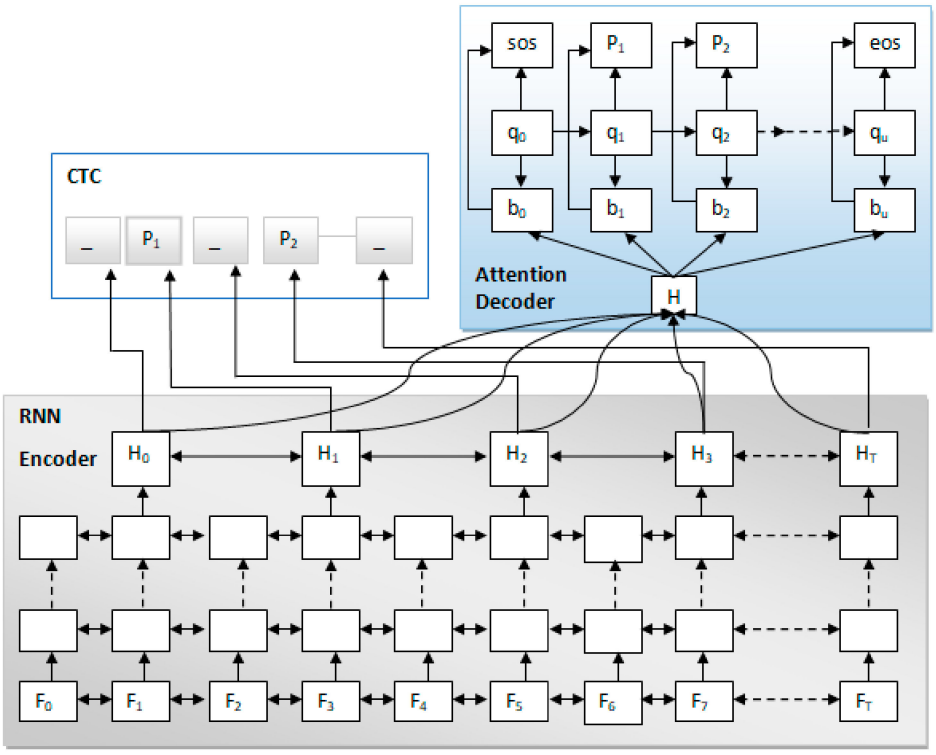

Figure 3. Connectionist temporal classification (CTC)-attention based end-to-end model.

This CTC-attention model is used with RNNLM to decode various language units, such as words, characters, phonemes, and subwords generated by the BPE segmentation algorithm. The CTC-attention model is used to address the OOV word issue because in CTC modeling only frequent words are used as target units. With the exception of these frequent words, other words are tagged as OOV words. Those OOV words cannot be included during network modeling, implying that they are excluded during the evaluation of a speech recognition system.

3.3. Our Proposed Speech Recognition System

Feature extraction techniques, acoustic modeling, and language modeling are the main components of our proposed end-to-end automatic speech recognition system. Feature extraction takes place at the front-end of the speech recognition system. The SpecAugment data augmentation technique is used during our feature extraction to increase the training data and to overcome overfitting problems due to the poorly resourced Amharic dataset. The feature extraction process takes place as explained in Section 3.1.1.

Acoustic modeling and language modeling are back-end processes of the speech recognition system. Acoustic modeling is also the main component of the ASR system and one of the methods outlined in Section 3.2. Among those models, the CTC-attention model was selected, and combines the advantages of CTC and attention mechanisms, as indicated in Section 3.2.3.

Language modeling is the most vital component of our proposed system because it considers epithetic vowels, which are not found in transcribed Amharic text but exist in speech utterances. Due to the inclusion of these epithetic vowels, the phoneme-level of our text data pre-processing is unique. The syllabification algorithm is used in addition to our G2P conversion algorithm, as indicated in Section 3.1.2. Finally, the phoneme-based BPE segmentation algorithm is applied for our subword-based end-to-end model, and these are used as the language units of RNN language modeling. The outputs of our proposed system are useful for machine and speech translation tasks. The overall architecture of the new proposed end-to-end Amharic speech recognition is presented in Figure 4.

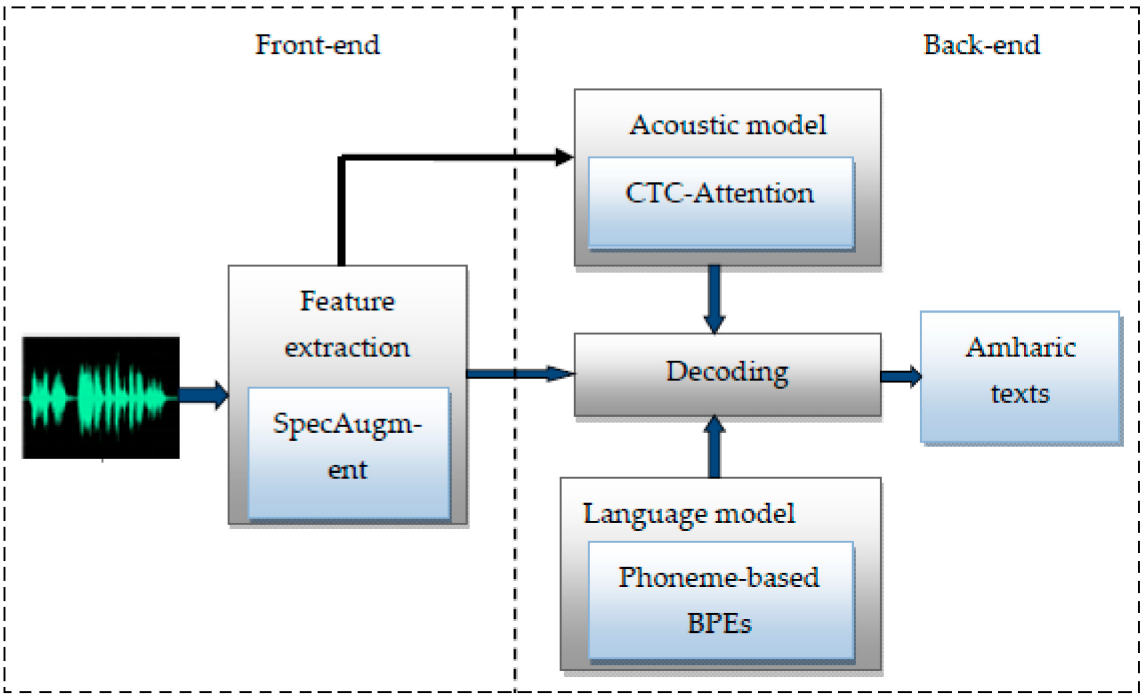

Figure 4. The architecture of the new proposed end-to-end Amharic automatic speech recognition (ASR) system.

4. Experiment Parameter Setups and Results

4.1. Parameter Setups and Configuration

In our experiments, a computer with a GPU (GeForce RTX 2080) and 64 GB memory were used to perform the speech recognition system's training and testing phases. The Espnet toolkit with Pytorch backend was used for both language and acoustic modeling, and the Kaldi tool was used for data preparation. Acoustic features were extracted from utterances of datasets for training and testing of end-to-end ASR systems of our Amharic speech corpus. FBANK+pitch features were extracted from the speech utterances. Input features of all models were 80-dimensional feature vectors, and their delta and acceleration coefficients were used as acoustic features [51]. On these features, non-overlapping frame stacking was applied and new superframes were made by stacking and skipping three frames. Utterance level mean normalization was also applied to the features.

The training was performed over 20 epochs using Pytorch modeling with a batch size of 30. During training, maxlen-in and maxlen-out were set at 800 and 150, respectively. In our experiment, the location-based attention mechanism, which was found to outperform in [12,45], was used in the hybrid CTC-attention architecture. The encoder consists of four BLSTM shared layers, with 320 cells used in our model. Its convolutional features were extracted by 10 centered convolution filters with 100 widths. The 0.2 dropout rate was used during the training of each BLSTM layer [9]. The sub-sampling was set as "1_2_2_1_1". The decoder was also set with 320 cells of one layer LSTM, 320 tanh nodes of a hidden

layer, 300 decoding units, and a softmax output layer. The type of end-to-end model was determined by CTC-weight (λ) and was described in terms of multi-task learning (mtlalpha). The value of mtlalpha was set as 0.5 for the hybrid CTC-attention end-to-end model.

In our language modeling, settings were made depending on the language units used. For word LM, a 1-layer RNN architecture with 1000 units, sgd optimization, 300 batch size and 40 maximum length were set. Unlike word LM, a two-layer RNN architecture with 650 units in each layer, Adam optimization, 1024 batch size, and 650 maximum lengths were set for both character and phoneme LMs. For all LMs, the number of epochs and patience were set as 20 and 3, respectively.

The AdaDelta algorithm was used with 0.1 language smoothing and an Adam method with standard settings was used for optimizing networks. Label smoothing was used as a regularization mechanism to protect the model from making overconfident predictions. This helps the model to have higher entropy during its forecast, and allows the model to be more adaptable. The ground truth label distribution was smoothed with a uniform distribution over all labels.

In all of our end-to-end models, the minibatch size was set to 30. The network parameters were also initialized with random values drawn from a uniform distribution with a range of $(-0.1, 0.1)$. Providing long input sequences can slow convergence at the beginning of training. Therefore, input data were sorted by the length of frames before creating mini-batches. In all end-to-end CTC, attention mechanism, and hybrid ASR models, the decoding process was performed in a beam size of 20 and language weight (lm-weight) of 1.0. The BPE segmentation algorithm was used to extract subword units from all training data and the size of subword units was set to 500.

4.2. Experiment Results

We trained our end-to-end CTC-attention ASR system using various vocabulary sizes. These vocabulary sizes were selected based on frequently occurring Amharic words. We also considered the OOV rates of less than 10% during vocabulary size selection. The vocabulary sizes were arranged in five intervals to determine the effects of OOV words on our small-sized text corpus. The SpecAugment data augmentation technique was also used to resolve the problem of smaller training data size and to make our system more robust. The models were evaluated with our test data without an OOV rate.

After the training and decoding process, the experiment results are discussed below for three main categories, namely, character-based acoustic modeling with word-RNNLM and character-RNNLM, phoneme-based acoustic modeling with word-RNNLM and phoneme-RNNLM, and subword-based language units generated by BPE segmentation algorithm in characters, phonemes, and phonemes with epithetic vowel acoustic modeling. The hybrid of the CTC and attention-based model takes advantage of the two models in combination. The CTC model in conjunction with a location-based attention decoder also helps the network to accelerate the training [30]. The discussions of each training and their decoding results are presented as character-based, phoneme-based, and subword-based end-to-end ASR models. The results of each end-to-end model are discussed individually and character-based results are considered as the baseline of our study.

4.2.1. Character-Based Baseline End-to-End Models

In our character-based experiment, word-based RNNLMs were used to investigate the recognition performance in various vocabulary sizes. The training processes took place in various vocabulary sizes, such as 6.5 k, 10 k, 15 k, and 20 k, and decoded with word-level RNNLM. These vocabulary sizes are parts of the training text data as used in Wall Street Journal (WSJ) and Switchboard (SWBD) corpora in [28]. Word sequences are easily generated in a word-based model by picking their corresponding posterior spikes. Limiting Amharic vocabularies is a challenging task due to its morphological behavior [52]. Furthermore, determining their corresponding size of OOV rate percentage was also a challenge. All OOV rates generated with their corresponding vocabulary sizes were evaluated in a 5 k

evaluation test size. Continuous CER and WER reductions were obtained from small to large vocabularies, but their corresponding OOV rates were reduced automatically. The results were evaluated using the CTC-attention end-to-end method in terms of character error rate (CER) and word error rate (WER) matrices. The minimum results were registered at 20 K vocabulary size and the overall results with different vocabulary sizes are indicated in Table 1.

Table 1. Character-based end-to-end model results in different vocabulary sizes.

Language Unit	Vocabulary Size	LM	Acoustic Model	CER (%)	WER (%)
character	6.5 k	Word-RNNLM	CTC-attention	28.09	39.30
character	10 k	Word-RNNLM	CTC-attention	26.91	37.60
character	15 k	Word-RNNLM	CTC-attention	25.60	37.01
character	20 k	Word-RNNLM	CTC-attention	25.21	36.80

Our experiment extended character-based acoustic modeling with its corresponding character-RNNLM. The experiment was also continued with and without the SpecAugment data augmentation technique. The SpecAugment augmentation technique with a combination of 5 maximum time warping, 30 frequency mask, and 40 time mask was used, and reduction of CER was achieved as indicated in Table 2.

Table 2. Character-based end-to-end model results with character recurrent neural network language modeling (RNNLM).

Language Unit	LM	Acoustic Model	CER (%)	WER (%)
characters	Character-RNNLM	CTC-attention	24.90	44.02
characters	Character-RNNLM	CTC-attention + SpecAugment	23.80	41.00

Data augmentation techniques such as SpecAugment help to make poorly resourced languages be competitive in end-to-end methods [37] by increasing the size of training data. Characters are the language units used during character-based language modeling for character-based end-to-end ASR. A total of 233 characters were used during this ASR after the normalization process [53].

The minimum CER and WER achieved were 23.80% and 41.00%, respectively, in CTC-attention with SpecAugment and character-RNNLM. In comparison to the above word-RNNLM results, the WER increased. Unlike WER, CER was reduced due to its context-independent character-level LM [54]. The results suggest continuation of our experiment using other subword units, such as phonemes and subword units generated by segmentation algorithms that consider their contexts.

4.2.2. Phoneme-Based End-to-End Models

Phoneme-based end-to-end ASR models are vital to improve the speech recognition system by addressing the variations of graphemes for similar pronunciation representations. In [55], an aligner with a pronunciation dictionary used what was called the Pronunciation Assisted Subword Modeling (PASM) method. This method adopts fast alignment to align with the pronunciation lexicon file and the result was also used to determine the common correspondence between subword units and phonetic units. In our phoneme-based speech recognition, all 39 Amharic phonemes were used as a language unit and phoneme level language modeling was also developed. A grapheme-to-phoneme (G2P) conversion algorithm was applied to the prepared grapheme-based text corpus to generate a phoneme-based text corpus. A sample of text conversion with three sentences is presented as follow:

Grapheme text:

1. እውቅና ን ማግኘቴ ለ እኔ ትልቅ ክብር ነው
2. ምን ለማ ለት ነው ግልጽ አድርገው
3. ከዚያ በ ተጨማሪ የ ስልጠና ውን ሂደት የሚ ያሻሽል ላቸው ይሻሉ

Their corresponding converted phoneme texts:

1. እውቅንኣ ን ምኣግኘኧትኤ ልኧ እንኤ ትልቅ ክብር ንኧው
2. ም ን ልኧምኣ ልኧት ንኧው ግልጽ ኣድርግኧው
3. ክኧዝኢይኣ ብኧ ትኧጭኧምኣርኢ ይኧ ስልጥኧንኣ ውን ሀኢድኧት ይኧምኢ ይኣሸኣሸል ልኣችኧው ይ ሸኣልኡ

Our phoneme-based experiment is an extension of the above character-based experiment. Phoneme-based end-to-end modeling with word-RNNLM in an experiment with four vocabulary sizes was conducted in our study. All results showed a reduction in word error rates; the overall phoneme error rate (PER) and word error rate (WER) results are presented in Table 3. In our phoneme-based CTC-attention end-to-end ASR model, WER reduction of 13.30% was achieved in 20 k vocabulary size compared to the above speech recognition results of the character-based with word-RNNLM baseline.

Table 3. Phoneme-based end-to-end model results in different vocabulary sizes.

Language Unit	Vocabulary Size	LM	Acoustic Model	PER (%)	WER (%)
phoneme	6.5 k	Word-RNNLM	CTC-attention	18.68	26.13
phoneme	10 k	Word-RNNLM	CTC-attention	17.36	24.26
phoneme	15 k	Word-RNNLM	CTC-attention	16.8	24.29
phoneme	20 k	Word-RNNLM	CTC-attention	16.1	23.50

Our phoneme-based experiment was extended with its corresponding phoneme-based RNNLM. This experiment was also continued with and without the SpecAugment data augmentation technique. This experiment was a continuation of the second experiment, and minimum PER (14.60%) and WER (34.01%) were achieved in CTC-attention with Spec-Augment, as indicated in Table 4. This result showed a 9.2% word error rate reduction compared to character-based with character-RNNLM. Finally, our phoneme-based CTC-attention with SpecAugment CER (10.61%) and WER (2.79%) error rate reductions were achieved compared to the above baseline character-based with word-RNNLM speech recognition results. The WER was unlike PER due to its context-independent phoneme-level LM [54]. The results again suggest continuing our experiment in context-dependent subword-based modeling generated by the BPE segmentation algorithm.

Table 4. Phoneme-based end-to-end modeling with phoneme-RNNLM decoding results.

Language Unit	LM	Acoustic Model	PER (%)	WER (%)
Phoneme	Phoneme-RNNLM	CTC-attention	15.80	36.20
Phoneme	Phoneme-RNNLM	CTC-attention + SpecAugment	14.60	34.01

4.2.3. Subword-Based End-to-End Models

In our subword-based end-to-end modeling, three different Amharic subwords were considered, namely, character-based subwords, phoneme-based subwords, and phoneme-based subwords with epithetic vowels inserted by a syllabification algorithm. These subwords were obtained by the byte-pair-encoding (BPE) segmentation algorithm based on the most frequent pairs of units [56]. Like our previous experiments, a hybrid CTC-attention-based end-to-end speech recognition system that works without any dictionary or lexicon was used. Subword units were used as language modeling units. Compared to the character-based and phoneme-based systems, the proposed subword-based system

significantly reduces both the CER and WER. The overall results of all subword modeling are presented in Table 5.

Table 5. Character-based and phoneme-based subword decoding results.

Language Unit	Subword Unit	Acoustic Model	C/PER (%)	WER (%)
Subword	character	CTC-attention	21.60	34.70
		CTC-attention + SpecAugment	16.90	31.30
Phoneme-based subword	phoneme	CTC-attention	15.80	22.60
		CTC-attention + SpecAugment	14.60	21.40
Proposed phoneme-based with epenthesis subword	Phoneme	CTC-attention	12.61	20.30
		CTC-attention + SpecAugment	12.80	18.42

Subword units of size 500 were extracted from training data. The subword-based CTC-attention acoustic modeling with SpecAugment system results were found to be 31.30%, 21.40%, and 18.42% WER using characters, phonemes, and phonemes, respectively, with epenthesis vowel subword sequences. The data augmentation technique also showed a slight improvement in all subword level speech recognition systems. The minimum PER obtained was 16.90%, 14.60%, and 12.61% in SpecAugment and subword sequences of characters, phonemes, and phonemes with epenthetic vowels, respectively. The sparseness problem was also evident during phoneme-based subword units and PER increased slightly. This problem was addressed by removing some phonemes during training and using the data augmentation technique. As a result, PER improved and the performance was almost equal to that of the previous result.

Finally, we compared the word error rate results concerning our objective, namely, reducing out-of-vocabulary (OOV) words. Better results were obtained in subword units with the BPE algorithm in our CTC-attention end-to-end speech recognition system. Our final phoneme-based models performed better than our final character-based models. Out-of-vocabulary (OOV) words were reduced using subwords as a decoding unit [57].

We also compared the WER results of character- and phoneme-based ASR systems without context versus character-based subwords and phoneme-based subwords. We found that subword units were best able to reduce WERs both in phoneme and character levels [27]. Character-based subword models have been used in previous research, however, phoneme-based subwords, particularly using a syllabification algorithm for Amharic, represent a new model of speech recognition. These subwords are used to achieve a minimum word error rate by reducing OOV words while maintaining their simplicity in end-to-end methods.

5. Discussion

Our proposed CTC-attention end-to-end AASR was evaluated using characters, phonemes, character-based subwords, and phoneme-based subwords. These proposed models are discussed using OOV words in terms of WER obtained in word-RNNLM, character-RNNLM, phoneme-RNNLM, and subwords obtained in the BPE segmentation algorithm.

We compared the WER of character-based and phoneme-based end-to-end ASR models in different vocabularies obtained using the most frequently occurring words. Their corresponding word-RNNLM was also prepared for decoding purposes. The results showed that the performance of the phoneme-based CTC-attention method was significantly better than the character-based performance because the former supports pronunciation-based labels. These pronunciation-based dictionaries are found using the G2P conversion algorithm because pronunciation dictionaries are used directly, like in conventional ASR approaches.

Due to the Amharic language's morphological richness, the OOV problem is evident in both character-based and phoneme-based word-RNNLM. Comparison of their WER with corresponding OOV rate is shown in Figure 5. The result shows that the WER decreased when the OOV rate was reduced. The phoneme-based WER was significantly less than the character-based WER because phonemes are assisted by pronunciation [55]. This result suggests our experiment should be continued with OOV reduction techniques.

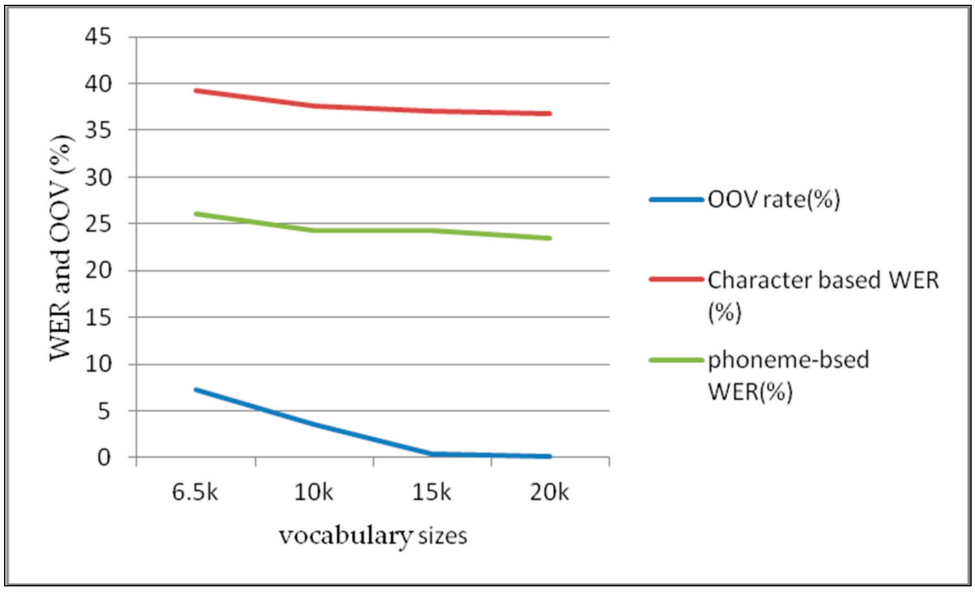

Figure 5. The word error rate (WER) and out-of-vocabulary (OOV) percentages in both CTC and CTC-attention.

To reduce the OOV words, the experiment continued with character-RNNLM and phoneme-RNNLM. These language models do not consider their context like word-RNNLM. The result showed the worst WER due to context-independent characters and phonemes [54], but minimum CER and PER were registered. This indicates that the OOV word problem was not resolved, and further experiments are required at the subword level.

Subword-based models have shown excellent results for machine translation (MT) [58]. The BPE segmentation algorithm is used in these models, and phoneme BPEs have been compared in terms of contiguous characters and phonemes [28]. Better results are obtained in subword units with BPE in our CTC-attention end-to-end speech recognition system. Our final phoneme-based models, which consider the epithetic vowels, perform better than our final character-based models, including the phoneme-based BPEs. Out-of-vocabulary (OOV) words were reduced using subwords as a decoding unit as per our proposal [57]. In general, better results are found in phoneme-based models compared to character-based models, and phoneme-based subword unit results are also better than those of character-based subword (BPE) units [27].

We compared the character-based and phoneme-based WERs using different vocabularies, which were based on the most frequently occurring words. These results showed that the performance of the phoneme-based CTC-attention method was significantly better than that of the character-based method because the latter is supported by pronunciation-based labels. These pronunciation-based dictionaries are found using the G2P conversion algorithm because pronunciation dictionaries are used directly, like om non-end-to-end approaches.

In addition to the CER and PER of our proposed CTC-attention method, CTC-attention with SpecAugment is helpful in accelerating the convergence during training; its training and validation losses are indicated in Figure 6. It can be observed that its losses became more robust and consistent as the number of epoch size increased up to 20.

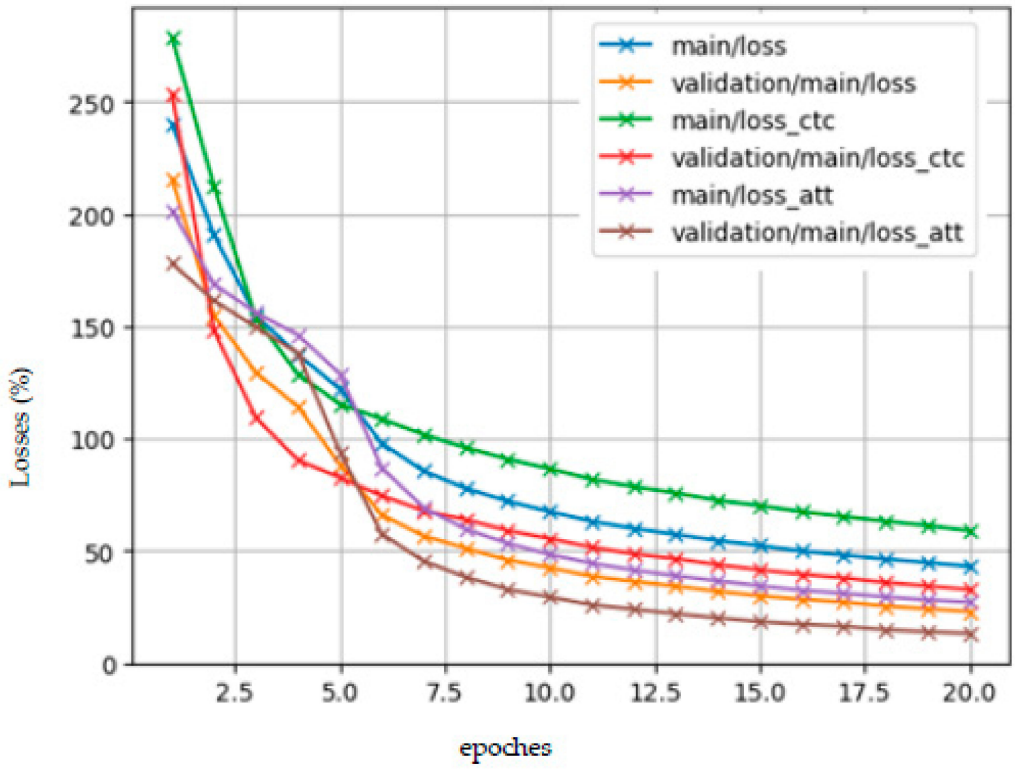

Figure 6. Phoneme-based CTC-attention with SpecAugment loss functions.

We also observed that the input-output alignment was appropriately learned. The input-output alignment sequences are shown from the beginning with almost a spectrogram representation of the utterance. When the training extends in different epochs, we observed the gap of alignments. This gap indicates that there were missing phonemes that can be analyzed in terms of deletion, insertion, and substitution during training [59]. The final training result indicates that the alignment became monotonic [9]. The monotonic sample alignment for utterance "የተለያዩ የትግራይ አውራጃ ተወላጆች ገንዘባቸውን አዋጥተው የልማት ተቋማትን እንዲመሰርቱ ትልማ አይፈቅድም" is indicated in Figure 7.

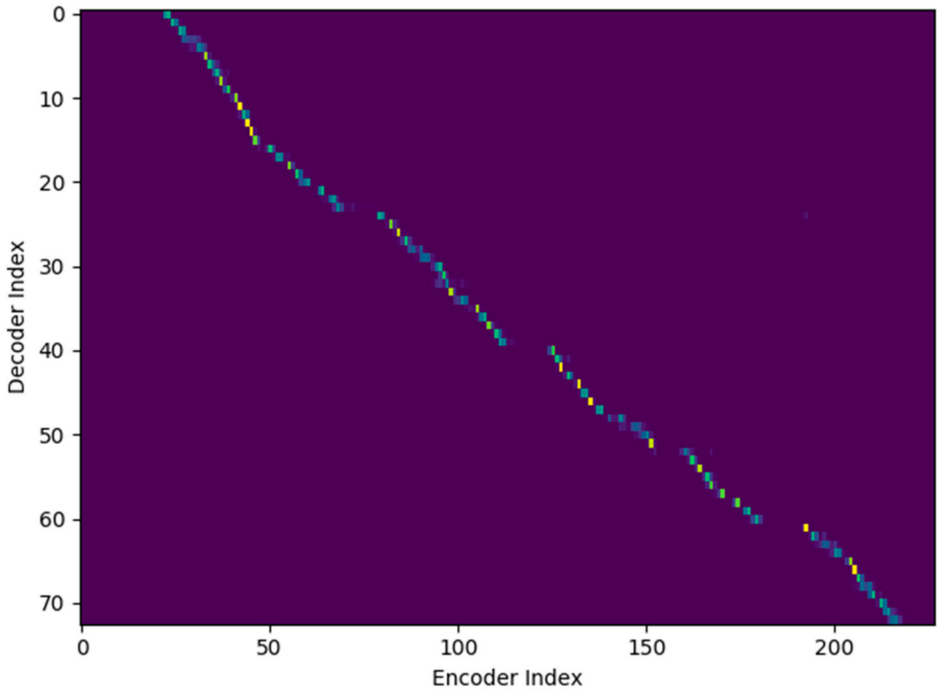

Figure 7. A phoneme-based alignment index in the CTC-attention model at the 20th epoch.

6. Conclusions and Future Works

In this paper, we proposed a subword modeling method with CTC-attention end-to-end speech recognition at the phoneme-level, which was obtained with grapheme-to-phoneme (G2P) conversion algorithms. We investigated the use of phoneme-based subwords in Amharic end-to-end ASR. During grapheme-to-phoneme conversion, a syllabification algorithm was considered for epenthesis and subword-based decoding as an extension of phoneme- and character-based subword systems. These end-to-end models were also trained using a 22 h speech dataset developed for speech recognition system and evaluated using a 5 k testing dataset. Character and phoneme Amharic language units were used as acoustic modeling units in end-to-end speech recognition approaches. To investigate the effects of OOV words in a speech recognition system, word-level RNNLMs in different vocabulary sizes, namely, 6.5 k, 10 k, 15 k, and 20 k, were also developed in both grapheme and phoneme levels due to the variation of out-of-vocabulary words. Context-independent character-based and phoneme-based RNNLM was developed for decoding purposes, and minimum CERs and PERs were obtained, respectively. The experiment was continued using subword modeling via the BPE segmentation algorithm. These subwords reduced OOV words and the minimum WER results were recorded. These results were 31.30%, 21.40%, and 18.42% in character-based subwords, phoneme-based subwords, and phoneme-based subwords with epithetic vowels, respectively. Finally, the experiment results showed that a phoneme-based BPE system with a syllabification algorithm was effective in achieving higher accuracy or minimum WER (18.42%) in the CTC-attention end-to-end method.

As future work, transformer-based end-to-end models will be used to obtain coverage to reduce the errors of the recognition system. A greater corpus size is also required in all end-to-end models; thus, collecting more data to increase the corpus size is a necessary task. In addition to minimizing the error rates, reducing latency while ensuring performance is an important research issue for end-to-end ASR models. From the perspective of the BPE segmentation algorithm, subwords types in addition to BPE can be explored and incorporated into subword regularization [42], which has been shown to improve character-based subword systems. We also plan to investigate the application of the proposed method in hybrid ASR, machine translation, and speech translation.

Author Contributions: Conceptualization, Methodology and writing original draft, E.D.E.; Methodology, advising, S.X.; formal analysis, review, Y.L.; validation, framework selection, M.D.; review and editing, A.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Key Research and Development Program of China (No.2016YFD0101900).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data can be found at https://www.openslr.org/25/.

Acknowledgments: Special thanks to Walelign Tewabe for the contribution to the Amharic language level concepts of the study.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Samples of Acmharic writing script/grapheme.

Consonants	1st Order	2nd Order	3rd Order	4th Order	5th Order	6th Order	7th Order
	ə	u	i	a	e	ɨ	o
h	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ
l	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ
m	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
s	ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ
r	ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ
.	.	.	.	.	.	.	.
f	ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ
p	ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ

Appendix B

Table A2. Amharic consonants are arranged based on articulation.

Manner of Articulation	Voicing	Labial	Dental	Palatal	Velar	Glottal
stops	Voiceless	P ፕ	t ት	k ክ	K^W ኩ	? ዕ
	Voiced	b ብ	d ድ	g ግ	g^W ጉ	
	glottalized	p′ ጵ	t′ ጥ	q ቅ	q^W ቁ	
	rounded					h ሀ
fricatives	Voiceless	f ፍ	s ስ	š ሽ		
	Voiced	v ቭ	z ዝ	ž ዥ		
	glottalized		s′ ጽ			h^W ኍ
	rounded					
Affricative	Voiceless			č ች		
	Voiced			ǧ ጅ		
	glottalized			č′ ጭ		
	rounded					
Nasals	voiced	m ም	n ን	ň ኝ		
Liquids	Voiceless		r ር			
	voiced		l ል			
Glides		w ዉ			y ይ	

References

1. Claire, W.Y.; Roy, S.; Vincent, T.Y. Syllable based DNN-HMM Cantonese speech-to-text system. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, 23–28 May 2016.
2. Novoa, J.; Fredes, J.; Poblete, V.; Yoma, N.B. Uncertainty weighting and propagation in DNN–HMM-based speech recog-nition. *Comput. Speech Lang.* **2018**, *47*, 30–46. [CrossRef]
3. Hori, T.; Cho, J.; Watanabe, S. End-to-end Speech Recognition With Word-Based Rnn Language Models. In Proceedings of the 2018 IEEE Spoken Language Technology Workshop (SLT), Athens, Greece, 18–21 December 2018; pp. 389–396.

4. Wu, L.; Li, T.; Wang, L.; Yan, Y. Improving Hybrid CTC/Attention Architecture with Time-Restricted Self-Attention CTC for End-to-End Speech Recognition. *Appl. Sci.* **2019**, *9*, 4639. [CrossRef]
5. Yoshimura, T.; Hayashi, T.; Takeda, K.; Watanabe, S. End-to-End Automatic Speech Recognition Integrated with CTC-Based Voice Activity Detection. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6999–7003.
6. Qin, C.-X.; Zhang, W.-L.; Qu, D. A new joint CTC-attention-based speech recognition model with multi-level multi-head attention. *EURASIP J. Audio Speech Music. Process.* **2019**, *2019*, 1–12. [CrossRef]
7. Graves, A.; Jaitly, N. Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of the International Conference on Machine Learning*; PMLR: Bejing, China, 2014; pp. 1764–1772.
8. Graves, A. Sequence transduction with recurrent neural networks. *arXiv* **2012**, arXiv:1211.3711.
9. Chorowski, J.K.; Bahdanau, D.; Serdyuk, D.; Cho, K.; Bengio, Y. Attention-based models for speech recognition. In *Ad-vances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2015; pp. 577–585.
10. Kim, S.; Hori, T.; Watanabe, S. Joint CTC-attention based end-to-end speech recognition using multi-task learning. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 4835–4839.
11. Watanabe, S.; Hori, T.; Hershey, J.R. Language independent end-to-end architecture for joint language identification and speech recognition. In Proceedings of the 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Okinawa, Japan, 16–20 December 2017; pp. 265–271.
12. Boyer, F.; Rouas, J.-L. End-to-End Speech Recognition: A review for the French Language. *arXiv* **2019**, arXiv:1910.08502.
13. Das, A.; Li, J.; Zhao, R.; Gong, Y. Advancing Connectionist Temporal Classification with Attention Modeling. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 4769–4773.
14. Fathima, N.; Patel, T.; C, M.; Iyengar, A. TDNN-based Multilingual Speech Recognition System for Low Resource Indian Languages. In Proceedings of the Interspeech 2018, Hyderabad, India, 2–6 September 2018; pp. 3197–3201.
15. Le, D.; Provost, E.M. Improving Automatic Recognition of Aphasic Speech with AphasiaBank. In Proceedings of the Interspeech 2016, Francisco, CA, USA, 8–12 September 2016; pp. 2681–2685.
16. Li, J.; Ye, G.; Zhao, R.; Droppo, J.; Gong, Y. Acoustic-to-word model without OOV. In Proceedings of the 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Okinawa, Japan, 16–20 December 2017; pp. 111–117.
17. Sikdar, U.K.; Gambäck, B. Named Entity Recognition for Amharic Using Stack-Based Deep Learning. In *International Conference on Computational Linguistics and Intelligent Text Processing*; Springer: Cham, Switzerland, 2018; pp. 276–287.
18. Abate, S.T.; Menzel, W.; Tafila, B. An Amharic speech corpus for large vocabulary continuous speech recognition. In Proceedings of the Ninth European Conference on Speech Communication and Technology, Lisbon, Portugal, 4–8 September 2005.
19. Melese, M.; Besacier, L.; Meshesha, M. Amharic speech recognition for speech translation. In Proceedings of the Atelier Traitement Au-tomatique des Langues Africaines (TALAF), JEP-TALN 2016, Paris, France, 4 July 2016.
20. Belay, B.H.; Habtegebrial, T.; Meshesha, M.; Liwicki, M.; Belay, G.; Stricker, D. Amharic OCR: An End-to-End Learning. *Appl. Sci.* **2020**, *10*, 1117. [CrossRef]
21. Gamback, B.; Sikdar, U.K. Named entity recognition for Amharic using deep learning. In Proceedings of the 2017 IST-Africa Week Conference (IST-Africa), Windhoek, Namibia, 30 May–2 June 2017; pp. 1–8.
22. Tachbelie, M.Y.; Abate, S.T.; Besacier, L. Using different acoustic, lexical and language modeling units for ASR of an under-resourced language–Amharic. *Speech Commun.* **2014**, *56*, 181–194. [CrossRef]
23. Dribssa, A.E.; Tachbelie, M.Y. Investigating the use of syllable acoustic units for amharic speech recognition. In Proceedings of the AFRICON 2015, Addis Ababa, Ethiopia, 14–17 September 2015; pp. 1–5.
24. Gebremedhin, Y.B.; Duckhorn, F.; Hoffmann, R.; Kraljevski, I.; Hoffmann, R. A new approach to develop a syllable based, continuous Amharic speech recognizer. In Proceedings of the Eurocon 2013, Zagreb, Croatia, 1–4 July 2013; pp. 1684–1689.
25. Kim, Y.; Jernite, Y.; Sontag, D.; Rush, A.M. Character-aware neural language models. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Taipei City, Taiwan, 26–29 November 2016.
26. Inaguma, H.; Mimura, M.; Sakai, S.; Kawahara, T. Improving OOV Detection and Resolution with External Language Models in Acoustic-to-Word ASR. In Proceedings of the 2018 IEEE Spoken Language Technology Workshop (SLT), Athens, Greece, 18–21 December 2018; pp. 212–218.
27. Zeyer, A.; Zhou, W.; Ng, T.; Schlüter, R.; Ney, H. Investigations on Phoneme-Based End-To-End Speech Recognition. *arXiv* **2020**, arXiv:2005.09336.
28. Wang, W.; Zhou, Y.; Xiong, C.; Socher, R. An investigation of phone-based subword units for end-to-end speech recogni-tion. *arXiv* **2020**, arXiv:2004.04290.
29. Xiao, Z.; Ou, Z.; Chu, W.; Lin, H. Hybrid CTC-Attention based End-to-End Speech Recognition using Subword Units. In Proceedings of the 2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP), Taipei City, Taiwan, 26–29 November 2018; pp. 146–150.
30. Yuan, Z.; Lyu, Z.; Li, J.; Zhou, X. An improved hybrid CTC-Attention model for speech recognition. *arXiv* **2018**, arXiv:1810.12020.
31. Schuster, M.; Nakajima, K. Japanese and Korean voice search. In Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012; pp. 5149–5152.

32. Huang, M.; Lu, Y.; Wang, L.; Qian, Y.; Yu, K. Exploring model units and training strategies for end-to-end speech recog-nition. In Proceedings of the 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Singapore, 14–18 December 2019; pp. 524–531.
33. Das, A.; Li, J.; Ye, G.; Zhao, R.; Gong, Y. Advancing Acoustic-to-Word CTC Model With Attention and Mixed-Units. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 1880–1892. [CrossRef]
34. Zhang, F.; Wang, Y.; Zhang, X.; Liu, C.; Saraf, Y.; Zweig, G. Fast, Simpler and More Accurate Hybrid ASR Systems Using Wordpieces. *arXiv* **2020**, arXiv:2005.09150.
35. Gokay, R.; Yalcin, H. Improving Low Resource Turkish Speech Recognition with Data Augmentation and TTS. In Proceedings of the 2019 16th International Multi-Conference on Systems, Signals & Devices (SSD), Istanbul, Turkey, 21–24 March 2019; pp. 357–360.
36. Liu, C.; Zhang, Q.; Zhang, X.; Singh, K.; Saraf, Y.; Zweig, G. Multilingual Graphemic Hybrid ASR with Massive Data Augmentation. In Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL), Marseille, France, 11–12 May 2020; pp. 46–52.
37. Laptev, A.; Korostik, R.; Svischev, A.; Andrusenko, A.; Medennikov, I.; Rybin, S. You Do Not Need More Data: Improving End-To-End Speech Recognition by Text-To-Speech Data Augmentation. *arXiv* **2020**, arXiv:2005.07157.
38. Park, D.S.; Chan, W.; Zhang, Y.; Chiu, C.-C.; Zoph, B.; Cubuk, E.D.; Le, Q.V. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv* **2019**, arXiv:1904.08779.
39. Hailu, N.; Hailemariam, S. Modeling improved syllabification algorithm for Amharic. In Proceedings of the International Conference on Management of Emergent Digital EcoSystems; Association for Computing Machinery: New York, NY, USA, 2012; pp. 16–21.
40. Mariam, S.H.; Kishore, S.P.; Black, A.W.; Kumar, R.; Sangal, R. Unit selection voice for amharic using festvox. In Proceedings of the Fifth ISCA Workshop on Speech Synthesis, Pittsburgh, PA, USA, 14–16 June 2004.
41. Hori, T.; Watanabe, S.; Hershey, J.; Barzilay, R.; Kan, M.-Y. Joint CTC/attention decoding for end-to-end speech recognition. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 518–529.
42. Watanabe, S.; Hori, T.; Kim, S.; Hershey, J.R.; Hayashi, T. Hybrid CTC/Attention Architecture for End-to-End Speech Recognition. *IEEE J. Sel. Top. Signal Process.* **2017**, *11*, 1240–1253. [CrossRef]
43. Graves, A.; Fernández, S.; Gomez, F.; Schmidhuber, J. Connectionist temporal classification: Labelling unsegmented se-quence data with recurrent neural networks. In Proceedings of the 23rd International Conference on Machine Learning; Association for Computing Machinery: New York, NY, USA, 2006; pp. 369–376.
44. Li, J.; Ye, G.; Das, A.; Zhao, R.; Gong, Y. Advancing Acoustic-to-Word CTC Model. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5794–5798.
45. Moritz, N.; Hori, T.; Le Roux, J. Triggered Attention for End-to-end Speech Recognition. In Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 5666–5670.
46. Shan, C.; Zhang, J.; Wang, Y.; Xie, L. Attention-Based End-to-End Speech Recognition on Voice Search. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 1–5.
47. Schuller, B.; Steidl, S.; Batliner, A.; Marschik, P.B.; Baumeister, H.; Dong, F.; Hantke, S.; Pokorny, F.B.; Rathner, E.-M.; Bartl-Pokorny, K.D.; et al. The interspeech 2018 computational paralinguistics challenge: Atypical\& self-assessed affect, crying\& heart beats. In Proceedings of the INTERSPEECH, Hyderabad, India, 2–6 September 2018; Volume 5.
48. Tjandra, A.; Sakti, S.; Nakamura, S. Attention-based Wav2Text with feature transfer learning. In Proceedings of the 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Okinawa, Japan, 16–20 December 2017; pp. 309–315.
49. Chan, W.; Jaitly, N.; Le, Q.; Vinyals, O. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 4960–4964.
50. Chen, M.; He, X.; Yang, J.; Zhang, H. 3-D convolutional recurrent neural networks with attention model for speech emo-tion recognition. *IEEE Signal Process. Lett.* **2018**, *25*, 1440–1444. [CrossRef]
51. Ueno, S.; Inaguma, H.; Mimura, M.; Kawahara, T. Acoustic-to-word attention-based model complemented with charac-ter-level CTC-based model. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5804–5808.
52. Tachbelie, M.Y.; Abate, S.T.; Menzel, W. Morpheme-based automatic speech recognition for a morphologically rich language-amharic. In Proceedings of the Spoken Languages Technologies for Under-Resourced Languages, Penang, Malaysia, 3–5 May 2010.
53. Mittal, P.; Singh, N. Subword analysis of small vocabulary and large vocabulary ASR for Punjabi language. *Int. J. Speech Technol.* **2020**, *23*, 71–78. [CrossRef]
54. Shaik, M.A.B.; Mousa, A.E.-D.; Hahn, S.; Schlüter, R.; Ney, H. Improved strategies for a zero OOV rate LVCSR system. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 19–24 April 2015; pp. 5048–5052.

55. Xu, H.; Ding, S.; Watanabe, S. Improving End-to-end Speech Recognition with Pronunciation-assisted Sub-word Model-ing. In Proceedings of the ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 7110–7114.
56. Soltau, H.; Liao, H.; Sak, H. Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition. *arXiv* **2016**, arXiv:1610.09975.
57. Andrusenko, A.; Laptev, A.; Medennikov, I. Exploration of End-to-End ASR for OpenSTT-Russian Open Speech-to-Text Dataset. *arXiv* **2020**, arXiv:2006.08274.
58. Sennrich, R.; Haddow, B.; Birch, A. Neural machine translation of rare words with subword units. *arXiv* **2015**, arXiv:1508.07909.
59. Markovnikov, N.; Kipyatkova, I. Investigating Joint CTC-Attention Models for End-to-End Russian Speech Recognition. In *International Conference on Speech and Computer*; Springer: Cham, Switzerland, 2019; pp. 337–347.