mathematics

Article

Deep Models for Low-Resourced Speech Recognition: Livvi-Karelian Case

Irina Kipyatkova * and Ildar Kagirov

St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), 199178 St. Petersburg, Russia; kagirov@iias.spb.su

* Correspondence: kipyatkova@iias.spb.su

Abstract: Recently, there has been a growth in the number of studies addressing the automatic processing of low-resource languages. The lack of speech and text data significantly hinders the development of speech technologies for such languages. This paper introduces an automatic speech recognition system for Livvi-Karelian. Acoustic models based on artificial neural networks with time delays and hidden Markov models were trained using a limited speech dataset of 3.5 h. To augment the data, pitch and speech rate perturbation, SpecAugment, and their combinations were employed. Language models based on 3-grams and neural networks were trained using written texts and transcripts. The achieved word error rate metric of 22.80% is comparable to other low-resource languages. To the best of our knowledge, this is the first speech recognition system for Livvi-Karelian. The results obtained can be of a certain significance for development of automatic speech recognition systems not only for Livvi-Karelian, but also for other low-resource languages, including the fields of speech recognition and machine translation systems. Future work includes experiments with Karelian data using techniques such as transfer learning and DNN language models.

Keywords: low-resource languages; automatic speech recognition; audio data augmentation; time delay neural network; hidden Markov models; long short-term memory

MSC: 68T10

check for updates

Citation: Kipyatkova, I.; Kagirov, I. Deep Models for Low-Resourced Speech Recognition: Livvi-Karelian Case. *Mathematics* **2023**, *11*, 3814. https://doi.org/10.3390/math11183814

Academic Editors: Jianping Gou and Junlin Hu

Received: 30 June 2023
Revised: 26 August 2023
Accepted: 4 September 2023
Published: 5 September 2023

1. Introduction

Current systems for automatic natural language processing (speech recognition [1], speech classification [2,3], sentiment analysis [4], and machine translation [5]) are developed with the use of machine learning technologies that require large datasets. However, the number of idioms with big language resources is very limited. According to various estimates, there exist between 5000 to 7000 languages in the world, and only about 20 of them can be considered as having sufficient data resources [6–8]. The overwhelming majority of the world's languages are poorly described and documented, and their linguistic data is often difficult to obtain. Any application of modern natural language processing methods to these languages, usually labeled as low-resource languages, is hindered. This problem has been repeatedly addressed in scientific publications [9–12]. Another important issue regarding natural speech processing is the choice of optimal training algorithms and data augmentation. The present paper provides an example of the application of speech technologies to low-resource languages. Among other research problems, data collection and preparation for use in speech recognition systems (and more broadly, in natural language processing) are discussed.

Our contributions are outlined as follows. Firstly, the speech and text corpora of Livvi-Karelian were collected and processed. Secondly, acoustic and language models for Livvi-Karelin speech recognition were developed. To the best of our knowledge, this is the first research endeavor on training a Livvi-Karelian speech recognition system. The information about this project is available on the website: https://hci.nw.ru/en/projects/25, accessed on 28 June 2023. The paper's structure is as follows: following this introduction, the issues of low-resource speech recognition are discussed in Section 2, and up-to-date methods for speech signal augmentation are reviewed. In Section 3, the collected Livvi-Karelian speech corpus is presented, and the augmentation procedures applied to these data described in detail. Section 4 provides a brief overview of speech recognition systems, and explicitly describes the developed Livvi-Karelian speech recognition system. The following Section 5 provides the reader with the results of experiments, which are further discussed in Section 6. A summary of the paper can be found in the last section, Section 7.

2. Low-Resource Languages: Data Scarcity Challenge

2.1. Low-Resource Speech Recognition

The Karelian language belongs to the Balto-Finnic group the Uralic language family. Linguists distinguish three main dialects of Karelian: Karelian Proper, Livvi-Karelian, and Luudi-Karelian [13]. It is worth mentioning, however, that Luudi-Karelian is treated as a separate language (Ludian) in some works [14]. Since today Livvi-Karelian is the most widespread dialect of Karelian [15], being widely represented in the Karelian media, the authors of this paper only focused on Livvi-Karelian data.

Livvi-Karelian falls within the category of the "low-resource languages". Under the term "low-resource languages" (or "under-resourced languages") are meant languages with a limited number of electronic resources available. This term was first introduced in [16,17]. A set of criteria was proposed to classify a language as low-resource, among which were a writing system, availability of data on the Internet, descriptive grammars, electronic bilingual dictionaries, parallel corpora, and others. In the following works [18], the notion of low-resource languages was further expanded to consider factors such as a low social status of a language and its limited study. Nowadays, however, the main criterion for classifying a language as low-resource is the scarcity of electronic data available to researchers [19].

Low-resource languages hold significance not solely for linguists due to their role as means of communication in many societies. Currently there exist about 2000 low-resource languages spoken by more than 2.5 billion people in Africa and India alone. Developing tools for natural communication with speakers of these languages can help address a wide range of economic, cultural, and environmental issues.

The scarcity of language data is a complex problem that impacts various aspects of language processing: phonetic, lexical, and grammatical [20]. In simple terms, lack of data hampers the direct application of "classical" approaches to automatic speech recognition and translation, which usually imply the use of acoustic, lexical, and grammatical (language) models. Usually, an automatic speech recognition (ASR) system (the "standard" approach) consists of an acoustic model (AM) that establishes the relationship between acoustic information and allophones of a language at issue [21], a language model (LM) necessary for building hypotheses of a recognized utterance, and a vocabulary of lexical units with phonetic transcriptions. The training of acoustic models involves utilizing a speech corpus, while the development of the language model draws upon probabilistic modeling using available target language texts (as illustrated in Figure 1).

A speech recognition system operates in two modes: training and recognition. In the training mode, acoustic and language models are created, and a vocabulary of lexical units with transcriptions is built up. In the recognition mode, the input speech signal is converted into a sequence of feature vectors, and the most probable hypothesis is found using pretrained acoustic and language models [22]. For this purpose, the maximum probability criterion is employed:

$$W_{hyp} = \underset{W}{\mathrm{argmax}} P(O|W)P(W), \qquad (1)$$

where O represents a sequence of feature vectors from a speech signal and W is a set encompassing all potential sequences of words. The probability $P(O \mid W)$ is calculated with

AM, while the probability $P(W)$ is derived through LM. Hidden Markov models (HMM) can be used as AM, with each acoustic unit being modeled by one HMM, typically with three states. In this case acoustic probability is computed using the following formula [23]:

$$P(O|W) = \sum_{q} P(O|q,W)P(q|W) \approx \max_{q|w} \pi(q_0) \prod_{t=1}^{T} a_{q_{t-1}q_t} \prod_{t=0}^{T} P(O_t|q_t), \tag{2}$$

where q is a sequence of HMM states, $\pi(q_0)$ and $a_{q_{t-1}q_t}$ are the initial state probability and state transition probability, respectively, determined by the HMM, and q_t is a state of HMM at time t.

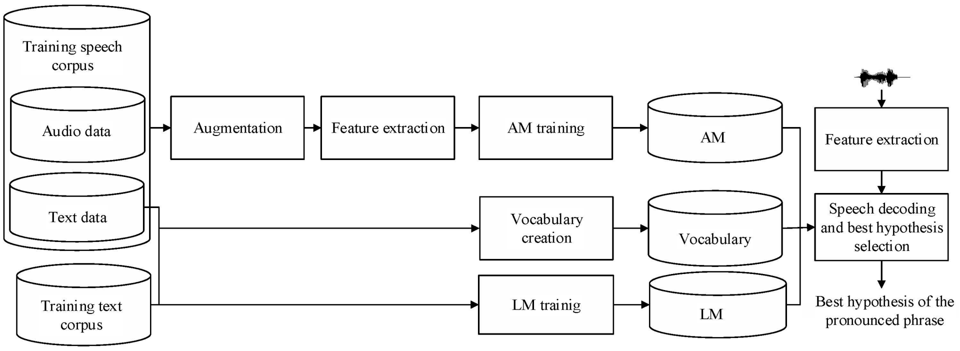


Figure 1. Outline of the Karelian Speech Recognition System.

Nowadays deep neural networks (DNNs) are widely used for training both the acoustic and language models. For acoustic models, DNN are combined with HMM, forming the so-called hybrid DNN/HMM models. In this case, DNN are employed to derive the posterior probability of HMM, wherein HMM capture long-term dependencies and DNN contribute discriminant training capabilities. At the decoding stage, posterior probability $P(O_t \mid q_t)$ should be converted to the likelihood [23]:

$$P(O_t|q_t) = \frac{P(q_t|O_t)P(O_t)}{P(q_t)}, \tag{3}$$

where $P(O_t)$ is independent of the word sequence and therefore it can be ignored. Thus, for the decoding pseudo-likelihood is used:

$$P(O_t|q_t) = \frac{P(q_t|O_t)}{P(q_t)}, \tag{4}$$

LMs are typically developed using either a statistical n-gram methodology or a recurrent NN (RNN) approach. In RNN, the hidden layer stores all preceding history, in contrast to feedforward NN which can store context only of restricted length.

The main methods for acoustic and language modeling are summarized in Table 1.

Despite the recent widespread use of the end-to-end [24] approach to speech recognition, the standard approach remains the preferred choice for low-resource languages due to its requirement for less training data. For example, the hybrid DNN/HMM approach was used in [25] for speech recognition in the low-resource Sinhala language. The results obtained by the authors show that the use of hybrid DNN/HMM acoustic models outperforms HMM based on Gaussian mixture models (GMM) by 7.48 in terms of word error rate (WER) on the test dataset.

Table 1. The main methods for acoustic and language modeling.

Methods	Advantages	Disadvantages
Acoustic modeling		
HMM	• Easy to implement • Ability to process variable-length inputs • Effective modeling both temporal and spectral variations of speech	• Poor discriminative power • Sequencies of observation vectors are consider to be statistically independent but it is not the case for speech signal • Cannot take into account long-term dependencies • Require the usage of LM and vocabulary
DNN/HMM	• Combine advantages of HMM and DNN with HMMs supporting long-term dependencies and DNN providing discriminative training	• Need more training data than HMM • Require the usage of LM and vocabulary
End-to-end	• Do not require LM and vocabulary • Higher decoding speed	• Need much more training data (thousand hours of speech)
Language Modeling		
n-gram	• Easy to implement • Can be used directly at decoding	• Poorly capturing long distance context
Recurrent DNN	• Allows to take into account long distance context	• High computation complexity • Difficult to use directly at decoding

In [26], the results of experiments on multilingual speech recognition of low-resource languages (10 languages from the set proposed as part of the OpenASR20 contest, as well as North American Cree and Inuit languages) were presented. The authors experimented with factorized time delay neural networks (TDNNs)—TDNN-F in hybrid DNN/HMM acoustic models and have shown that this architecture outperforms long short-term memory (LSTM) neural networks (NNs) in terms of WER. A similar conclusion was made in [27] for the Somali language data.

In a range of studies addressing the Russian language, it has also been shown that hybrid acoustic models based on TDNN are superior to HMM, or hybrid DNN/HMM [28,29].

Based on these examples, the authors of this paper decided to adopt the standard approach in developing their speech recognition system for Livvi-Karelian, and to choose hybrid DNN/HMM acoustic models.

2.2. Speech Data Augmentation: Main Approaches

As previously mentioned, one of the most important prerequisites for an ASR system development is the availability of training data (audio and text corpora). This holds particular significance for the Karelian language. An effective approach to the data scarcity problem is data augmentation. Data augmentation refers to a set of methods used to create additional data for training models, either by modifying data or by adding data from external databases. It is well known that augmentation techniques can solve overfitting problems and improve the performance of ASR systems [30] to the audio spectrogram. By employing these augmentations, the dataset is effectively expanded due to numerous variations of input data. One can list the following methods for data augmentation: speech signal modification, spectrogram modification, data generation.

2.2.1. Speech Signal Modification

Speech signal modification can be performed by changing voice pitch, speech rate, and speech volume, adding noise, and modifying features extracted from the speech signal. An illustrative example of speech signal modification through augmentation is presented in [31], where the authors changed speech rate by multiplying the original speed by coefficients of 0.9, 1.0, and 1.1. The effect of these transformations was a 4.3% reduction in WER. Augmentation by adding random values to speech features is presented in [32]. Some researchers combine several types of augmentation. For example, a two-stage speech data augmentation is proposed in [33]. In the first stage, random noise was added and speech rate was modified in order to enhance the robustness of acoustic models. In the second stage, feature augmentation was performed on the adapted speaker-specific features.

Voice conversion technology can also be classified as an augmentation technique, involving modification of the speaker's voice (source voice) so that it sounds like another speaker's (target voice) while linguistic features of the speech remain unchanged [34]. Generative Adversarial Networks (GANs) are commonly used for this purpose [35], along with their variants, such as Wasserstein GAN and StarGAN. In [36], a method called VAW-GAN is proposed for non-parallel voice conversion, combining a conditional variational autoencoder (C-VAE) and Wasserstein GAN. The former models the acoustic features of each speaker; the latter synthesizes the voice of another speaker. The StarGAN architecture is used in [37], which presents the StarGAN-VC method for voice conversion. Several types of data augmentation were applied in the work [38] for Turkish speech recognition. The authors explored different augmentation techniques, such as speech rate modification, volume modification, joint modification of speech rate and volume, and speech synthesis (investigating Google's speech-to-text conversion system and an integrated system for synthesizing Turkish speech based on deep convolutional networks). Additionally, various combinations of the described methods were employed. The best result was achieved by jointly applying all methods, resulting in a 14.8% reduction in WER.

Mixup [39] is another speech augmentation method that creates new training samples with linear interpolation between pairs of samples and their corresponding labels. Mixup generates a new training sample by taking a weighted average of the input features and labels for two randomly selected samples and their labels. This encourages the model to learn from the combined characteristics of multiple samples, leading to better generalization. Mixup has been widely used in image classification tasks, but can also be effectively applied to other domains, such as audio processing.

The SamplePairing technique [40] involves pairing samples from the training set in random order. For each pair, the features are combined by taking an average value, in a way similar to the Mixup technique. The labels of the paired samples are typically ignored during training, and the model is trained to predict an average output. This method enhances robustness by exposing the model to diverse combinations of samples.

Mixup with label preserving methods is another augmentation approach [41] which extends Mixup by incorporating label preserving. It applies modifications to the input features while preserving the original labels. This results in model learning invariant features while maintaining the correct class assignments.

2.2.2. Spectrogram Modification

SpecAugment [42] operates on the spectrogram representation of audio signals; the main idea behind this technique is applying a range of random transformations to the spectrogram during training. These transformations include time warping, frequency masking, and time masking.

Time warping in SpecAugment involves stretching or compressing different segments of the spectrogram in the time domain. It introduces local temporal distortions by warping the time axis of the spectrogram. This transformation helps the ASR model handle variations in speech speed, allowing it to be more robust to different speaking rates exhibited by different speakers.

Time masking technique allows selecting anchor points along the time axis of the spectrogram and warping the regions between them. The anchor points are randomly chosen, and the regions between them are stretched or compressed by a certain factor. Stretching or compressing introduces local temporal distortions, simulating variations in speech speed.

Frequency masking transformation masks consecutive frequency bands in the spectrogram. The model becomes more robust regarding variations in pitch and speaker characteristics due to randomly masking a range of frequencies, while time masking transformation masks consecutive time steps in the spectrogram. By randomly masking out segments of the audio signal, the model learns to be invariant to short-term temporal variations.

Vocal Tract Length Perturbation (VTLP) [43] is a method of spectrogram transformation using random linear distortion by frequency measurement. The main idea is to apply normalization, not to remove variations but, on the contrary, to add variations to audio. This can be obtained by normalizing to an arbitrary target instead of normalizing to a canonical mean. For VTLP, a deformation coefficient is generated for each sample, and the frequency axis is deformed so that the frequency (f) is mapped to the new frequency (f′). VTLP is typically applied by modifying the speech features, such as the mel-frequency cepstral coefficients (MFCCs) or linear predictive coding (LPC) coefficients. The perturbation is achieved by scaling the feature vectors along the frequency axis, mimicking the effects of different vocal tract lengths on the spectral envelope of the speech signal.

2.2.3. Data Generation

Another method of speech data augmentation is speech synthesis. Recently, among the most successful models are WaveGAN and SpecGAN [44,45]. The main difference between WaveGAN and SpecGAN lies in the domain in which they generate audio data. WaveGAN is a GAN used for the synthesis of high-fidelity raw audio waveforms. WaveGAN consists of two main modules: a generator and a discriminator. The generator network takes random noise as input and generates synthetic audio waveforms. The discriminator network tries to distinguish between the real audio samples from the dataset and the generated samples from the generator. In order to capture the temporal dependencies WaveGAN uses a convolutional neural network (CNN) architecture for both the generator and discriminator networks. The generator progressively upsamples the noise input using transposed convolutions to generate longer waveforms, while the discriminator performs convolutions to analyze and classify the input waveforms.

SpecGAN, also known as Spectrogram GAN, is a GAN architecture network specifically designed for the generation of audio spectrograms. It generates audio by synthesizing spectrograms and is trained on spectrograms extracted from real audio data.

The Tacotron 2 generative NN model [46] developed by Google is used for speech synthesis. For instance, this method was used in the work [47] for synthesizing child speech in the Punjabi language. Furthermore, in this study, augmentation was achieved by modifying formants in the speech recordings of an adult speech corpus. In [48], speech synthesis was employed for augmenting speech data in the development of an integrated speech recognition system, significantly reducing WER. Additionally, SpecAugment was applied, resulting in further WER reduction. A drawback of this method is a requirement for training speech data. Insufficient training data may result in unsatisfactory synthesized speech quality. An illustrative case is [49], where the incorporation of synthesized data during the training of acoustic models failed to enhance recognition accuracy. In their work, the researchers employed statistical parametric speech synthesis techniques. Despite their efforts, they encountered challenges in achieving satisfactory quality when synthesizing speech through Tacotron 2 and WGANSing models (a speech synthesizer utilizing GANs). The authors attributed the poor synthesis quality to a lack of training data.

The main approaches to speech data augmentation are presented in Table 2:

Table 2. Speech Data Augmentation Methods.

Augmentation Method	Modified Features	Short Description
Speech Signal Modification	Tempo [31]	Modification of speech rate by certain coefficients
	Feature Perturbation [32]	Adding random values to speech features
	Mixup [39] SamplePairing [40] Mixup with label preserving methods [41]	Taking a weighted average of the input features mentation
Spectrogram Modification	SpecAugment [42]	Warping the features, masking blocks of frequency channels, and masking blocks of time steps
	VTLP [43]	Random linear distortion by frequency
Data Synthesis	WaveGAN [44] SpecGAN [45]	GAN architecture for voice and spectrogram augmentation
	Tacotron 2 [47]	Encoder-decoder architecture

Overall, the best augmentation technique or combination of techniques depends on the specific ASR task, available training data, and desired improvements in performance. It is often beneficial to experiment with multiple techniques and assess their impact on the ASR system's performance. For example, speech signal modification must not make speech data linguistically implausible. Voice conversion requires parallel training data, where the source and target voices are aligned, which may limit its application in some scenarios. The quality of voice conversion may vary depending on the training data and the similarity between source and target speakers. Spectrogram modification usually requires fine-tuning to strike a balance between data diversity and maintaining speech quality. The effectiveness of Mixup and SamplePairing techniques, as well as data generation, may vary drastically depending on dataset, potentially resulting in unsatisfactory speech quality.

Concluding the review of related work, it should be noted that DNN-based ASR systems are typically trained using tens and hundreds of hours of speech data. The current research aimed to investigate the feasibility of training DNN models for an ASR system on very limited training data, approximately 3 h of speech, and explore data augmentation methods to enhance speech recognition results. Moreover, despite a long-standing literature tradition and current interest of linguists to the language and folklore of the Karelians, Karelian remains an under-resourced language. The survey of related work has shown that there is no ASR system for Karelian language.

3. Speech Corpus Preparation

3.1. Karelian Data

To collect Karelian speech data, recordings of radio broadcasts were employed, including interviews featuring two or more Karelian speakers (a total of 15, of which 6 were men and 9 were women). Utterances which were found as unfit for processing were removed from the corpus. The primary reasons for utterance exclusion were instances of speech overlap between different speakers, code-switching from Livvi-Karelian to Russian, and the significant presence of background noise or music. Notably, the research focus did not encompass noise reduction techniques or code-switching, and hence these samples were removed from the dataset. The final corpus volume is 3.5 h; the total number of recorded utterances was 3819. The data were subsequently partitioned into a training subset, which comprises 90% of the utterances, and a separate test subset, encompassing 10% of the utterances (Table 3).

Table 3. Karelian Speech Corpus.

Corpus Features	Value
Number of Speakers	15 (6 male, 9 female)
Total Duration	3.5 h
Total Data Volume	2.2 Gb
Number of Utterances	3819
Sampling Frequency	16,000 Hz
Quantization	16 Bit
Training/Testing Datasets Ratio	9:1

3.2. Speech Data Augmentation

In order to enlarge the volume of training speech data, augmentation was conducted. Pitch and speech rate perturbation, along with SpecAugment and their combinations were employed. To perform pitch and speech rate perturbation, SoX toolkit (http://sox.sourceforge.net/sox.html, accessed on 28 June 2023) was used. The pitch was altered on the number of semitones obtained randomly from uniform distribution in range [−2, 2]. Speech rate perturbation was achieved using the tempo command in SoX, which modifies the tempo without affecting the pitch. The tempo was adjusted by a coefficient randomly chosen from a uniform distribution in the range of [0.7, 1.3]. Additionally, simultaneous modification of both pitch and speech rate were performed, resulting in three modified copies of the speech data.

For spectrogram modification, SpecAugment implemented in the Kaldi [50] toolkit was used as the "component spec-augment-layer". This technique involves time and frequency masking operations. In Kaldi, SpecAugment is applied randomly on-the-fly during each epoch, ensuring that the volume of the training data remains unchanged. The authors of this study set the maximum proportion of frequency frames to be zeroed out as 0.5, the proportion of time frames to be zeroed out as 0.2, and the maximum length of a zeroed region in the time axis to 20 frames.

4. Karelian Speech Recognition System

4.1. Acoustic Models

The task of acoustic models involves predicting the sequence of phonemes based on the audio input. To initiate this process, the selection of an appropriate phoneme set becomes of high importance. In the context of this paper, it is necessary to briefly introduce some concepts which had been developed within Karelian studies, viz. treatment of long and short phonemes. The distinction in duration is fundamental within the Balto-Finnic phonological systems, and all researchers have identified long and short phonemes.

In the descriptions of Karelian, up to four consonant duration degrees are present: extra-short (glides in borrowings from Russian), short, half-long (a geminated consonant after a sonorant in closed syllables) and geminates proper [51,52]. As [15] argues, in fact, these degrees are hardly observed without special equipment, often being results of allophonic alternations. Thus, the most promising solution, chosen as a guideline within the present paper, is identifying short and long consonants (geminates) only, especially since this opposition is further supported by minimal pairs.

As for duration degrees of the Balto-Finnic vowels, most grammars introduce the basic opposition of short and long phonemes. In some applied research papers, however, long vowels are treated as geminated ones [53], but the main reason for this solution seems to be that detection of long duration degrees is not easy from a technical point of view, since it requires observation of such prosodic features as neighboring syllables and word structure [54]. In this paper, the traditional approach is used, and long and short vowels are treated as different.

In the phoneme set used, stressed and unstressed vowels were distinguished; additionally, the back row allophone of the /i/ phoneme was considered as an independent phoneme, and hard and soft variants of consonants were distinguished. Long vowels were treated as separate phonemes. Long consonants were treated as reduplicated phonemes of the given phoneme.

For acoustic modeling hybrid DNN/HMM models with TDNN-Fs proposed in [55] were used. TDNN is 1-demensional Convolutional Neural Networks (1-d CNNs). TDNN-F is the TDNN with layers compressed via Singular Value Decomposition (SVD). As shown in [55], when applying SVD on $m \times n$ weight matrix A, one gets:

$$A_{m\times n} = Q_{m\times m} D_{m\times n} G^{T}_{n\times n}, \tag{5}$$

where D is a diagonal matrix with singular values of matrix A on the diagonal in decreasing order and Q and G are orthogonal matrixes.

Thus, the idea of TDNN-F consists in taking the existing TDNN topology and factorizing it into products of two smaller matrices with the following discarding of the smaller singular values. Then parameters of the network are fine-tuned.

Acoustic models were developed using the chain model from the Kaldi s5c recipe. In the Kaldi recipe, the initial step involves training GMM/HMM models using 13-dimensional MFCC features, along with their first and second derivatives. Subsequently, a series of iterative techniques are applied, including Linear Discriminant Analysis (LDA), Maximum Likelihood Linear Transform (MLLT), Speaker Adaptive Training (SAT), and feature space Maximum Likelihood Linear Regression (fMLLR). The resulting fMLLR models are then utilized to generate force-alignment for NN training. The architecture of the neural network is illustrated in Figure 2.

The network takes acoustic features that consist of 40-dimensional MFCC features and 100-dimensional i-vectors as input. For the application of SpecAugment, the MFCCs are converted to Mel filterbanks using inverse discrete cosine transform (DCT). DNN architecture comprises three TDNN-F blocks. The first block consists of three TDNN-F layers and processes the input vectors with a time context of {−1, 0, 1}. The second block consists of a single TDNN-F layer without splicing. The third block consists of ten TDNN-F layers operating with a time context of {−3, 0, 3}. Each TDNN-F layer has a dimension of 1024 and a bottleneck dimension of 128. Within each TDNN block, a TDNN layer is followed by a rectified linear unit (ReLU) activation function and batch normalization. Skip connections, as introduced in [55], are utilized in the TDNN layers, where the output of each layer (except the first layer) is appended to the output of the previous layers with a scale factor of 0.66. The TDNN-F layers are followed by a linear layer with a dimension of 256.

The training process was conducted using the lattice-free maximum mutual information (LF-MMI) criterion as the objective function [56]. In contrast to traditional MMI where word-level LM is used, LF-MMI employs a phone-level LM. The LF-MMI criterion modifies the equation for MMI criteria by excluding the acoustic scaling factor and dividing by the prior. Thus, the LF-MMI criterion is defined as follows [57]:

$$F(\theta) = \sum_{u=1}^{U} \log \frac{p(O_u|M_{s_u}, \theta)p(s_u)}{p(O_u|M_{den}, \theta)}, \tag{6}$$

where U is the set of all utterances, u is an utterance, O_u and s_u are the acoustic feature vector sequences and their corresponding phone sequences of the u-th training utterance, M_{s_u} is the numerator graph that is utterance specific, M_{den} is the denominator graph that is a finite-state transducer (FST) graph that includes all possible sequences of words, and θ is the model parameter. The network has two output blocks that are the block based on LF-MMI criteria (chain) and the block which uses cross-entropy (CE) criteria (Xent) [56]. LF-MMI and CE criteria are combined with weighted sum [58]:

$$\begin{gathered} L_{funal}(x) = L_{LFMMI}(x) + \alpha L_{CE}(x) \\ L_{CE}(x) = -\sum_{j} d_j \log q_j \end{gathered} \tag{7}$$

where L_{LFMMI} is LF-MMI loss, L_{CE} is CE loss, α is interpolation weight, d is reference output, and the network output for each training step is j.

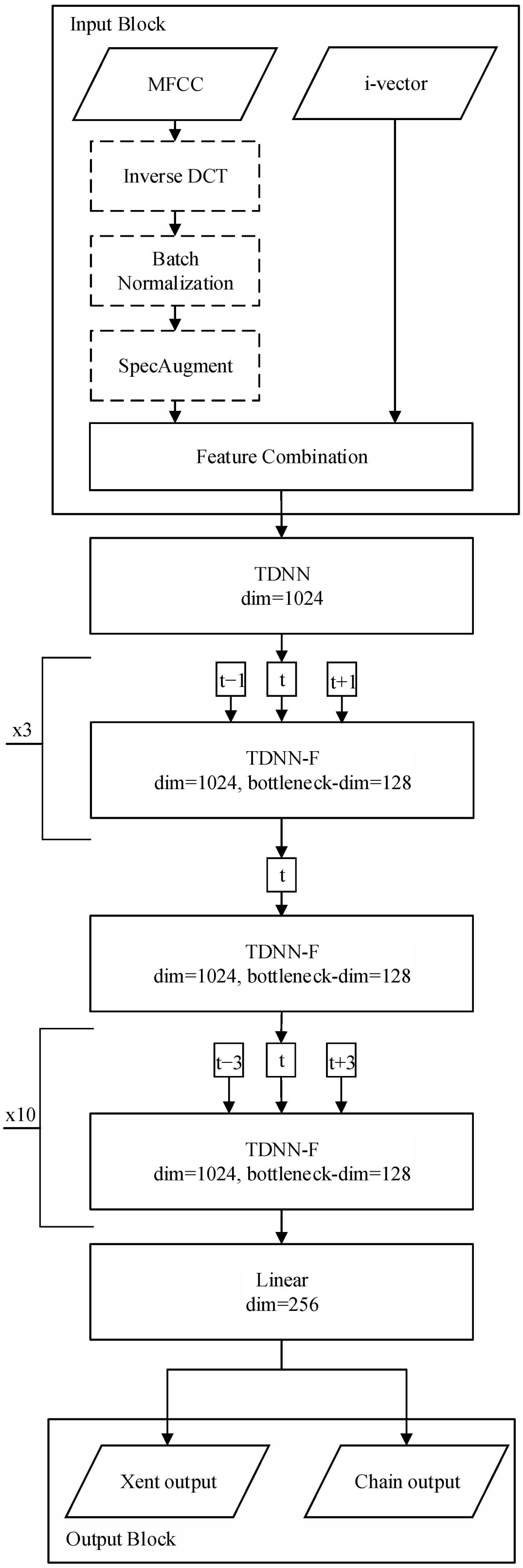


Figure 2. DNN architecture for acoustic modeling.

During the training stage, both the LF-MMI and CE criteria are employed for loss computation. This approach helps prevent overtraining and improves the generalization of the model. However, during the decoding stage, only the LF-MMI criterion is utilized.

The learning rate is adjusted dynamically throughout the training process. It starts at 0.0005 and gradually decreases to 0.00005. In the selected implementation of Kaldi, early stopping utilizing a validation set is not employed; instead, a fixed number of epochs is utilized. The specific number of epochs was determined through preliminary experiments. The training was conducted for a total of eight epochs. The number of training epochs was increased when applying SpecAugment, as recommended in [42]. In this case, the training was extended to 48 epochs to account for the augmented data and enhance the model's performance. The batch size was 64. Training was carried out on Nvidia GeForce GTX 1080 Ti GPU with CUDA.

4.2. Language and Lexical Models

For LM training and vocabulary creation, a text corpus was collected, consisting of various sources. These data include texts from the open corpus of Veps and Karelian languages "VepKar" (http://dictorpus.krc.karelia.ru/ru, accessed on 28 June 2023), Livvi-Karelian periodicals, and transcriptions from the training speech corpus. The corpus comprises approximately 5 million words. The text corpus was divided into training and validation parts in a ratio of 9:1.

In order to develop the vocabulary for the system, all the words from the transcriptions in the training data were included, along with words from other texts that appeared in the corpus at least twice. This solution was chosen due to abundance of texts in PDF format, which were converted to text using semi-automatic text recognition tools, thereby introducing the possibility of errors. By excluding words that occurred only once, the final dictionary size was reduced to 143.5 thousand words. Phonemic transcriptions were generated automatically using a software module that performed grapheme-to-phoneme conversion for a given Karelian word list. The process of automatic transcription consists of the following steps:

1. the first vowel in the word is marked as stress vowel;
2. marking palatalization of the consonant preceding front vowels;
3. processing the long phonemes.

Two types of LMs were created: a statistical n-gram LM (SLM) used for the decoding stage and an NN-based LM employed for 500-best list rescoring. The statistical word-based n-gram LM predicts the probability of a word (w_t) given a sequence of n-1 preceding words [59]:

$$p(w_1, w_2, \ldots, w_N) = \prod_{t=1}^{N} p(w_t | w_1 w_2 \ldots w_{t-1}) \tag{8}$$

During the research, 3-gram LM were created with Kneser–Ney discounting using the SRI language modeling toolkit (SRILM) [60].

The neural LM utilized in the system is based on an LSTM network. A LSTM unit consists of a cell that stores information and gates that regulate the flow of information [61].

Within a basic LSTM, there are three gates: the input gate, the forget gate, and the output gate. The input gate determines which information should be stored in the cell, the forget gate determines which information should be discarded from the cell, and the output gate controls which information should be output from the cell. This architecture enables the LSTM network to effectively capture and retain long-term dependencies in the input data [61]. The LSTM model is defined as follows:

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\ f_t &= \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ h_t &= o_t \circ \tanh(c_t) \end{aligned} \tag{9}$$

where x_t is the input vector at time t, h_t is the vector of hidden state at time t, i_t, o_t, f_t are vectors of the input, output, and forget gate, respectively, W_i, W_o, and W_f are weight matrices of the input, output, and forget gate, respectively, W_h is the weight matrix of recurrent connection, b is bias, σ is sigmoid activation function, *tanh* is hyperbolic tangent activation function, and $\circ$ is element-wise product.

LSTM-based LM was trained using TheanoLM toolkit [62] and consists of the projection layer with the size of 500, two LSTM layers with the size of 512 with dropout of 0.5, and the softmax layer. the optimization criterion was Nesterov momentum. Batch size was equal to 16. The scheme of LSTM-based LM is presented in Figure 3.

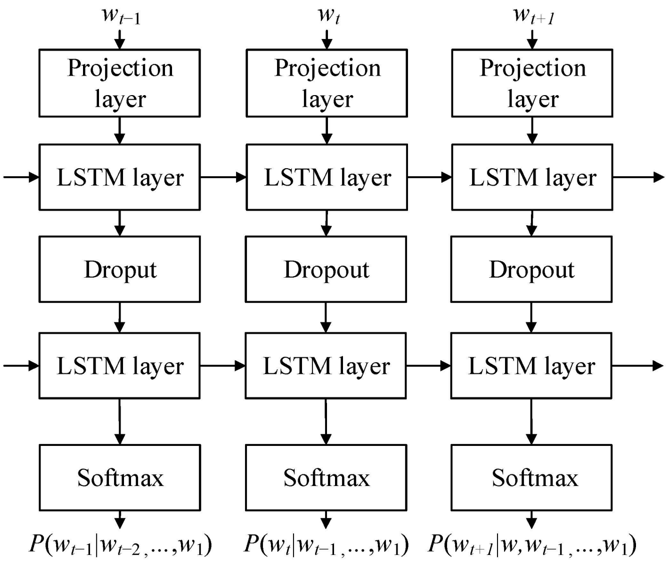


Figure 3. Architecture of LSMT-based LM.

Additionally, a linear interpolation of 3-gram and LSTM-based LMs was made. In this case, the probability score was computed as follows [63]:

$$P_{LM\_Int}(w_t|w_{t-1},\ldots,w_1) = \lambda P_{LSTM}(w_t|w_{t-1},\ldots,w_1) + (1-\lambda)P_{SLM}(w_t|w_{t-2},w_{t-1}), \tag{10}$$

where $P_{LSTM}(w_t \mid w_{t-1},\ldots,w_0)$ is a probability computed by the LSTM-based LM; $P_{SLM}(w_t \mid w_{t-2}, w_{t-1})$ is a probability computed by the statistical 3-gram model; and λ is an interpolation coefficient.

NN-based LM is typically incorporated in a two-pass decoding approach. In this scenario, the initial decoding occurs using an n-gram model, resulting in the generation of an N-best list. Subsequently, during the second pass, the N-best list undergoes rescoring through the utilization of the NN-based model. The procedure for speech recognition with N-best list rescoring is illustrated in Figure 4.

In the case of application of N-best rescoring, ASR system generates a list of hypotheses ranked based on their probabilities computed by the acoustic and n-gram LMs. The higher value of hypothesis's probability, the higher the hypothesis's position in N-best list. Each hypothesis's probability is further calculated using the LSTM language model (LSTM LM), or in the general case, the LSTM LM interpolated with the n-gram LM. Subsequently, the probabilities computed by the n-gram LM are replaced with new values computed by the neural (or interpolated) LM and combined with AM score. Then *argmax* is computed over hypothesis in N-best list according to Equation (1). This results in a re-ranking of the hypotheses by their value of probability computed by NN-based LM, and the hypothesis with the highest probability (top ranked hypothesis) is selected as the best hypothesis.

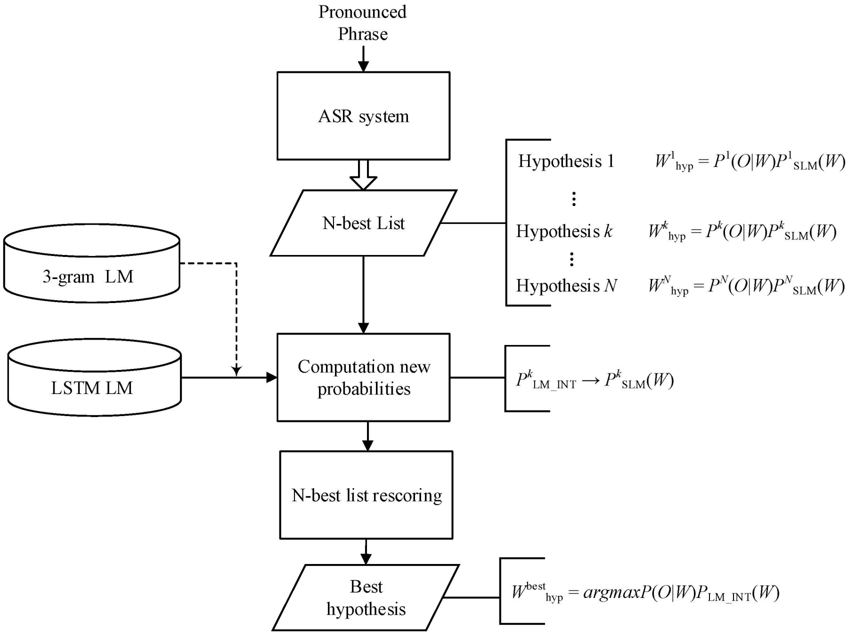

Pronounced Phrase
ASR system
N-best List
Hypothesis 1 $W^1{}_{\text{hyp}} = P^1(O|W)P^1{}_{\text{SLM}}(W)$
Hypothesis k $W^k{}_{\text{hyp}} = P^k(O|W)P^k{}_{\text{SLM}}(W)$
Hypothesis N $W^N{}_{\text{hyp}} = P^N(O|W)P^N{}_{\text{SLM}}(W)$
3-gram LM
LSTM LM
Computation new probabilities
$P^k{}_{\text{LM\_INT}} \rightarrow P^k{}_{\text{SLM}}(W)$
N-best list rescoring
Best hypothesis
$W^{\text{best}}{}_{\text{hyp}} = argmaxP(O|W)P_{\text{LM\_INT}}(W)$

Figure 4. The scheme of N-best list rescoring.

5. Results of Speech Recognition Experiments

Speech recognition was performed using Kaldi's decoder based on weighted finite-state transducers (WFST). Kaldi computes HCLG graphs created by composing other graphs: H∘ (C∘ (L∘G)), where G is the language model, L is the lexicon, C represents the context-dependency, and H contains the HMM definition [50].

The performance of the developed system was evaluated using WER, which is determined by aligning the reference transcription with the recognized sequence of words using the Levenshtein distance algorithm. The WER is computed as follows [64]:

$$WER = \frac{S + D + I}{T} \cdot 100\%, \tag{11}$$

where T is the number of words in reference transcription, and S, D, and I are the number of substituted, deleted, and inserted word respectively.

Evaluation of decoding speed was performed using Real Time Factor (RTF) [65]:

$$RTF = \frac{P}{J}, \tag{12}$$

where P is the time taken by the system to process the input and J is speech input duration.

LMs are evaluated in terms of perplexity. Perplexity can be thought as average number of equally probable words following any given words, and it is computed as it is shown in [63]:

$$PP = \sqrt[N]{\frac{1}{P(w_1, w_2, \ldots, w_N)}} \tag{13}$$

Perplexity should be computed using unseen text data. In this study, transcriptions of the test portion of the speech corpus were employed for this purpose.

At first experiments were performed using 3-gram language model. The value of perplexity of 3-gram model was 4030. Out-of-vocabulary (OOV) rate (the number of words from the test corpus that were absent in the vocabulary) was 5%. WER obtained with GMM/HMM and DNN/HMM AMs trained on not augmented data are presented in Table 4. Application of DNN/HMM models results in relative WER reduction of 39% compared to the triphone model and 30% compared to the fMLLR model.

Table 4. Experimental results obtained using not augmented data.

Type of AM	WER (%)
Triphone	44.57
LDA + MLLT	41.71
fMLLR	38.78
DNN/HMM	27.28

The results obtained with DNN/HMM models trained on data with different augmentation methods are presented in Table 5 as well as in Figure 5.

Table 5. Experimental results obtained using different types of augmentation.

Training Data	Data Duration	WER (%)
Natural	3 h 8 min	27.28
Natural + SpecAugment	3 h 8 min	27.48
Natural + Pitch Augmentation	6 h 16 min	27.44
Natural + Pitch Augmentation + SpecAugment	6 h 16 min	28.59
Natural + Tempo Augmentation	6 h 24 min	**25.57**
Natural + Tempo Augmentation + SpecAugment	6 h 24 min	27.64
Natural + Pitch + Tempo Augmentation	9 h 32 min	25.77
Natural + Pitch + Tempo Augmentation + SpecAugment	9 h 32 min	27.48
Natural + Combo Augmentation	6 h 24 min	26.80
Natural + Combo Augmentation + SpecAugment	6 h 24 min	26.37
Natural + Pitch + Tempo + Combo Augmentation	12 h 48 min	26.53
Natural + Pitch + Tempo + Combo Augmentation + SpecAugment	12 h 48 min	28.79

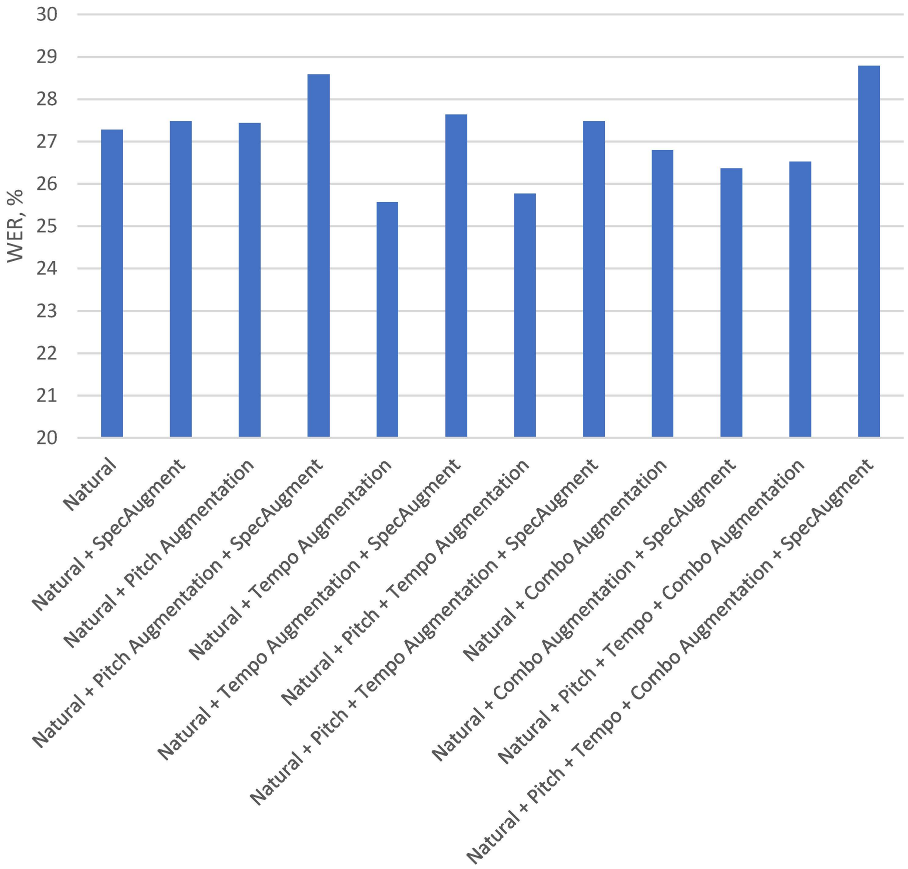


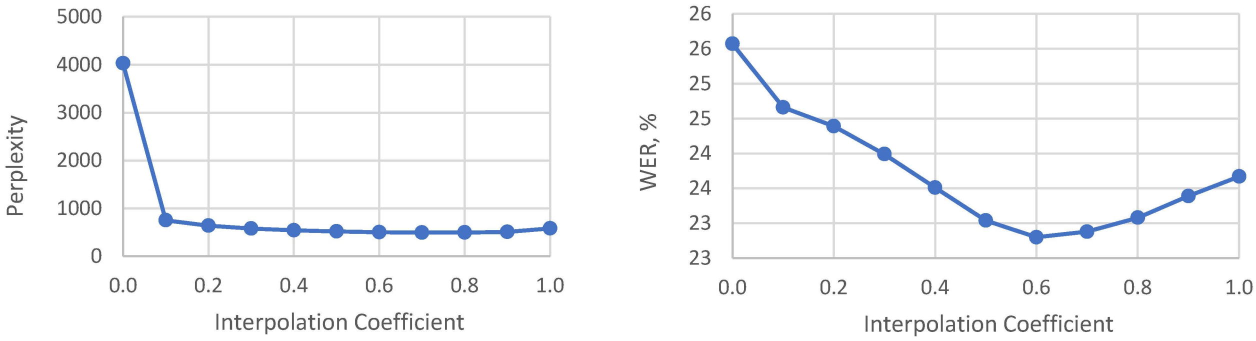
Figure 5. The Histogram of WER obtained using different types of augmentation.

As can be seen from Table 5 and Figure 5, the best result was obtained for data with tempo augmentation. The change in fundamental frequency did not lead to a decline in the WER value, possibly because the testing and training parts of the dataset were recorded with assistance of the same informants. On the whole, the use of SpecAugment did not result in an improvement in ASR performance. The decoding speed was measured at 0.1 RTF.

The AM trained on both unmodified data and data modified with tempo augmentation was used in the following experiments on 500-best list rescoring and choosing the best recognition hypothesis with the help of the NN-based model. Additionally, rescoring was performed using interpolated models. The results obtained from these experiments are presented in Table 6. An interpolation coefficient of 0 indicates that only the 3-gram model was utilized, while an interpolation coefficient of 1.0 implies that only the LSTM-best LM was employed. Furthermore, Table 6 presents the perplexity values of the NN-based LMs. Graphs depicting the correlation between perplexity values and word error rate (WER) based on the interpolation coefficient of the NN-based model are illustrated in Figure 6.

Table 6. Experimental results obtained with NN-based LMs (%).

Interpolation Coefficient	**0**	**0.5**	**0.6**	**0.7**	**1.0**
Perplexity	4030	519	504	**496**	582
WER	25.57	23.04	**22.80**	22.88	23.67



(**a**) (**b**)

Figure 6. The graphs of perplexity and WER versus interpolation coefficient of NN-based LM: (**a**) The graphs of perplexity versus interpolation coefficient. (**b**) The graphs of WER versus interpolation coefficient.

The table and graphs clearly reveal that LSTM-based language models exhibit significantly lower perplexity values compared to the 3-gram LM. Moreover, interpolation of LSTM-based model with 3-gram model gave additional reduction of perplexity. N-best list rescoring with the help of LSTM-based LM results in reducing the WER, and the application of interpolated models led to additional performance improvement. The best result was achieved with LSTM LM interpolated with n-gram LM with interpolation coefficient 0.6. In this case, WER was equal to 22.80.

6. General Discussion

Development of a speech recognition system for a low-resource language, as was emphasized in this paper, is a tricky task. That is why the results presented in this study should be compared with the results obtained for other low-resource languages, and not these of high-resource languages.

The authors of this paper believe that their results can be improved, but the aim of their work was to explore possibility to train the DNN-based ASR system on very limited data and research methods for data augmentation for improving performance of the system.

First of all, it is obvious that in order to improve the quality of speech recognition (as well as to increase the relevance of the collected corpus for linguistic research from a wider area), it is necessary to continue work on the collection and processing of speech and texts in the Karelian language. Mainly, it concerns speech data:

1. Imbalance of speakers: There are more female speakers than male in the corpus; in addition, the corpus contains recordings mainly of middle-aged and older speakers. This situation is caused, apparently, by the sociolinguistic realities in Karelia: the number of speakers among young people is quite limited [66,67].
2. Imbalance of the speakers regarding the number of utterances: It is necessary to achieve an equal representation of speakers in the collected data.

As can be seen from the presented paper, the standard speech recognition system, which implies the use of several modules—acoustic, linguistic and lexical—generally works for low-resource languages. As Table 6 illustrates (Section 5), utilizing an LSTM-based LM for N-best list rescoring allows an effective reduction in WER. The introduction of interpolated models further enhanced performance as well. The most favorable outcome was obtained by combining the LSTM LM with the n-gram LM using an interpolation coefficient of 0.6. As a result, the WER reached a value of 22.80%.

It should be recognized that the results of augmentation remained not completely clear. On the one hand, it is clear that data augmentation helped to solve the problem of data scarcity; on the other hand, the features of the data do not allow an unequivocal answer to the question of which type of augmentation is the most promising. It is evident from Table 5 (Section 5), however, that the best results were obtained when utilizing data with tempo augmentation. It is worth mentioning that altering the fundamental frequency did not lead to a decrease in WER. This could be attributed to the fact that both the testing and training portions of the dataset were recorded with assistance of the same speakers. Furthermore, the implementation of SpecAugment did not result in a reduction in the WER.

The Karelian data presents several characteristics that contribute to an increase in recognition errors. Firstly, in Karelian spontaneous speech there are many loanwords from the Russian language and dialect-specific words. However, texts are mostly literarily processed, resulting in the limited presence of such words. This results in increasing the OOV rate and degradation of ASR performance as a consequence. Moreover, the pronunciation of proper names typically follows Russian phonetic rules; therefore, proper names were often recognized incorrectly. Furthermore, Karelian, being an agglutinative language, constructs words by attaching affixes to a stem. As a result, numerous word-forms have very similar phonemic representation, differing only in endings (affixes). In spontaneous speech, endings of words are often reduced, and this causes the ASR system to produce incorrect word-forms where the stem is accurate.

Thus, this study allows framing a set of new questions:

1. How would the standard ASR system used in this research perform if enriched with transfer-learning methods?
2. How will improvement of data quality (speakers' age, gender, and number) affect the results in reference to augmentation method used in this study?
3. How will improvement of data quality (balancing the number of utterances spoken by each speaker) affect the results in reference to augmentation method used in this study?

These questions will form the guideline for future works within the framework of this project.

7. Conclusions

This paper presents an automatic speech recognition system for Livvi-Karelian. The biggest challenge the authors met was the scarcity of Karelian data. In order to increase the accuracy of the system, speech data was augmented, and NN-based LM as well as NN-based LM interpolated with 3-gram LM were used for N-best list rescoring. The WER value obtained during the experiments was 22.80%, which can be considered as a good result for a low-resource language. For the purpose of comparison, the results obtained by other studies for different low-resourced languages are presented in Table 7.

Table 7. Comparison of the results obtained for other low-resourced ASR system.

Related Work	Methods	Training Dataset	WER, %
[25]	GMM/HMM, DNN/HMM, TDNN/HMM, TDNN + LSTM/HMM	22 h of Sinhala speech	35.87~42.64
[26]	TDNN-F/HMM in unilingual and multilingual training	71 h of Kurmanji Kurdish, 101 h of Cree, 78 h of Inuktut	48~69.6
[27]	CNN-TDNN-BLSTM/HMM, TDNN-F/HMM	17.55 of Somali speech	49.59~53.75
[68]	TDNN/HMM + transfer learning	20 h of Amharic	24.50~28.48
[69]	TDNN-F/HMM	20–29 h of Amharic, Tigrigna, Oromo, and Wolaytta	32~23

It is important to highlight that a direct comparison of ASR results across different languages is impossible due to a dependency of the results not only on the used models, but on training and test data as well. Nevertheless, a conclusion can be drawn that the obtained results are at the level of word results for other low-resourced languages.

Another important outcome of this study was the collection of a training dataset that includes speech recordings in Karelian, their transcriptions, and a text corpus. In the presented study, the dataset was used to train language and acoustic models, but it can be used in other studies on the Karelian language regarding natural language processing tasks.

Despite all the results achieved, there is a need for further research and improvement of the developed system. For example, the problem of code-switching remained outside the scope of this study. In general, increasing the training dataset will significantly improve the performance of the developed system. In the future, it is planned to apply the methods of transfer learning to acoustic model training and to use other DNN-based approaches for acoustic and language models training as well as to investigate the methods for addressing code-switching phenomena.

Author Contributions: Conceptualization, I.K. (Irina Kipyatkova) and I.K. (Ildar Kagirov); methodology, I.K. (Irina Kipyatkova); software, I.K. (Irina Kipyatkova); validation, I.K. (Irina Kipyatkova); investigation, I.K. (Irina Kipyatkova) and I.K. (Ildar Kagirov); resources, I.K. (Irina Kipyatkova) and I.K. (Ildar Kagirov); writing—original draft preparation, I.K. (Irina Kipyatkova) and I.K. (Ildar Kagirov); writing—review and editing, I.K. (Irina Kipyatkova) and I.K. (Ildar Kagirov); visualization, I.K. (Irina Kipyatkova); supervision, I.K. (Irina Kipyatkova); project administration, I.K. (Irina Kipyatkova); funding acquisition, I.K. (Irina Kipyatkova). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Russian Science Foundation, grant number № 22-21-00843, https://rscf.ru/en/project/22-21-00843/, accessed on 28 June 2023.

Data Availability Statement: In this work, the following large-scale publicly available dataset was used: open corpus of Veps and Karelian languages "VepKar"—http://dictorpus.krc.karelia.ru/ru accessed on 28 June 2023.

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

Abbreviations

The following abbreviations are used in this manuscript:

AM	Acoustic Model
ASR	Automatic Speech Recognition
C-VAE	Conditional Variational Autoencoder
CE	Cross-Entropy
CNN	Convolutional Neural Network
DCT	Discrete Cosine Transform
DNN	Deep Neural Networks
fMLLR	Feature Space Maximum Likelihood Linear Regression
FST	Finite-State Transducer
GAN	Generative Adversarial Networks
GMM	Gaussian Mixture Models
GPU	Graphics Processing Unit
HMM	Hidden Markov Models
LDA	Linear Discriminant Analysis
LF-MMI	Lattice-Free Maximum Mutual Information
MMI	Maximum Mutual Information
LM	Language Model
LSTM	Long Short-Term Memory
LPC	Linear Predictive Coding
MLLT	Maximum Likelihood Linear Transform
MFCC	Mel-Frequency Cepstral Coefficients
NN	Neural Network
OOV	Out-of-vocabulary
ReLu	Rectified Linear Unit
SAT	Speaker Adaptive Training
SLM	Statistical LM
SRILM	SRI Language Modeling Toolkit
SVD	Singular Value Decomposition
TDNN	Time Delay Neural Network
TDNN-F	Factorized Time Delay Neural Network
VTLP	Vocal Tract Length Perturbation
WER	Word Error Rate
WFST	Weighted Finite-State Transducers

References

1. Nassif, A.B.; Shahin, I.; Attili, I.; Azzeh, M.; Shaalan, K. Speech recognition using deep neural networks: A systematic review. *IEEE Access* **2019**, *7*, 19143–19165. [CrossRef]
2. Ryumin, D.; Ivanko, D.; Ryumina, E. Audio-Visual Speech and Gesture Recognition by Sensors of Mobile Devices. *Sensors* **2023**, *23*, 2284. [CrossRef] [PubMed]
3. Ivanko, D.; Ryumin, D.; Karpov, A. A Review of Recent Advances on Deep Learning Methods for Audio-Visual Speech Recognition. *Mathematics* **2023**, *11*, 2665. [CrossRef]
4. Birjali, M.; Kasri, M.; Beni-Hssane, A. A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowl.-Based Syst.* **2021**, *226*, 107134. [CrossRef]
5. Stahlberg, F. Neural machine translation: A review. *J. Artif. Intell. Res.* **2020**, *69*, 343–418. [CrossRef]
6. Baumann, P.; Pierrehumbert, J. Using Resource-Rich Languages to Improve Morphological Analysis of Under-Resourced Languages. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, 26–31 May 2014.
7. Magueresse, A.; Carles, V.; Heetderks, E. Low-resource Languages: A review of past work and future challenges. *arXiv* **2006**, arXiv:2006.07264v1. [CrossRef]
8. Joshi, P.; Santy, S.; Budhiraja, A.; Bali, K.; Choudhury, M. The State and Fate of Linguistic Diversity and Inclusion in the NLP World. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL'20), Seattle, WA, USA, 5–10 July 2020.
9. Bender, E.M. On achieving and evaluating language-independence in NLP. *Linguist. Issues Lang. Technol.* **2011**, *6*, 1–26. [CrossRef]
10. Ponti, E.M.; O'Horan, H.; Berzak, Y.; Vulic, I.; Reichart, R.; Poibeau, T.; Shutova, E.; Korhonen, A. Modeling language variation and universals: A survey on typological linguistics for natural language processing. *Comput. Linguist.* **2019**, *45*, 559–601. [CrossRef]
11. Laptev, A.; Andrusenko, A.; Podluzhny, I.; Mitrofanov, A.; Medennikov, I.; Matveev, Y. Dynamic acoustic unit augmentation with BPE-dropout for low-resource end-to-end speech recognition. *Sensors* **2021**, *21*, 3063. [CrossRef] [PubMed]

12. Andrusenko, A.; Nasretdinov, R.; Romanenko, A. UCONV-conformer: High Reduction of Input Sequence Length for End-to-End Speech Recognition. In Proceedings of the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'23), Rhodes Island, Greece, 4–10 June 2023.
13. Sarhimaa, A. Karelian. In *The Oxford Guide to the Uralic Languages*; Bakró-Nagy, M., Laakso, J., Skribnik, E., Eds.; Oxford Academic: Oxford, UK, 2022; pp. 269–290.
14. Laakso, J. The Finnic languages: Typology and contact. In *The Circum-Baltic Languages. Vol. I: Past and Present*; Dahl, Ö., Koptjevskaja-Tamm, M., Eds.; John Benjamins Publishing Company: Amsterdam, The Netherlands, 2001; pp. 179–212.
15. Novak, I.; Penttonen, M.; Ruuskanen, A.; Siilin, L. *Karelian in Grammars: A Study of Phonetic and Morphological Variation*; Scientific Electronic Edition; KarRC RAS Publications: Petrozavodsk, Russia, 2022. Available online: http://resources.krc.karelia.ru/illh/doc/knigi_stati/karelian_in_grammar.pdf (accessed on 28 June 2023).
16. Krauwer, S. The Basic Language Resource Kit (BLARK) as the First Milestone for the Language Resources Roadmap. In Proceedings of the International Workshop on Speech and Computer (SPECOM-2003), Moscow, Russia, 27–29 October 2003.
17. Berment, V. Méthodes pour Informatiser des Langues et des Groupes de Langues "Peu Dotées". Ph.D. Thesis, Université Joseph-Fourier, Grenoble, France, 18 May 2004.
18. Cieri, C.; Maxwell, M.; Strassel, S.; Tracey, J. Selection Criteria for Low Resource Language Programs. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, 23–28 May 2016.
19. Romanenko, A.N. Robust Speech Recognition for Low-Resource Languages. Ph.D. Thesis, Universität Ulm, Ulm, Germany, ITMO University, St. Petersburg, Russia, 23 November 2020.
20. Besacier, L.; Barnard, E.; Karpov, A.; Schultz, T. Automatic speech recognition for under-resourced languages: A survey. *Speech Commun.* **2014**, *56*, 85–100. [CrossRef]
21. Bhatt, S.; Jain, A.; Dev, A. Acoustic modeling in speech recognition: A systematic review. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 397–412. [CrossRef]
22. Kipyatkova, I.S.; Kagirov, I.A. Analytical review of methods for solving data scarcity issues regarding elaboration of automatic speech recognition systems for low-resource languages. *Inform. Autom.* **2022**, *21*, 678–709. (In Russian) [CrossRef]
23. Yu, D.; Deng, L. *Automatic Speech Recognition—A Deep Learning Approach*; Springer: London, UK, 2015; 322p.
24. Markovnikov, M.; Kipyatkova, I. An analytic survey of end-to-end speech recognition systems. *SPIIRAS Proc.* **2018**, *58*, 77–110. (In Russian) [CrossRef]
25. Karunathilaka, H.; Welgama, V.; Nadungodage, T.; Weerasinghe, R. Low-Resource Sinhala Speech Recognition Using Deep Learning. In Proceedings of the 2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer2020), Colombo, Sri Lanka, 5–6 November 2020.
26. Gupta, V.; Boulianne, G. Progress in Multilingual Speech Recognition for Low Resource Languages Kurmanji Kurdish, Cree and Inuktut. In Proceedings of the 13th Conference on Language Resources and Evaluation (LREC'22), Marseille, France, 20–25 June 2022.
27. Biswas, A.; Menon, R.; van der Westhuizen, E.; Niesler, T. Improved Low-Resource Somali Speech Recognition by Semi-Supervised Acoustic and Language Model Training. In Proceedings of the 20th Annual Conference of the International Speech Communication Association (INTERSPEECH'19), Graz, Austria, 15–19 September 2019.
28. Obukhov, D.S. Speech recognition system for Russian language telephone speech. *Large-Scale Syst. Control.* **2021**, *89*, 106–122. (In Russian)
29. Kipyatkova, I. Improving Russian LVCSR Using Deep Neural Networks for Acoustic and Language Modeling. In Proceedings of the 20th International Conference on Speech and Computer (SPECOM'18), Leipzig, Germany, 18–22 September 2018.
30. Oneata, D.; Cucu, H. Improving Multimodal Speech Recognition by Data Augmentation and Speech Representations. In Proceedings of the 6th Multimodal Learning and Applications Workshop (MULA'23), in Conjunction with the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2023 (CVPR'23), Vancouver, BC, Canada, 18 June 2023.
31. Ko, T.; Peddinti, V.; Povey, D.; Khudanpur, S. Audio Augmentation for Speech Recognition. In Proceedings of the 16th Annual Conference of the International Speech Communication Association (INTERSPEECH'15), Dresden, Germany, 6–10 September 2015.
32. Rebai, I.; BenAyed, Y.; Mahdi, W.; Lorré, J.P. Improving speech recognition using data augmentation and acoustic model fusion. *Procedia Comput. Sci.* **2017**, *112*, 316–322. [CrossRef]
33. Hartmann, W.; Ng, T.; Hsiao, R.; Tsakalidis, S.; Schwartz, R. Two-Stage Data Augmentation for Low-Resourced Speech Recognition. In Proceedings of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH'16), San Francisco, CA, USA, 8–12 September 2016.
34. Jin, Z.; Finkelstein, A.; DiVerdi, S.; Lu, J.; Mysore, G.J. Cute: A Concatenative Method for Voice Conversion Using Exemplar-Based Unit Selection. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'16), Shanghai, China, 20–25 March 2016.
35. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems (NIPS'14), Montreal, QC, Canada, 8–13 December 2014.

36. Hsu, C.-C.; Hwang, H.-T.; Wu, Y.-C.; Tsao, Y.; Wang, H. Voice Conversion from Unaligned Corpora Using Variational Autoencoding Wasserstein Generative Adversarial Networks. In Proceedings of the 18th Annual Conference of the International Speech Communication Association (INTERSPEECH'17), Stockholm, Sweden, 20–24 August 2017.
37. Kameoka, H.; Kaneko, T.; Tanaka, K.; Hojo, N. StarGAN-VC: Non-Parallel Many-to-Many Voice Conversion Using Star Generative Adversarial Networks. In Proceedings of the 2018 IEEE Spoken Language Technology Workshop (SLT'18), Athens, Greece, 18–21 December 2018.
38. 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 and Devices (SSD'19), Istanbul, Turkey, 21–24 March 2019.
39. Meng, L.; Xu, J.; Tan, X.; Wang, J.; Qin, T.; Xu, B. MixSpeech: Data Augmentation for Low-Resource Automatic Speech Recognition. In Proceedings of the 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'21), Toronto, ON, Canada, 6–11 June 2021.
40. Wei, S.; Xu, K.; Wang, D.; Liao, F.; Wang, H.; Kong, Q. Sample Mixed-Based Data Augmentation for Domestic Audio Tagging. In Proceedings of the 2018 Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE'18), Surrey, UK, 19–20 November 2018.
41. Dong, Z.; Hu, Q.; Guo, Y.; Cordy, M.; Papadakis, M.; Le Traon, Y.; Zhao, J. Enhancing code classification by Mixup-based data augmentation. *arXiv* **2022**, arXiv:2210.03003v2. [CrossRef]
42. Park, D.S.; Chan, W.; Zhang, Y.; Chiu Ch-Ch Zoph, B.; Cubuk, E.D.; Le, Q.V. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In Proceedings of the 20th Annual Conference of the International Speech Communication Association (INTERSPEECH'19), Graz, Austria, 15–19 September 2019.
43. Sarkar, A.K.; Tan, Z.-H. Vocal tract length perturbation for text-dependent speaker verification with autoregressive prediction coding. *IEEE Signal Process. Lett.* **2021**, *28*, 364–368. [CrossRef]
44. Mertes, S.; Baird, A.; Shiller, D.; Shuller, B.W.; André, E. An Evolutionary-Based Generative Approach for Audio Data Augmentation. In Proceedings of the 2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP'20), Tampere, Finland, 21–24 September 2020.
45. Donahue, C.; McAuley, J.; Puckette, M. Synthesizing Audio with Generative Adversarial Networks. In Proceedings of the 7th International Conference on Learning Representations (ICLR'19), New Orleans, LA, USA, 6–9 May 2018.
46. Shen, J.; Pang, R.; Weiss, R.; Schuster, M.; Jaitly, N.; Yang, Z.; Chen, Z.; Zhang, Y.; Wang, Y.; Skerry-Ryan, R.J.; et al. Natural TTS Synthesis by Conditioning WaveNet on Mel-Spectrogram Predictions. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'18), Calgary, AB, Canada, 15–20 April 2018.
47. Dua, M.; Kadyan, V.; Banthia, N.; Bansal, A.; Agarwal, T. Spectral warping and data augmentation for low resource language ASR system under mismatched conditions. *Appl. Acoust.* **2022**, *190*, 108643. [CrossRef]
48. Du, C.; Yu, K. Speaker Augmentation for Low Resource Speech Recognition. In Proceedings of the 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'20), Virtual Barcelona, Spain, 4–8 May 2020.
49. Bagchi, D.; Wotherspoon, S.; Jiang, Z.; Muthukumar, P. Speech synthesis as augmentation for low-resource ASR. *arXiv* **2012**, arXiv:2012.13004v1. [CrossRef]
50. Povey, D.; Ghoshal, A.; Boulianne, G.; Burget, L.; Glembek, O.; Goel, N.; Hannemann, M.; Motlíček, P.; Qian, Y.; Schwarz, P.; et al. The Kaldi Speech Recognition Toolkit. In Proceedings of the 2011 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU'11), Waikoloa, HI, USA, 11–15 December 2011.
51. Nirvi, R.E. *Suistamon Keskusmurteen Vokalismi*; SKS: Helsinki, Finnland, 1932; 96p. (In Finnish)
52. Turunen, A. *Lyydiläismurteiden Äännehistoria I. Konsonantit*; SUS: Helsinki, Finnland, 1946; XXI + 338p. (In Finnish)
53. Alumäe, T.; Vohandu, L. Limited-vocabulary Estonian continuous speech recognition system using Hidden Markov Models. *Informatica* **2004**, *15*, 303–314. [CrossRef]
54. Nakai, S.; Kunnari, S.; Turk, A.; Suomi, K.; Ylitalo, R. Utterance-final lengthening and quantity in Northern Finnish. *J. Phon.* **2009**, *37*, 29–45. [CrossRef]
55. Povey, D.; Cheng, G.; Wang, Y.; Li, K.; Xu, H.; Yarmohamadi, M.; Khudanpur, S. Semi-Orthogonal Low-Rank Matrix Factorization for Deep Neural Networks. In Proceedings of the 19th Annual Conference of the International Speech Communication Association (INTERSPEECH'18), Hyderabad, India, 2–6 September 2018.
56. Povey, D.; Peddinti, V.; Galvez, D.; Ghahremani, P.; Manohar, V.; Na, X.; Wang, Y.; Khudanpur, S. Purely Sequence-Trained Neural Networks for ASR Based on Lattice-Free MMI. In Proceedings of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH'16), San Francisco, CA, USA, 8–12 September 2016.
57. Madikeri, S.R.; Khonglah, B.K.; Tong, S.; Motlicek, P.; Bourlard, H.; Povey, D. Lattice-Free Maximum Mutual Information Training of Multilingual Speech Recognition Systems. In Proceedings of the 21st Annual Conference of the International Speech Communication Association (INTERSPEECH'20), Shanghai, China, 25–29 October 2020.
58. Yang, X.; Li, J.; Zhou, X. A novel pyramidal-FSMN architecture with lattice-free MMI for speech recognition. *arXiv* **2019**, arXiv:1810.11352v2. [CrossRef]
59. Rabiner, L.; Juang, B.-H. *Fundamentals of Speech Recognition*; PTR Prentice Hall: Hoboken, NJ, USA, 1993; 507p.
60. Stolcke, A.; Zheng, J.; Wang, W.; Abrash, V. SRILM at Sixteen: Update and Outlook. In Proceedings of the 2011 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU'11), Waikoloa, HI, USA, 11–15 December 2011.
61. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]

62. Enarvi, S.; Kurimo, M. TheanoLM—An Extensible Toolkit for Neural Network Language Modeling. In Proceedings of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH'16), San Francisco, CA, USA, 8–12 September 2016.
63. Moore, G.L. Adaptive Statistical Class-Based Language Modelling. Ph.D. Dissertation, Cambridge University, Cambridge, UK, October 2001.
64. Zechner, K.; Waibel, A. Minimizing Word Error Rate in Textual Summaries of Spoken Language. In Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics (ANLP 2000), Seattle, WA, USA, 29 April–4 May 2000.
65. Malik, M.; Malik, M.K.; Mehmood, K.; Makhdoom, I. Automatic speech recognition: A survey. *Multimed. Tools Appl.* **2021**, *80*, 9411–9457. [CrossRef]
66. Karjalainen, H.; Ulriikka, P.; Riho, G.; Svetlana, K. *Karelian in Russia: ELDIA Case-Specific Report*; (Studies in European Language Diversity 26); Reetta, T., Anneli, S., Eva, K., Eds.; Research Consortium ELDIA: Mainz, Germany, 2013. Available online: https://phaidra.univie.ac.at/detail/o:314612 (accessed on 28 June 2023).
67. Kovaleva, S.V.; Rodionova, A.P. *Traditional and Innovative in the Vocabulary and Grammar of Karelian*; (based on a sociolinguistic research); KarNC RAN Publications: Petrozavodsk, Russia, 2011; 138p. (In Russian)
68. Woldemariam, Y. Transfer Learning for Less-Resourced Semitic Languages Speech Recognition: The Case of Amharic. 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.
69. Abate, S.T.; Tachbelie, M.Y.; Schultz, T. Deep neural networks based automatic speech recognition for four Ethiopian languages. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.