UNIVERSITY OF TARTU
Institute of Computer Science
Software Engineering Curriculum

Karel Roots

Semi-Supervised Automatic Speech Recognition for Low Resource Languages

Master's Thesis (30 ECTS)

Supervisor Mark Fišel, PhD

Tartu 2022

Pool-juhendamisega automaatne kõnetuvastus väheste ressurssidega keeltele

Lühikokkuvõte:

Automaatne kõnetuvastus on arvutiteaduse valdkond, mis on suunitletud kõne tuvastamise ja tekstiks ümbertöötlemise meetodite ja tehnoloogiate väljatöötamisele. Kõnetuvastus leiab laialdaselt rakendust erinevates inimese-arvuti liidestes. See tehnoloogia võimaldab näiteks puuetega inimestel mõista transkribeeritud kõnet ja aitab neil juhtida arvutisüsteeme kasutades kõnel põhinevat sisendit.

Üks peamisi väljakutseid kõnetuvastuse mudelite loomisel väheste ressurssidega keeltele nagu eesti keel, on piisava koguse andmete puudus, mis on tarvilik närvivõrkudel põhinevate masinõppe mudelite treenimiseks. Hiljuti aga on välja töötatud mitmekeelsed pool-juhendamisega masinõppe mudelid, mis kasutavad juhendamata eeltreenimist suurel kogusel märgendamata andmestikul ja juhendatud peen töötlemist väiksel kogusel märgendatud andmetel. Need mudelid on demonstreerinud kõrget potentsiaali väheste ressurssidega keelte kõnetuvastuse parendamiseks.

Käesolevas töös on uurimise all wav2vec 2.0 nimeline kõnetuvastuse masinõppe mudeli arhitektuur. Töös implementeeritakse ühekeelne masinõppe mudel, mis on eeltreenitud ja peentöödeldud vaid eestikeelsetel andmetel ning mitmekeelne masinõppe mudel, mis on eeltreenitud eesti- ja ingliskeelsetel andmetel ning peentöödeldud eestikeelsetel andmetel. Mudeleid hinnatakse eestikeelsetel test andmetel ning nende tulemusi võrreldakse kasutades sõna ja tähemärgi vea määra.

Eksperimentide tulemused näitavad, et mitmekeelne eeltreenimine saavutab eestikeelsetel andmetel sõna vea määra 12.1% ja tähemärgi vea määra 5%. Samal ajal ühekeelne eeltreenimine saavutab sõna vea määra 26.9% ja tähemärgi vea määra 5.9%. Need tulemused esindavad mitmekeelse mudeli jaoks 53.6% madalamat sõna vea määra ja 15.3% madalamat märgi vea määra kui ühekeelsel mudelil ning toovad esile pool-juhendatud mitmekeelse mudeli potentsiaali parandada väheste ressurssidega keelte kõnetuvastust kasutades märgendamata andmeid.

Võtmesõnad: Automaatne kõnetuvastus, pool-juhendamisega õpe, väheste ressurssidega keeled

CERCS: P176 Tehisintellekt

Semi-Supervised Automatic Speech Recognition for Low Resource Languages

Abstract:

Automatic speech recognition (ASR) is a field of computer science that focuses on the development of methods and technologies that recognise and transform spoken language to text. It has a wide variety of applications in the field of human-computer interaction. It can be used to help people with disabilities to understand transcribed speech and control computer systems with speech-based input.

Presently, the major challenge in developing speech recognition models for low resource languages such as the Estonian language is the lack of large amounts of data that is needed for neural network based machine learning models. Recently however, multilingual self-supervised pre-training of machine learning models on large datasets and fine-tuning on small amounts of labelled data of the target language has shown great promise in improving speech recognition for low resource languages.

In this thesis, the wav2vec 2.0 machine learning model architecture for speech recognition is explored. We evaluate and compare a monolingual model exclusively pre-trained on unlabelled data and fine-tuned and evaluated on labelled data of the Estonian language to a multilingual model that is pre-trained on unlabelled English and Estonian data and fine-tuned and tested on labelled Estonian data.

The performed experiments reveal that the multilingual pre-training achieves an average word error rate of 12.1% and character error rate of 5% compared to 26.9% and 5.9% for the respective metrics on the monolingual model evaluation. These results represent a 53.6% decrease in word errors and 15.3% decrease in character errors for the multilingual model and highlight the potential of improving speech recognition of low resource languages by means of self-supervised learning on multilingual unlabelled speech data.

Keywords: Automatic speech recognition (ASR), semi-supervised learning, low resource languages

CERCS: P176 Artificial Intelligence

Contents

1 Introduction

Automatic speech recognition (ASR) is a field of computer science that focuses on developing methodologies and technologies to recognise and transform spoken language to text. Interest in ASR research has significantly increased over recent decades due to speech being a natural way of communication between people and technological advancements enabling high accuracy in speech recognition. ASR research began with simple systems capable of recognising limited sounds, but advancements in the field have produced complex systems capable of efficiently processing natural language in many different languages [1].

ASR has a wide variety of applications in human-computer interaction (HCI), where it enables humans to communicate commands which can be recognised and processed by the computer. In addition to a wide range of applications in smart home and entertainment systems, speech recognition has also found significant use in assisting people with disabilities as it allows transcribing speech into closed captions and operating computer systems without physical input [2].

1.1 The Problem

Recently, the fast-evolving field of machine learning has found extensive use for performing speech recognition tasks. The machine learning models for speech recognition are commonly trained on large monolingual datasets [3]. Models that have been trained on data from a single source language are generally only capable of transcribing speech in the same language that was used to train the model. Moreover, monolingual speech recognition models typically require a large amount of pre-transcribed speech as training data which normally is collected through manual transcription of the speech audio. As such, the major challenge for speech recognition of languages that are spoken by a small number of people is the lack of sufficient data for training the models.

Several recent studies have explored a novel approach for semi-supervised pre-training of speech recognition models [3, 4]. The semi-supervised models are trained on large amounts of unlabelled multilingual data instead of the more difficult to obtain labelled data of a specific language. However, with the semi-supervised approach, a small amount of

labelled data in the target language is still necessary for fine-tuning the model for more accurate transcriptions. Multilingual models trained in the semi-supervised approach have been shown to outperform the monolingual models that are exclusively trained on labelled data of the target language in a supervised manner [5, 6]. This type of cross-lingual learning can be used to build models that leverage data from different languages. Consequently, pre-training the speech recognition models on multilingual data by generalising across languages could therefore improve the performance of transcribing languages that have very little labelled speech data available.

In recent research, multilingually pre-trained speech recognition models such as XLSR [7], XLS-R [8] and XLS-T [9] have shown notable advancements in multilingual speech comprehension. These novel approaches are a promising insight into the development of universal speech recognition technology that improves the performance of low resource languages by using data from high-resource languages. Additionally, in this case it is only necessary to manage a single multilingual model instead of many monolingual models when performing speech recognition tasks for an application that has to perform well in multiple different languages.

1.2 Purpose and Overview

The purpose of this thesis is to verify the viability of semi-supervised learning for speech recognition of low resource languages such as the Estonian language. The goal is to pre-train a speech recognition model on a large unlabelled multilingual dataset and fine-tune it on a small labelled dataset of the target Estonian language. The multilingual model will be verified to outperform a second monolingual model with identical architecture that is trained exclusively on unlabelled data of the Estonian language. The experiments aim to determine if self-supervised multilingual speech recognition models could be used to improve speech recognition performance compared to monolingual self-supervised models in the Estonian language.

The thesis is divided into four main chapters. After the initial introductory chapter, the second chapter gives a comprehensive overview of the speech recognition problem space and current methodology and applications for performing automatic speech recognition. It introduces common speech data preprocessing methods, explores multiple different

approaches for speech to text transcription, gives an overview of the current state-of-the-art approaches, and investigates common applications for speech recognition.

The third chapter describes the overall experimental approach and datasets used for the model pre-training and fine-tuning steps. It presents the overall architecture of the chosen neural network-based speech recognition model, describes the multilingual datasets used for training and evaluation, gives an explanation of the data preprocessing methodology, and details the training procedure for pre-training and fine-tuning of the multilingual models.

The fourth chapter gives an overview of how the training of the models was performed. It also describes the model evaluation procedure, the process of validating the results, the explanation of measuring the statistical significance of the results, the results and analysis of the model evaluations, and finally, potential future work proposals.

The final chapter summarises the purpose and results of the work and gives a reflective evaluation of the goals accomplished.

2 Overview of the Problem

The following chapter gives an overview of the problem space of automatic speech recognition, including the purpose, methodology, and applications of speech recognition. The first subchapter briefly introduces automatic speech recognition and its two major objectives - human-human communication and human-computer interaction. In the second subchapter, the most common data preprocessing methods for speech recognition are detailed. In the third subchapter, a comprehensive overview of speech to text transcription methods is given. A survey of various practical applications for speech recognition is presented in the final subchapter.

2.1 Automatic Speech Recognition

Automatic speech recognition is a subfield of computational linguistics and computer science focused on developing computer-based methods and technologies for recognising and translating spoken language into human-readable text [10]. It has been researched for more than five decades and is considered a significant component for improved communication between humans and computers. Audrey, the first speech recognition system designed to recognise digits, was introduced already in 1952 by Bell labs researchers. By 2001, Google had presented the first Voice Search application that could make search queries by speaking to the ASR application [11].

In recent years, advancements in speech recognition technology have begun to change our daily lives and work, and speech has become one of the principal ways of interacting with different electronic devices and systems. The rapid increase of available computational power through the development of graphical processing units (GPUs) designed for machine learning and the rise of parallel computing has been a major driving force behind the increased viability of ASR. Modern complex and computation-intensive models have substantially reduced error rates in speech recognition systems. In addition, the availability of publicly available speech data has substantially improved with the rapidly increasing popularity and accessibility of the internet, which is a rich source of speech samples. Several projects exist

for collecting speech data from willing volunteers [12] and publicly available resources such as videos [13].

Finally, many new consumer devices focus on the mobility aspect of technology, and alternative input methods like speech are progressively becoming more favourable to the standard touch-based inputs. All the previously mentioned aspects are increasingly making ASR a viable option for interacting with our everyday technology.

2.1.1 Human-Human Communication

One way of improving communication between humans is with the assistance of speech recognition technology. Previously, people speaking different languages required human translators to understand each other, which is a substantial constraint on the availability of such communication. Recently, however, speech-to-speech translation systems have been developed that are capable of live translation of spoken words in different languages [14]. These systems can be integrated into various communication applications on our smartphones or computers and allow people to communicate with others in many different languages that they wouldn't understand otherwise.

In addition, speech technology can be used to transcribe speech to conveniently send text messages through instant messaging [15] or emails. It can also be used to transcribe and index lectures, speeches, and podcasts [16] for handily searching for information in the source audio material.

2.1.2 Human-Computer Interaction

Human-computer interaction is another field in which speech technology can be used to improve the user experience. In recent times, one of the more popular applications for HCI is voice search functionality, which can quickly search for information on the internet through speech.

Searching for information on mobile devices is also often paired with a personal digital assistant (PDA) such as the Apple Siri, Amazon Alexa or Google Assistant voice assistant software that can listen to command input through speech. PDAs often utilise

personalised interaction and past speech data of the user to improve the accurate identification of voice commands further.

Personalised speech recognition is also leading its way into our homes and lives with the rising popularity of living room interaction systems and in-vehicle information and entertainment systems. Systems like that enable interaction with the system through speech for querying information, playing music, or controlling intelligent home systems [17].

2.2 Speech Data Preprocessing

Data preprocessing is a crucial step for most machine learning algorithms. The raw data must usually be cleaned and converted into a suitable format for the machine learning model. In speech preprocessing, the most common techniques are data normalisation and audio signal resampling.

2.2.1 Normalisation

The most common normalisation approaches for raw speech audio are global feature standardisation and per-sample feature normalisation.

The purpose of global feature standardisation is to transform the data of each dimension with a global transformation such that the final vectors are in a similar range. When training deep neural networks (DNNs), standardising features enable using the same learning rate across all weight dimensions without sacrificing model performance.

The purpose of per-sample feature normalisation is to reduce the variability of the features used as input to the DNN. In speech recognition tasks this approach could reduce acoustic channel distortions [10].

2.2.2 Resampling

An analog audio signal is defined as a continuous representation of a signal over time with unlimited samples existing between any two points in time [18]. A sample in this context is

the value of the signal at a point in space and/or time. Sampling is a technique in signal processing used to reduce the continuous time signal to a discrete time signal.

For audio signals sampling is used to choose a number of samples per second from the raw audio signal to convert an analog sound wave to a sequence of samples. The purpose of this is to convert the continuous signal to a discrete signal so that it could be processed and stored more efficiently in computer memory. The sampling rate or frequency in this context is the number of samples selected per second [19].

In machine learning applications, the raw digital audio data is often resampled or downsampled to a lower sampling frequency. This can be useful, because using data with a lower sampling rate reduces the computational cost of training the computationally heavy machine learning algorithms for speech recognition.

2.3 Speech-to-Text Transcription

Speech-to-text transcription in natural language processing is a way to transform speech data into human-readable text. Speech data may be processed in the form of streamed or stored audio or video files. In a speech to text translation system, the objective is to analyse and convert raw speech audio data to the textual representation of the spoken words.

2.3.1 Neural Networks

In machine learning, a network consisting of artificial neurons is called an artificial neural network (ANN) [20]. ANNs are computational networks that roughly attempt to imitate the decision processes of nerve cells. Even though traditional computers can manage the same tasks that ANNs are capable of, the significance of ANNs is their capability to execute elementary operations like additions, multiplications, and logic operations, to solve complicated nonlinear or stochastic problems [21].

Many different types of ANNs exist, from simpler structures such as the feedforward and backpropagation neural network to complex architectures such as convolutional neural networks (CNNs) [28]. A neural network is called a deep neural network (DNN) when the

network architecture contains more than one hidden layer [22]. Recently, DNNs have become the principal approach used in current state-of-the-art speech recognition systems [10].

2.3.1.1 Recurrent Neural Networks

Recurrent neural networks (RNNs) are a class of ANNs where the network is composed of nodes of a temporal sequence-based directed or undirected graph. A commonly used type of RNN, the long short-term memory (LSTM) network, was invented in 1997 [23]. Due to their ability to handle temporal information of input data such as text, audio, and video, RNNs have become dominant for processing temporal sequences. They revolutionised the speech recognition domain around 2007 [24] when LSTM based models first outperformed traditional acoustic models used in speech recognition.

Based on multilayer feedforward neural networks [25], RNNs use the internal state or memory to process input sequences of different lengths, making them suitable for processing the continuous speech audio sequences for speech recognition. In feedforward networks, the input sequence history is represented by a context of N - 1 samples, limiting its ability to process long sequences of contextual data. The RNN however consists of multiple successive recurrent layers where history is represented by neurons of recurrent connections, which makes the history length unlimited.

RNNs have been used to achieve state-of-the-art results in image generation [26], speech recognition [27], machine translation [28] and video processing for applications such as deepfake detection [29] and emotion recognition [30].

2.3.1.2 Convolutional Neural Networks

Recently, convolutional neural networks (CNNs) have become popular for solving machine learning tasks in many different domains. Several current state-of-the-art machine learning models for speech recognition are based on the CNN architecture [3, 31]. The original inspiration for CNNs were biological processes. The connection patterns of the artificial neurons closely resemble the visual cortex organisational structure [22]. Kunihiko Fukushima proposed the first convolution-based neural network architecture when he presented the “neocognitron” structure in 1980 [32].

A CNN consists of one or more convolutional and subsampling layers. These layers can be followed by one or more fully connected hidden layers. This architecture uses input data in a two-dimensional format, such as audio signals or other similarly structured data. The speech audio data used in this thesis are two-dimensional because the signals have temporal (data points over time) and spatial (number of samples per second) dimensions. CNNs have been shown to be effective in image [33], natural language processing [34] and speech recognition [3, 31, 35].

The two main components of a CNN are the feature extractor, which contains multiple convolutions and pooling layers, and the trainable component, which consists of the fully connected multilayer perceptron [36]. The convolution layers are used to extract features from the input data automatically. The pooling layer is used to perform data downsampling. Finally, the perceptron component is used for target classification based on the features that were learned in the previous layers.

2.3.2 Machine Learning Methods

While there are many different types of machine learning methods, in the context of this thesis we evaluate and compare the categories of supervised, unsupervised, and semi-supervised machine learning techniques for speech recognition.

2.3.2.1 Supervised Learning

The machine learning task of iteratively learning a function that maps a given input to an output based on provided input and output pairs is called supervised learning [37]. The function is inferred using labelled training data consisting of the input object pair and the expected output value. The input object is commonly a vector, and the output value is also known as the supervisory signal.

In supervised learning, the machine learning algorithm is used to analyse data and produce the function used to map new input examples. The learning algorithm generalises from the training data to determine the class labels for previously unseen input examples. The most common supervised learning algorithms are support-vector machines (SVMs), linear

and logistic regression, naive Bayes classification, decision trees, k-nearest neighbours (KNN), and multilayer feedforward neural networks [38].

In automatic speech recognition, supervised learning techniques have been used for recognition errors detection [39], speaker identification [40], and speech to text transcription [41]. For speech recognition supervised learning using RNNs have been used for previous state-of-the-art results on the TIMIT dataset [27]. On the Librispeech dataset supervised learning has been used for recent state-of-the-art results of word error rate as low as 1.9% [31].

2.3.2.2 Unsupervised Learning

In contrast to supervised learning, where all data must be labelled, unsupervised learning does not use any labelled data. Supervised learning algorithms are capable of self-organising behaviour to capture patterns in the data as probability densities [42]. Their ability to discover similarities and differences in raw unlabelled data can be used to label large amounts of data through classification, find hidden patterns in data, reduce data dimensionality, and classify samples without the need for difficult to obtain labelled data.

The most popular unsupervised learning algorithms are k-means clustering, k-nearest-neighbours (KNN), principal component analysis (PCA), independent component analysis (ICA), and unsupervised ANNs [43].

Unsupervised learning has recently found significant use in speech recognition. The Facebook and Google AI teams collaborated on an unsupervised speech recognition model called wav2vec Unsupervised. The model leverages self-supervised speech representations from unlabelled audio to learn a mapping from representations to phonemes. The team reduced the phone error rate for the TIMIT benchmark to 11.3% and achieved a 5.9% word error rate on the English Librispeech benchmark [44]. While these results are still surpassed by the state-of-the-art benchmarks of supervised [31] and semi-supervised methods [7], it is a promising glimpse into the future of using unlabelled data for speech recognition.

2.3.2.3 Semi-Supervised Learning

Combining the use of a small amount of labelled data and a large quantity of unlabelled input data during the model's training is called semi-supervised learning. It is a compromise between supervised and unsupervised learning, where a large amount of labelled data is not needed to achieve state-of-the-art speech recognition results. The acquisition of unlabelled data is comparatively low-cost while acquiring large amounts of labelled data is often a challenge due to the cost of the manual labelling process.

Baevski *et al.* [3] were the first AI research group to demonstrate that learning representations from speech audio alone with the addition of fine-tuning on labelled speech can achieve state-of-the-art results. The self-supervised learning framework dubbed wav2vec 2.0 masks the input speech data and performs a contrastive task on the quantised latent self-learned speech representations. Using only 10 minutes of labelled data, it can outperform the previous state of the art result on the 100-hour subset of the Librispeech dataset while using 100 times less labelled data for training.

2.3.3 Current state-of-the-art

The following subchapter gives an overview of current state-of-the-art neural network models for both monolingual and multilingual speech recognition. Several current state-of-the-art speech recognition models are based on and extend the wav2vec 2.0 framework, which utilises semi-supervised learning for speech to text transcription. A more comprehensive overview of the wav2vec 2.0 architecture is given in Chapter 3.3.

2.3.3.1 Monolingual Speech Recognition

One of the primary benchmarks for automatic speech recognition has been the Librispeech English dataset. The current state-of-the-art result for Librispeech was achieved by Zhang *et al.* with their noisy student Conformer based model with wav2vec2 pre-training

to obtain 1.4% and 2.6% word error rates on Librispeech test and test-other subsets respectively [4].

Another model inspired by the success of masked language modelling in model pre-training is w2v-BERT introduced by Chung *et al.* [45] achieves similar results with word error rates of 1.4% on the Librispeech test and 2.5% on the test-other subset.

Finally, Xu *et al.* [46] show that a combination of self-training and pre-training can be complementary and effective for improving speech recognition systems with unlabelled data. The approach indicates that pseudo-labelling and pre-training with the wav2vec 2.0 model can be used to achieve 3% and 5.2% WER on the Librispeech test-clean and test-other datasets, respectively.

2.3.3.2 Multilingual Speech Recognition

Most speech recognition models focus on training a language-dependent model for each language separately, but recent research has improved multilingual speech recognition models to the point that the performance of a single multilingually trained model achieves similar or exceeds monolingual model performance.

XLSR [7] and XLS-R [8] are two recent multilingual speech recognition models developed specifically with multilingual transcription capabilities in mind.

Conneau *et al.*, present a multilingual model that is inspired by the wav2vec2 approach and is able to learn cross-lingual speech representations from the raw speech in multiple languages. The model is pre-trained across 53 different languages from the Common Voice [12] dataset. The model achieves a 72% relative phoneme error rate reduction to best know results on the Common Voice dataset and a 16% relative word error rate reduction on the BABEL [47] dataset.

Babu *et al.,* introduce a large-scale model extending the wav2vec2 approach that is trained on half a million hours of speech audio in 128 languages. The team produced a new state-of-the-art result on the BABEL, MLS [48], CommonVoice, VoxPopuli [49] and VoxLingua107 [13] datasets. More importantly, the team was able to show that cross-lingual pre-training is able to outperform English-only pre-training when translating English speech into other languages [8].

2.4 Speech Recognition Applications

Smart home devices and applications that simplify our daily lives and offer novel possibilities for entertainment are an area of significant commercial interest in recent years. Speech recognition has become an integral part of smart home solutions due to the convenience of interacting with the systems using voice commands. Recent surveys show that the usage of Personal Voice Assistants (PVAs) has grown to 21% of the United States population owning at least one smart speaker such as the Amazon Echo or Google Home [50].

In addition to smart home applications and PVAs, speech recognition covers a wide domain of potential applications including speech emotion recognition, speaker identification, and accessibility. In the following subchapters recent advancements in common application domains for speech recognition are explored.

2.4.1 Speech Emotion Recognition

Speech emotion recognition (SER) systems are methodologies that process and classify speech to identify emotion from the subject's voice. SER has been a research subject for over two decades and has various applications in human-computer interaction, computer games, mobile services, and psychological assessment [51].

Recent research demonstrates that smart home personal assistant systems with artificial intelligence (AI) based speech recognition can be used to analyse the speaker's emotions with up to 95% accuracy for speaker-independent experiments [52]. Emotion recognition from speech data has also been an essential aspect of teaching human social intelligence to robots used for assistance [53].

2.4.2 Speaker Identification

Voice assistant applications are widely used for interacting with smart home devices. However, they introduce potential privacy and security risks if malicious actors impersonate smart home users to send speech commands [54]. Thus speaker verification is an important research focus for speech recognition to reduce the potential risks. Additionally, speaker

identification has been used to reduce crime by identifying criminals and terrorists using voice data [55].

In recent research by Chung *et al.*, the group created a convolutional neural network-based large scale speaker identification system. The group achieved an error rate as low as 3.95% on a self-curated dataset of over 6000 speakers and over a million utterances dubbed VoxCeleb [56].

In another study by Zhou *et al.* ResNet based architecture was used to further improve the equal error rate (EER) metric by 18.5% relative to the previous work [54].

2.4.3 Accessibility

Another essential domain where speech recognition could potentially increase the everyday quality of life is accessibility. In multiple studies, speech recognition has been explored for applications that improve accessibility for the physically impaired.

In a study by Mandal *et al.* an ASR system called Shruti-II is proposed for converting continuous speech to Unicode so that the visually impaired community could use speech as input for controlling computer systems and sending an email [57].

In addition, some research focuses on improving speech recognition for speech impaired speakers suffering from various motor speech disorders such as dysarthria. Shahamiri *et al.* utilised array learners to improve the mean word recognition rate for dysarthria patients by 10% [58]. In work by Rosdi *et al.* [59], a speaker-independent ASR system was developed to measure speech-impaired speakers' speech intelligibility.

2.4.4 Other

Recently, speech recognition has become an essential part of speech enhancement systems, which are used to denoise and dereverberation of noisy signals. Subramanian *et al.* demonstrated the effectiveness of an end-to-end ASR system for speech enhancement [60] in a recent study.

Speech recognition can also be combined with other automatic recognition systems, such as written character recognition. One study proposes combining automatic speech and

character recognition to improve parcel sorting [61]. The experiments by the team showed a 90.2% zip code recognition rate compared to a traditional optical character recognition system with an 80.6% recognition rate.

Other areas of research include far-field ASR for recognition of speech that is spoken at a distance from the microphone [62], recognition of creaky voice from emergency calls to detect a person's emotional state [63], and even speaker identification for fighting crime and terrorism [55].

3 Experimental Approach

In the following chapter, the dataset and experimental approach for training the neural network-based speech recognition model are described. In the first subchapter, a detailed overview of the datasets used for pre-training, validation, fine-tuning and testing is given. The data preprocessing steps performed before training and testing the models are described in the second subchapter. The third subchapter provides an extensive description of the chosen neural network architecture. Finally, the neural network pre-training and fine-tuning procedures are presented in the fourth subchapter.

3.1 The Datasets

Table 1: The Librispeech corpus data structure

file (string)	'path_to_file.flac'
audio (dict)	'path': 'path.flac', 'array': array([0.00048828, 0.00018311, 0.00137329, ...], 'sampling_rate': 16000
id	'1272-141231-0000'
text	'A MAN SAID TO THE UNIVERSE SIR I EXIST'

The Librispeech corpus [64] contains approximately 1000 hours of English read speech sampled at 16kHz. The data is compiled from audiobooks read by volunteers from the LibriVox project. The project has collected around 8000 public domain audiobooks mostly as recordings based on public domain texts from Project Gutenberg [65]. Due to the size of the corpus, the training set is split into three subsets with 100, 360 and 500 hours respectively. The Librispeech data structure example can be seen in Table 1. The multilingual neural network model is pre-trained on 960 hours of unlabelled Librispeech audio data in the English language.

The multilingual dataset used for fine-tuning originates from the Mozilla Common Voice project [12]. The Common Voice dataset results from crowd-sourced data collection to produce publicly available multilingual speech data. It is an audio dataset where each sample consists of a unique MP3 file of spoken text and a corresponding transcription text file that the subject reads. The data is continuously collected through the project's website and it is

open for new contributions and previously collected data validation. There are presently a total of 13,905 recorded hours in the dataset. In addition to the speech data, it includes demographic information about the speaker, such as accent, sex, and age. The dataset includes 11,192 validated hours of speech samples in 76 different languages.

Table 2: The Common Voice data field structure

client_id (string)	'e570aa634f53f3496f29b20b54b7fc501e1b5b9e6d2cfc41ebbd090bbc3682555b'
path (string)	'common_voice_et_18039906.mp3'
audio (audio)	(audio waveform)
sentence (string)	Kusjuures selle nimel Mägi riskis isiklikult ja riskis suurelt.
up_votes (int)	2
down_votes (int)	0
age (string)	thirties
gender (string)	male
accent (string)	
locale (string)	et
segment (string)	

The CommonVoice dataset contains 43 hours of speech in the Estonian language with 32 hours of user validated speech. The Estonian language can be considered a low resource as high resource languages typically have hundreds or thousands of hours of recorded speech data available. The Estonian speech data is used in the fine-tuning phase for training, validation and testing of the pre-trained multilingual model. Additionally, the training split of this dataset provided by the HuggingFace datasets library [66] is separately used for training and the test split for testing the monolingual model selected for comparing the performance between the two approaches. An example of the CommonVoice data structure can be seen in Table 2.

3.2 Data Preprocessing

The Common Voice dataset contains samples that have been validated by users of the Common Voice project and examples that have not yet been validated. Any unvalidated samples were filtered out from the final dataset used for training and testing.

Additionally, each validated sample has associated metadata such as the number of "upvotes" and "downvotes" that signify the opinion of the validator if the spoken sample matches the labelled text. All data samples with a downvote to upvote ratio over 0.5 were removed to have only the highest quality samples in the dataset.

The Common Voice dataset contains many features irrelevant to the speech recognition task. The features *segment, up_votes, down_votes, accent, client_id, gender, segment, locale, and age* were removed from all data splits before training/evaluation.

Furthermore, the speech transcriptions contain special characters such as *,.?!;:*. It is challenging to classify speech into such special characters without a language model because they do not have a corresponding sound. These characters are also not relevant to understanding the meaning of the speech and are consequently removed from the target text using regular expression matching.

For the training and validation data split, we also extract all the unique letters used in the labels for the model tokenizer. These are necessary for training the model to classify each speech chuck into a corresponding target letter.

Finally, the original audio samples in the Common Voice dataset are sampled at 46kHz bitrate, while the Librispeech dataset contains data sampled at 16kHz. All the Common Voice data is resampled to 16kHz to have a uniform and optimal sampling rate for the models.

3.3 Model Architecture

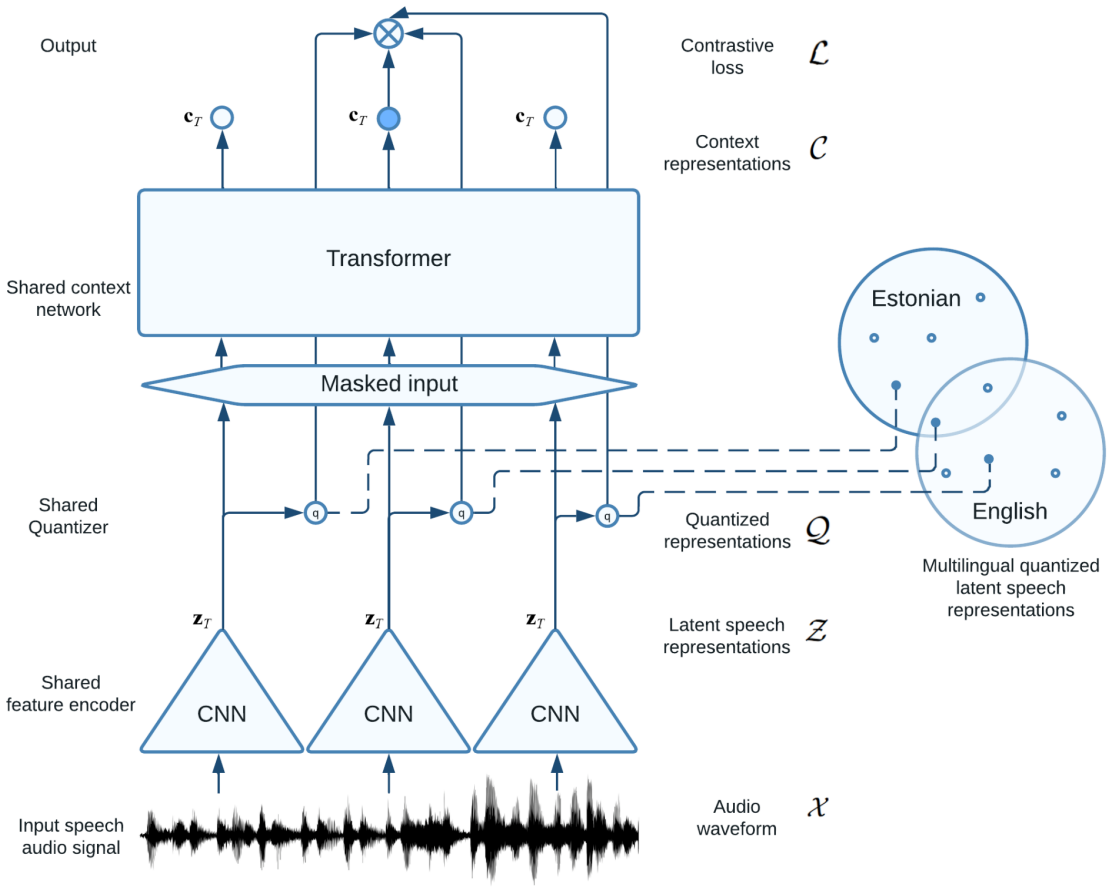

Figure 1. The wav2vec2 model architecture

Both the mono- and multilingual models are implemented using the XLSR approach introduced by Conneau *et al.* [7]. The model architecture is illustrated in Figure 1. This approach extends the wav2vec 2.0 model [3] developed for semi-supervised monolingual speech recognition with the addition of the encoder, quantizer and masked Transformer being shared across all languages used for pre-training. This model architecture was chosen as it is the basis for all current state-of-the-art multilingual speech recognition models and has been demonstrated to be suitable for improving speech recognition for low resource languages [9, 67].

3.3.1 Feature Encoder

The model comprises of a convolutional multi-layer masked feature encoder $f : X \mapsto Z$. The encoder maps audio samples X to latent speech representations $\mathbf{z}_1,...,\mathbf{z}_T$ in multiple languages. Each $\mathbf{z}_t$ corresponds to approximately 25ms of speech audio with a stride of 20ms. The encoder consists of multiple blocks of temporal convolutions followed by a normalisation layer [68] and Gaussian Error Linear Unit (GELU) [69] activation function. The raw audio waveform is normalised to zero mean and unit variance to have a uniform audio across all samples. For the purpose of stabilisation, L2 regularisation penalty [70] is applied over the feature encoder outputs. The stride of the encoder is used to determine the time steps T used as input for the Transformer.

3.3.2 Transformer

The contextualised representations of latent speech $\mathbf{z}_1,...,\mathbf{z}_T$ for $\boldsymbol{T}$ time-steps are produced on the unlabelled speech data using a Transformer architecture [71, 72] based context network. The Transformer architecture follows the BERT context network design [73].

The Transformer $g : Z \mapsto C$ builds context representations $\mathbf{c}_1,...,\mathbf{c}_T$ that attain information from the whole feature encoder output sequence [74]. In the Transformer architecture a convolutional layer is used for encoding relative positional embeddings. The output of the convolutions is activated by the GELU activation function and finally layer normalisation is applied to the output result.

3.3.3 Shared Quantizer

In parallel to the transformer the feature encoder representation output is discretized to $\mathbf{q}_t$ with a quantization module $h : Z \mapsto Q$. The contextualised representations are mapped to a finite set of speech representations using product quantization [75]. Quantization is used to generate exponentially large codebooks with a low memory and time cost.

Product quantization is achieved by choosing and concatenating quantized representations from different codebooks. One entry from each codebook B with entries $v \in$

$\mathbb{R}^{V \times d/B}$ is chosen, the resulting vectors $v_1,...,v_B$ are concatenated and a linear transformation $\mathbb{R}^d \mapsto \mathbb{R}^f$ is applied to get the output $\mathbf{q} \in \mathbb{R}^f$.

Additionally, Gumbel softmax is used to differentially choose discrete codebook entries. The straight-through estimator described by Dongwei *et al.* is used [72] along with B hard Gumbel softmax operations [76]. The output $\mathbf{z}$ from the feature encoder is mapped to logits $\mathbf{l} \in \mathbb{R}^{B \times V}$.

Finally, the probability of choosing the v-th codebook entry from codebook b is described as (Eq 3.1):

$$p_{b,v} = \frac{exp(l_{b,v} + n_v)/\tau}{\sum_{k=1}^{V} exp(l_{g,k} + n_k)/\tau},$$

Eq. 3.1: Probability of choosing v-th codebook entry from codebook b

where $n = -\log(-\log(u))$ and u are uniformly picked samples from $\mathcal{U}(0, 1)$ and τ is a non-negative temperature [76].

3.4 Training Procedure

The training process consists of two major phases - pre-training and fine-tuning. The model learns context representations of unlabelled input speech in the pre-training phase. While in the fine-tuning phase small amounts of labelled samples of the target language are used to fine-tune the model for speech recognition in the target language. In the following subchapters a more detailed overview of the pre-training and fine-tuning procedure are given.

3.4.1 Pre-training

The objective of the pre-training is to learn context representations of input speech audio by solving a contrastive learning task L_m. The task is to find the correct quantized latent speech representation for each masked time step in a set of distractors. In addition, the result is augmented by codebook diversity loss L_d with the aim of using the codebook entries equally

as often. The speech audio representations L (Eq. 3.2) with a tuneable hyperparameter β can be described as

$$L = L_m + \beta \cdot L_d .$$

Eq. 3.2: Formula for context representations L

The contrastive loss task L_m is solved by identifying the quantized latent speech representation $\mathbf{q}_t$ in a given collection of $D + 1$ quantized representations $\mathbf{q} \in \mathbf{Q}_t$, where $\mathbf{Q}_t$ is composed of the correct representation $\mathbf{q}_t$ and D distractors [77]. The distractors are homogeneously chosen from the other masked time steps of the given input sample. The resulting loss function (Eq. 3.3) is defined as

$$L_m = -\log \frac{exp(\frac{sim(b_t, q_t)}{k})}{\sum_{q \in Q_t} exp(\frac{sim(b_t, q)}{k})} ,$$

Eq. 3.3: Formula for contrastive loss function

where the cosine similarity *sim* [78] (Eq. 3.4) between the context representations C and quantized latent speech representations Q is computed as

$$sim(\mathbf{x}, \mathbf{y}) = \frac{x^T \cdot y}{|x| \cdot |y|} .$$

Eq. 3.4: Formula for calculating the cosine similarity

Finally, the contrastive task L is dependent on the codebook to contain both positive and negative examples in an equal distribution. The diversity loss L_d is aimed to increase the use of quantized codebook entries. The equal use of entries V in codebooks B is promoted by the maximisation of the averaged softmax distribution entropy over each codebook's entries p_b in a batch of samples. The diversity loss function (Eq. 3.5) is defined as

$$L_d = \frac{1}{BV} \sum_{b=1}^{B} - H(p_b) = \sum_{b=1}^{B} \sum_{v=1}^{V} p_{b,v} \cdot log\, p_{b,v} .$$

Eq. 3.5: Diversity loss function

Pre-trained models for both English and Estonian exist for the wav2vec2.0 architecture in the HuggingFace transformers library. However, the models are pre-trained in multiple languages, the data splits used for pre-training are not clearly specified, and a model pre-trained on only Estonian or only English and Estonian data is not available. As a consequence, to have comparable results for the experiments pre-training on only Estonian data and both English and Estonian data was performed in the experimental validation part of this thesis.

The models for pre-training are implemented using the Fairseq [79] toolkit developed by the Facebook AI research team. The Fairseq toolkit is an open-source sequence modelling toolkit for the Python programming language that allows customising machine learning pipelines for translation and speech recognition tasks and loading preconfigured model parameters. It is based on the PyTorch [80] library and is used for fast distributed training across multiple GPUs. The source code used for pre-training can be found in the link provided in Appendix I.

3.4.2 Fine-tuning

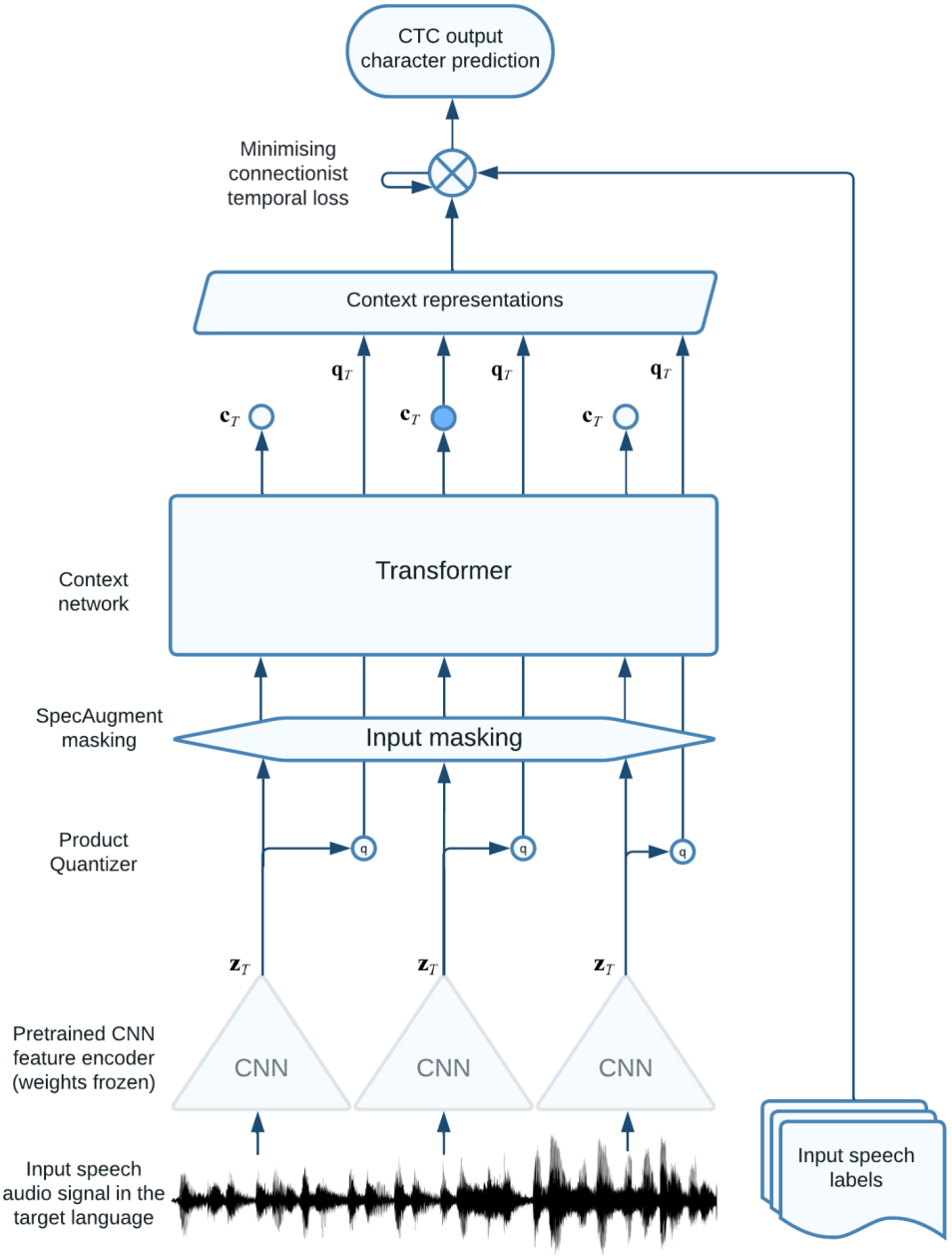

Figure 2. The fine-tuning procedure

The pre-trained speech representations of the model need to be fine-tuned for the speech recognition task using some labelled data from the target language. The fine-tuning procedure is illustrated in Figure 2. The models are fine-tuned by adding a randomly initialised linear

projection output layer on top of the Transformer network to make character predictions into *N* classes that represent the vocabulary of the target language.

This classifier represents the output vocabulary of the target language and is used to adjust the model for transcribing the target language speech. For the Common Voice dataset, there are $N = 35$ tokens from the Estonian language for classifier character targets.

The weights of the underlying feature encoder are frozen and not updated during the fine-tuning to preserve the speech representations learned during pre-training.

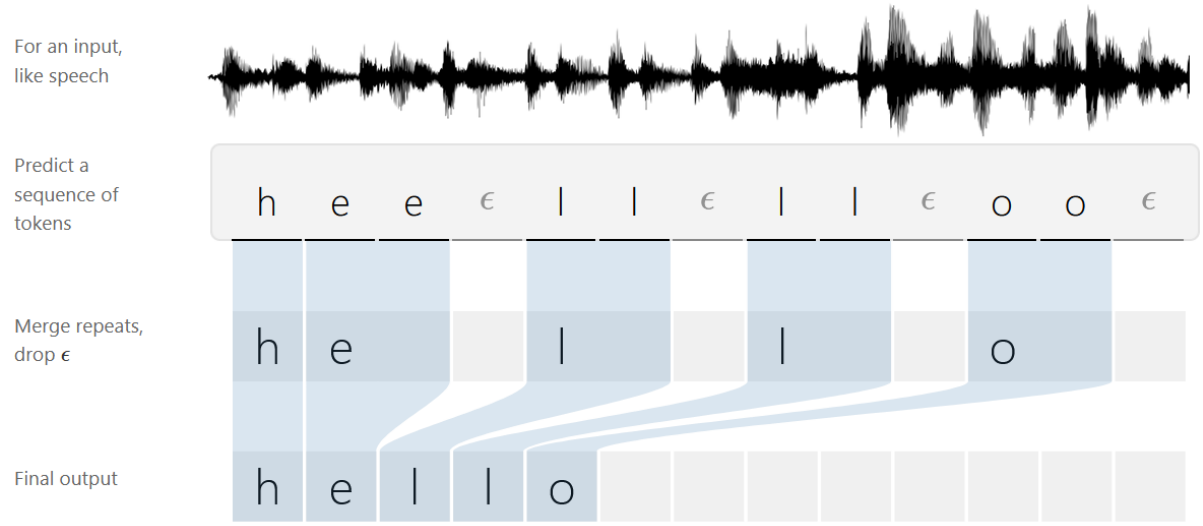

Figure 3. Connectionist Temporal Classification [81]

In speech audio, multiple time slices can correspond to a single spoken phone and since the alignment of the observed sequence is unknown it is difficult to give accurate probabilities to labels that span across multiple time slices. The transformer part of the model is trained on the labelled data by minimising the Connectionist Temporal Classification (CTC) loss [82]. The CTC algorithm takes a sequence of observations as input and outputs a sequence of labels that can also include blank labels. CTC was designed for temporal classification tasks where the alignment of the observed sequence is unknown. It predicts a probability distribution for the labels at each time step with a continuous output [81]. As illustrated in Figure 3 on a high level the algorithm predicts a sequence of tokens corresponding to the input speech representations, merges character repeats and drops the blank token $\boldsymbol{\varepsilon}$ to produce the final output sequence. The token $\boldsymbol{\varepsilon}$ is added to the vocabulary of allowed outputs as a representation of the transition state between two classes.

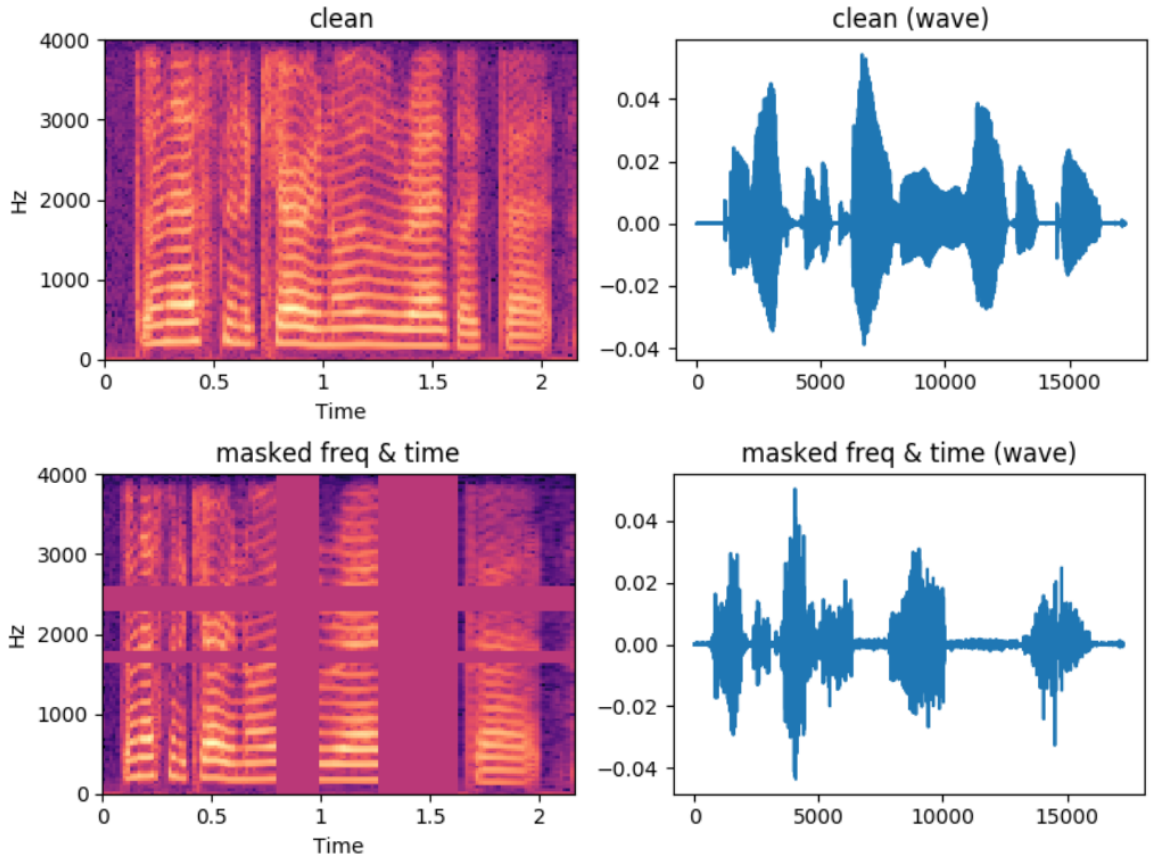

Figure 4: Comparison of clean and masked audio spectrogram and waveform [83]

An altered variant of SpecAugment [83] masking reduces overfitting on training data and improves error rates on the small amount of available labelled training data. The SpecAugment augmentation warps the features and masks blocks of frequency channels and time steps. A number of starting steps are randomly chosen, and a span of the next ten time-steps is substituted with an embedding mask. Additionally, a number of channels are chosen for masking as starting indices, and each of them is expanded to cover the following 64 channels.

When masking, 6.5% of the samples are randomly chosen to be starting indices and the following $M = 10$ time steps are masked. The number of distractors mixed in with the target true quantized latent $\mathbf{q}_t$ is $D = 100$ for each masked time-step. This means that roughly 49% of time steps are masked with a mean span length of 299ms.

A side by side comparison of clean audio compared to masked audio spectrograms and waveforms can be seen in Figure 4. As a final step, a form of structured dropout called LayerDrop [84] is applied at a rate of 0.05 to reduce overfitting.

The fine-tuning of models is implemented using the HuggingFace transformers library [85], which provides pre-configured model configurations for the wav2vec2 model that can be extended by loading custom weights from the pre-training step. The source code used for fine-tuning can be found in the link provided in Appendix I.

4 Experimental Validation

In the following chapter, the experimental validation procedure and the results of the experiments are presented and analysed. The state-of-the-art classifier wav2vec adapted for multilingual speech recognition with the XLSR approach [44] was evaluated with the proposed preprocessing method. The speech transcription results of a monolingually trained model are compared to a multilingual model that was trained on a significantly larger amount of multilingual data. The experimental validation of the thesis consists of pre-training the two types of speech recognition models - monolingual and multilingual. After pre-training the models are fine-tuned in the target language. The download link for the Python source code developed for preprocessing, pre-training, fine-tuning, and evaluation can be found in Appendix I.

In the following chapter, the experimental validation steps are described. The first subchapter presents the model evaluation process, the hyperparameters and hardware used for training and testing the models**.** In the second subchapter, the results validation methodology is detailed. The statistical significance test and its application to the experimental results is described in the third subchapter. The results and analysis of the experiments on the mono- and multilingual models are given in the fourth subchapter. Lastly, future work ideas and possibilities are discussed in the final subchapter.

4.1 Model Evaluation

The pre-training of the monolingual model was performed on the training split of the Common Voice dataset, the fine-tuning was performed on the validation split of the Common Voice dataset, and the evaluation of the fully trained model was performed on the testing split of the Common Voice dataset. The data splits correspond to randomly selected samples where 70% of the overall data is used for pre-training, 10% for fine-tuning and 20% for testing. The training was executed and the performance of the monolingual and multilingual models was evaluated as follows.

The monolingual model was initially pre-trained on 5587 samples of unlabelled Estonian speech from the Common Voice dataset for 200 epochs. After pre-training, the

model was fine-tuned on 798 samples of labelled Estonian speech for 200 epochs. Finally, the model was evaluated on 1596 randomly selected samples from the Common Voice dataset, and the WER and CER metrics were calculated.

The multilingual model was pre-trained on 960 hours of unlabelled English speech from the Librispeech dataset and the same 5587 samples of unlabelled Estonian speech from the training split of the Common Voice dataset that was used for monolingual pre-training. After pre-training the multilingual model was fine-tuned on 798 samples of labelled Estonian speech data from the Common Voice validation split for a total of 200 epochs. Finally, the model was evaluated on the 1596 samples from the testing split of the Common Voice dataset. The CER, WER and McNemar's test *p*-value relative to the monolingual model were calculated on the evaluation results of the fully trained model.

Both models were pre-trained, fine-tuned and tested on the University of Tartu High Performance Computing (HPC) distributed computing clusters running on Linux CentOS 7 operating system. The pre-training and fine-tuning processes were executed on a Nvidia Tesla A100 graphical processing unit (GPU) with 80GB vRAM, 64 AMD EPYC 7713 central processing unit (CPU) cores and 64GB of random access memory (RAM). The testing process was executed on 16 cores of the Xeon(R) E5-2660 v2 CPU and 16GB of RAM.

4.2 Results Validation

The fully trained models are validated using the word error rate (WER) and character error rate (CER) metrics. The WER metric consists of the number of all word errors divided by the total number of words. It can be computed as

$$\mathrm{WER} = \frac{S_w + D_w + I_w}{N_w}$$

where S_w is the number of substituted words, D_w is the number of deleted words, I_w is the number of inserted words needed to construct the reference from the transcription, and N_w is the total number of words in the reference.

Similarly, the CER metric consists of the number of all character errors divided by the total number of characters. It can be computed as

$$\text{CER} = \frac{S_c + D_c + I_c}{N_c}$$

where S_c is the number of substituted characters, D_c is the number of deleted characters, I_c is the number of inserted characters needed to assemble the reference label from the transcription text, and N_c is the total number of characters in the transcription.

The WER and CER metrics are calculated for each transcription and an average rate for both metrics over all the samples is reported.

4.3 Statistical Significance

A common approach to evaluate the classification methods in machine learning is the *K*-fold cross-validation technique [86]. In addition, the paired Student's t-test [87] is usually performed on the cross validation results to confirm the statistical significance in the difference of the results of the evaluated models. However, when performing *K*-fold cross-validation during training, each sample is used in the training data *K*-1 times. As a result, the skill scores estimation cannot be considered independent and the *K*-fold cross-validation approach has an increased likelihood of type I error [88].

Table 3: McNemar's test contingency table

	Classifier 2 Correct	Classifier 2 Incorrect
Classifier 1 Correct	**Correct**/**Correct** (*a*)	**Correct**/Incorrect (*b*)
Classifier 1 Incorrect	Incorrect/**Correct** (*c*)	Incorrect/Incorrect (*d*)

To overcome this limitation, the statistical significance of the experimental results can be evaluated using McNemar's statistical test [89]. The McNemar's test is a paired nonparametric or distribution-free statistical hypothesis test. The test evaluates if the disagreements in errors between the two classifiers under evaluation are similar. The test

statistic is computed by forming a contingency table (see Table 3) which sums up the number of samples *a* that the two models both predicted correctly, the number of samples *d* that both predicted incorrectly, or the number of samples *b* and *c* that one model predicted correctly and the other predicted incorrectly. Using the values *b* and *c* from the contingency table, McNemar's test statistic can be calculated with the formula (Eq. 4.1):

$$\mathcal{X}^2 = \frac{(b-c)^2}{b+c}$$

Eq. 4.1: The McNemar's test statistic

The classifiers under comparison should have the same error rate under the null hypothesis. The null hypothesis can only be rejected if the models make differing errors and have a differing relative proportion of errors when evaluated on the test data. McNemar's test has been shown to have a lower rate of type I errors and can be used to demonstrate statistical significance as a potentially more reliable alternative to the *K*-fold cross validation technique [88].

4.4 Experimental Results & Analysis

Model	Monolingual	Multilingual
Pretrained Eng (h)	N/A	960h
Pretrained Est (h)	7.5h	7.5h
Finetuned Est (h)	1.25h	1.25h
Testing Est (h)	2.5h	2.5h
WER	26.9	**12.1**
CER	5.9	**5.0**
p-value	< 0.001	1

Table 4: Experimental results for the evaluated monolingual and multilingual models

The monolingual model was initially pre-trained on 5587 samples or 7.5 hours of unlabelled Estonian speech from the Common Voice dataset for a total of 200 epochs. After pre-training the model was fine-tuned on 798 samples or 1.25 hours of labelled Estonian speech for a total of 200 epochs. Finally, the model was evaluated on randomly selected 1596 samples or 2.5 hours of labelled data from the Common Voice dataset and the CER and WER metrics were calculated. The monolingual model was evaluated to have an average of 26.9% WER and 5.9% CER on the Common Voice test data split.

The multilingual model was pre-trained on 960h of unlabelled English speech from the Librispeech dataset in addition to the 5587 samples of unlabelled Estonian speech from Common Voice dataset. After pre-training the model was fine-tuned on 798 samples of labelled Estonian speech for a total of 200 epochs and evaluated on 1596 randomly selected data from the Common Voice dataset and the CER, WER and McNemar's test *p*-value relative to the monolingual model were calculated.

The multilingual model achieved 12.1% WER and 5% CER on the test Common Voice data split. This represents a 53.6% decrease in word errors and 15.3% decrease in character errors. The evaluation metrics and the *p*-value[1] relative to the monolingual model can be seen in Table 4.

These results can be considered statistically significant, because McNemar's test *p*-value is below 0.001, which allows us to reject the null hypothesis that the models make similar errors. Furthermore, the results suggest that the multilingually pre-trained semi-supervised machine learning model is well-suited for speech recognition for low resource languages, because it can leverage unlabelled data from other languages to statistically significantly improve error rates compared to the monolingual pre-training approach.

4.5 Future Work

During the research and experimental validation of this work, several occasions were identified that require further study and experimental investigation. This subchapter proposes

[1] The *p*-value is the probability of obtaining results at least as extreme as the results observed during testing. The value was calculated relative to the multilingual model and it shows the statistical significance of the difference of the two evaluated models.

ideas for future experiments with the multilingual speech recognition model to improve speech recognition error rates for low resource languages.

This research was mainly focused on verifying the feasibility of multilingual unlabelled pre-training for improving speech recognition for the Estonian language. However, there are thousands of less commonly spoken low resource languages that could benefit from unsupervised pre-training to have improved results for speech recognition. More research should be performed on different low resource languages to verify if this type of model architecture and approach can be generalised across more languages.

In addition, the current research focused on using the English and Estonian languages for multilingual pre-training, where English was chosen for its relatively high amount of available data. However, these two languages are not linguistically closely related. This means that the language representations learned from English data during pre-training are not optimally representative of the target Estonian language. Using additional languages for pre-training from the same language family as Estonian could further improve the multilingual model's performance. However, this is presently limited by the insufficient availability of data from the Finno-Ugric language group to significantly impact the resource-demanding process of pre-training.

Furthermore, the statistical significance evaluation could be further improved by decreasing the likelihood of type I errors. Some alternatives to McNemar's test include the 5x2-fold cross-validation or the modified *K*-fold cross-validation that utilises the corrected paired Student’s t-test.

Finally, the testing of the models was limited to the Common Voice dataset due to its quality and inclusion of a large amount of Estonian speech data. However, the Common Voice dataset contains only very clean speech recordings that are read by the subjects and performance on less precise speech should be verified. The Common Voice dataset also contains predominantly male speaking voices and the models might struggle to recognise the female speaking voice as accurately. Alternative datasets from a wider range of sources for testing the model performance in the Estonian language should be explored.

Summary

This thesis aimed to investigate improving speech recognition for low resource languages such as the Estonian language. Recent studies have demonstrated the efficacy of speech recognition models pre-trained on unlabelled data of high resource languages and fine-tuned on small amounts of data in the target language to improve the error rate compared to monolingual models. Similarly, the goal of this research was to show that using a model based on the wav2vec 2.0 approach that is pre-trained on a large dataset of English speech with minimal fine-tuning on the target language can be used to improve speech recognition of the Estonian language.

The monolingual and multilingual models implement the wav2vec 2.0 framework based architecture. The monolingual model was pre-trained on unlabelled Estonian language data and fine-tuned and tested on labelled Estonian language dataset from the Common Voice project. The multilingual model was pre-trained on unlabelled data from the Librispeech English dataset and unlabelled data from the Common Voice Estonian dataset, but also fine-tuned and tested on the Estonian language data.

The monolingual model evaluation resulted in an average of 26.9% WER and 5.9% CER on the Common Voice dataset, while the multilingual model achieved lower error rates of 12.1% WER and 5.0% CER. These results represent a 53.6% decrease in word errors and a 15.3% decrease in character errors for the multilingual model compared to the monolingual model. The multilingual model is able to utilise the unlabelled multilingual data to reduce the need for labelled data of the target language. However, the multilingual model uses a significantly larger amount of data for pre-training, and the training process is much slower.

These results demonstrate the potential of multilingual self-supervised pre-training to improve speech recognition for low resource languages. However, further experiments with multiple low resource languages should be explored to verify that these results can be generalised across more languages. Additionally, alternative datasets for testing the model performance in the Estonian language should be explored as the Common Voice dataset contains only very clean speech recordings that are read by the subjects and performance on less precise speech should be verified.

Bibliography

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

[2] J. Noyes and C. Frankish, "Speech recognition technology for individuals with disabilities," *Augment. Altern. Commun.*, vol. 8, no. 4, pp. 297–303, Jan. 1992, doi: 10.1080/07434619212331276333.

[3] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," *ArXiv200611477 Cs Eess*, Oct. 2020, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2006.11477

[4] Y. Zhang *et al.*, "Pushing the Limits of Semi-Supervised Learning for Automatic Speech Recognition," *ArXiv201010504 Cs Eess*, Oct. 2020, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2010.10504

[5] M. Rivière, A. Joulin, P.-E. Mazaré, and E. Dupoux, "Unsupervised pretraining transfers well across languages," *ArXiv200202848 Cs Eess*, Feb. 2020, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2002.02848

[6] G. Lample and A. Conneau, "Cross-lingual Language Model Pretraining," *ArXiv190107291 Cs*, Jan. 2019, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/1901.07291

[7] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised Cross-lingual Representation Learning for Speech Recognition," *ArXiv200613979 Cs Eess*, Dec. 2020, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2006.13979

[8] A. Babu *et al.*, "XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale," *ArXiv211109296 Cs Eess*, Dec. 2021, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2111.09296

[9] Z.-Q. Zhang, Y. Song, M.-H. Wu, X. Fang, and L.-R. Dai, "XLST: Cross-lingual Self-training to Learn Multilingual Representation for Low Resource Speech Recognition," *ArXiv210308207 Cs Eess*, Mar. 2021, Accessed: May 07, 2022. [Online]. Available: http://arxiv.org/abs/2103.08207

[10] D. Yu and L. Deng, *Automatic Speech Recognition*. London: Springer London, 2015. doi: 10.1007/978-1-4471-5779-3.

[11] P. K. Donepudi, "Voice Search Technology: An Overview," *Eng. Int.*, vol. 2, no. 2, pp. 91–102, Dec. 2014, doi: 10.18034/ei.v2i2.502.

[12] R. Ardila *et al.*, "Common Voice: A Massively-Multilingual Speech Corpus," *ArXiv191206670 Cs*, Mar. 2020, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/1912.06670

[13] J. Valk and T. Alumae, "VOXLINGUA107: A Dataset for Spoken Language Recognition," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, Shenzhen, China, Jan. 2021, pp. 652–658. doi: 10.1109/SLT48900.2021.9383459.

[14] A. Arkhangorodsky *et al.*, "MeetDot: Videoconferencing with Live Translation Captions," *ArXiv210909577 Cs*, Sep. 2021, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2109.09577

[15] M. R. Primorac, Sanja, "Android application for sending SMS messages with speech recognition interface.," *2012 Proc. 35th Int. Conv. MIPRO*, 2012.

[16] J. Besser, M. Larson, and K. Hofmann, "Podcast search: user goals and retrieval technologies," *Online Inf. Rev.*, vol. 34, no. 3, pp. 395–419, Jun. 2010, doi: 10.1108/14684521011054053.

[17] V. Kepuska and G. Bohouta, “Next-generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home),” in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, Jan. 2018, pp. 99–103. doi: 10.1109/CCWC.2018.8301638.
[18] R. Pallás-Areny and J. G. Webster, *Analog signal processing*. New York: Wiley, 1999.
[19] M. H. Weik, *Communications standard dictionary*, 3rd ed. New York: Chapman & Hall, 1996.
[20] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities.,” *Proc. Natl. Acad. Sci.*, vol. 79, no. 8, pp. 2554–2558, Apr. 1982, doi: 10.1073/pnas.79.8.2554.
[21] D. Graupe, *Principles of artificial neural networks*, 3rd edition. New Jersey: World Scientific, 2013.
[22] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge, Massachusetts: The MIT Press, 2016.
[23] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[24] Y. Yu, X. Si, C. Hu, and J. Zhang, “A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures,” *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019, doi: 10.1162/neco_a_01199.
[25] D. Svozil, V. Kvasnicka, and J. Pospichal, “Introduction to multi-layer feed-forward neural networks,” *Chemom. Intell. Lab. Syst.*, vol. 39, no. 1, pp. 43–62, Nov. 1997, doi: 10.1016/S0169-7439(97)00061-0.
[26] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra, “DRAW: A Recurrent Neural Network For Image Generation,” *ArXiv150204623 Cs*, May 2015, Accessed: Apr. 16, 2022. [Online]. Available: http://arxiv.org/abs/1502.04623
[27] A. Graves, A. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, May 2013, pp. 6645–6649. doi: 10.1109/ICASSP.2013.6638947.
[28] M. Auli, M. Galley, C. Quirk, and G. Zweig, “Joint Language and Translation Modeling with Recurrent Neural Networks,” Oct. 2013, Proc. of EMNLP. [Online]. Available: https://www.microsoft.com/en-us/research/publication/joint-language-and-translation-modeling-with-recurrent-neural-networks/
[29] D. Guera and E. J. Delp, “Deepfake Video Detection Using Recurrent Neural Networks,” in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Auckland, New Zealand, Nov. 2018, pp. 1–6. doi: 10.1109/AVSS.2018.8639163.
[30] S. Ebrahimi Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal, “Recurrent Neural Networks for Emotion Recognition in Video,” in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, Seattle Washington USA, Nov. 2015, pp. 467–474. doi: 10.1145/2818346.2830596.
[31] W. Han *et al.*, “ContextNet: Improving Convolutional Neural Networks for Automatic Speech Recognition with Global Context,” *ArXiv200503191 Cs Eess*, May 2020, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2005.03191
[32] K. Fukushima, “Neocognitron,” *Scholarpedia*, vol. 2, no. 1, p. 1717, 2007, doi: 10.4249/scholarpedia.1717.
[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[34] R. Collobert and J. Weston, “A unified architecture for natural language processing: deep neural networks with multitask learning,” in *Proceedings of the 25th international conference on Machine learning - ICML ’08*, Helsinki, Finland, 2008, pp. 160–167. doi: 10.1145/1390156.1390177.

[35] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, “Convolutional Neural Networks for Speech Recognition,” *IEEEACM Trans. Audio Speech Lang. Process.*, vol. 22, no. 10, pp. 1533–1545, Oct. 2014, doi: 10.1109/TASLP.2014.2339736.

[36] F. C. Morabito *et al.*, “Deep convolutional neural networks for classification of mild cognitive impaired and Alzheimer’s disease patients from scalp EEG recordings,” in *2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI)*, Bologna, Italy, Sep. 2016, pp. 1–6. doi: 10.1109/RTSI.2016.7740576.

[37] S. J. Russell, P. Norvig, and E. Davis, *Artificial intelligence: a modern approach*, 3rd ed. Upper Saddle River: Prentice Hall, 2010.

[38] R. Caruana and A. Niculescu-Mizil, “An empirical comparison of supervised learning algorithms,” in *Proceedings of the 23rd international conference on Machine learning - ICML ’06*, Pittsburgh, Pennsylvania, 2006, pp. 161–168. doi: 10.1145/1143844.1143865.

[39] R. Errattahi, A. E. Hannani, H. Ouahmane, and T. Hain, “Automatic speech recognition errors detection using supervised learning techniques,” in *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, Agadir, Morocco, Nov. 2016, pp. 1–6. doi: 10.1109/AICCSA.2016.7945669.

[40] P. Matejka *et al.*, “Analysis of DNN approaches to speaker identification,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, Mar. 2016, pp. 5100–5104. doi: 10.1109/ICASSP.2016.7472649.

[41] A. Singh and R. S. Anand, “Speech Recognition Using Supervised and Unsupervised Learning Techniques,” in *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, Jabalpur, India, Dec. 2015, pp. 691–696. doi: 10.1109/CICN.2015.320.

[42] G. E. Hinton and T. J. Sejnowski, Eds., *Unsupervised learning: foundations of neural computation*. Cambridge, Mass: MIT Press, 1999.

[43] M. E. Celebi and K. Aydin, Eds., *Unsupervised Learning Algorithms*. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-24211-8.

[44] A. Baevski, W.-N. Hsu, A. CONNEAU, and M. Auli, “Unsupervised Speech Recognition,” in *Advances in Neural Information Processing Systems*, 2021, vol. 34, pp. 27826–27839. [Online]. Available: https://proceedings.neurips.cc/paper/2021/file/ea159dc9788ffac311592613b7f71fbb-Paper.pdf

[45] Y.-A. Chung *et al.*, “W2v-BERT: Combining Contrastive Learning and Masked Language Modeling for Self-Supervised Speech Pre-Training,” *ArXiv210806209 Cs Eess*, Sep. 2021, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2108.06209

[46] Q. Xu *et al.*, “Self-Training and Pre-Training are Complementary for Speech Recognition,” in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada, Jun. 2021, pp. 3030–3034. doi: 10.1109/ICASSP39728.2021.9414641.

[47] P. Roach *et al.*, “BABEL: an Eastern European multi-language database,” in *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP ’96*, Philadelphia, PA, USA, 1996, vol. 3, pp. 1892–1893. doi: 10.1109/ICSLP.1996.608002.

[48] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, “MLS: A Large-Scale Multilingual Dataset for Speech Research,” 2020, doi: 10.48550/ARXIV.2012.03411.
[49] C. Wang *et al.*, “VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation,” *ArXiv210100390 Cs Eess*, Jul. 2021, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/2101.00390
[50] P. Cheng and U. Roedig, “Personal Voice Assistant Security and Privacy—A Survey,” *Proc. IEEE*, vol. 110, no. 4, pp. 476–507, Apr. 2022, doi: 10.1109/JPROC.2022.3153167.
[51] M. B. Akçay and K. Oğuz, “Speech emotion recognition: Emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers,” *Speech Commun.*, vol. 116, pp. 56–76, Jan. 2020, doi: 10.1016/j.specom.2019.12.001.
[52] R. Chatterjee, S. Mazumdar, R. S. Sherratt, R. Halder, T. Maitra, and D. Giri, “Real-Time Speech Emotion Analysis for Smart Home Assistants,” *IEEE Trans. Consum. Electron.*, vol. 67, no. 1, pp. 68–76, Feb. 2021, doi: 10.1109/TCE.2021.3056421.
[53] B. A. Erol, A. Majumdar, P. Benavidez, P. Rad, K.-K. R. Choo, and M. Jamshidi, “Toward Artificial Emotional Intelligence for Cooperative Social Human–Machine Interaction,” *IEEE Trans. Comput. Soc. Syst.*, vol. 7, no. 1, pp. 234–246, Feb. 2020, doi: 10.1109/TCSS.2019.2922593.
[54] T. Zhou, Y. Zhao, and J. Wu, “ResNeXt and Res2Net Structures for Speaker Verification,” in *2021 IEEE Spoken Language Technology Workshop (SLT)*, Shenzhen, China, Jan. 2021, pp. 301–307. doi: 10.1109/SLT48900.2021.9383531.
[55] B. Ionescu, M. Ghenescu, F. Rastoceanu, R. Roman, and M. Buric, “Artificial Intelligence Fights Crime and Terrorism at a New Level,” *IEEE Multimed.*, vol. 27, no. 2, pp. 55–61, Apr. 2020, doi: 10.1109/MMUL.2020.2994403.
[56] J. S. Chung, A. Nagrani, and A. Zisserman, “VoxCeleb2: Deep Speaker Recognition,” *Interspeech 2018*, pp. 1086–1090, Sep. 2018, doi: 10.21437/Interspeech.2018-1929.
[57] Sandipan Mandal, Biswajit Das, and Pabitra Mitra, “Shruti-II: A vernacular speech recognition system in Bengali and an application for visually impaired community,” in *2010 IEEE Students Technology Symposium (TechSym)*, Kharagpur, India, Apr. 2010, pp. 229–233. doi: 10.1109/TECHSYM.2010.5469156.
[58] S. R. Shahamiri and S. K. Ray, “On the use of array learners towards Automatic Speech Recognition for dysarthria,” in *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, Auckland, New Zealand, Jun. 2015, pp. 1283–1287. doi: 10.1109/ICIEA.2015.7334306.
[59] F. Rosdi, M. B. Mustafa, and S. S. Salim, “Assessing automatic speech recognition in measuring speech intelligibility: A study of malay speakers with speech impairments,” in *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)*, Langkawi, Nov. 2017, pp. 1–6. doi: 10.1109/ICEEI.2017.8312396.
[60] A. S. Subramanian *et al.*, “Speech Enhancement Using End-to-End Speech Recognition Objectives,” in *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, Oct. 2019, pp. 234–238. doi: 10.1109/WASPAA.2019.8937250.
[61] A. Singh, A. Sangwan, and J. H. L. Hansen, “Improved parcel sorting by combining automatic speech and character recognition,” in *2012 IEEE International Conference on Emerging Signal Processing Applications*, Las Vegas, NV, USA, Jan. 2012, pp. 52–55. doi: 10.1109/ESPA.2012.6152444.
[62] R. Haeb-Umbach, J. Heymann, L. Drude, S. Watanabe, M. Delcroix, and T. Nakatani, “Far-Field Automatic Speech Recognition,” *Proc. IEEE*, vol. 109, no. 2, pp. 124–148, Feb. 2021, doi: 10.1109/JPROC.2020.3018668.

[63] L. Tavi, T. Alumäe, and S. Werner, “Recognition of Creaky Voice from Emergency Calls,” in *Interspeech 2019*, Sep. 2019, pp. 1990–1994. doi: 10.21437/Interspeech.2019-1253.
[64] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An ASR corpus based on public domain audio books,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, Queensland, Australia, Apr. 2015, pp. 5206–5210. doi: 10.1109/ICASSP.2015.7178964.
[65] B. Stroube, “Literary freedom: project Gutenberg,” *XRDS Crossroads ACM Mag. Stud.*, vol. 10, no. 1, pp. 3–3, Sep. 2003, doi: 10.1145/973381.973384.
[66] Ardila, R. and Branson, M. and Davis, K. and Henretty, M. and Kohler, M. and Meyer, J. and Morais, R. and Saunders, L. and Tyers, F. M. and Weber, G., “Huggingface Common Voice dataset.” Huggingface, 2020. Accessed: May 17, 2022. [Online]. Available: https://huggingface.co/datasets/common_voice
[67] C. Yi, J. Wang, N. Cheng, S. Zhou, and B. Xu, “Applying Wav2vec2.0 to Speech Recognition in Various Low-resource Languages,” *ArXiv201212121 Cs*, Jan. 2021, Accessed: May 07, 2022. [Online]. Available: http://arxiv.org/abs/2012.12121
[68] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer Normalization,” *ArXiv160706450 Cs Stat*, Jul. 2016, Accessed: Apr. 09, 2022. [Online]. Available: http://arxiv.org/abs/1607.06450
[69] D. Hendrycks and K. Gimpel, “Gaussian Error Linear Units (GELUs),” *ArXiv160608415 Cs*, Jul. 2020, Accessed: Apr. 09, 2022. [Online]. Available: http://arxiv.org/abs/1606.08415
[70] C. Cortes, M. Mohri, and A. Rostamizadeh, “L2 Regularization for Learning Kernels,” *ArXiv12052653 Cs Stat*, May 2012, Accessed: May 07, 2022. [Online]. Available: http://arxiv.org/abs/1205.2653
[71] S. Singh and A. Mahmood, “The NLP Cookbook: Modern Recipes for Transformer Based Deep Learning Architectures,” *IEEE Access*, vol. 9, pp. 68675–68702, 2021, doi: 10.1109/ACCESS.2021.3077350.
[72] D. Jiang *et al.*, “Improving Transformer-based Speech Recognition Using Unsupervised Pre-training,” *ArXiv191009932 Cs Eess*, Oct. 2019, Accessed: Apr. 12, 2022. [Online]. Available: http://arxiv.org/abs/1910.09932
[73] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *ArXiv181004805 Cs*, May 2019, Accessed: Apr. 08, 2022. [Online]. Available: http://arxiv.org/abs/1810.04805
[74] A. Baevski, M. Auli, and A. Mohamed, “Effectiveness of self-supervised pre-training for speech recognition,” *ArXiv191103912 Cs*, May 2020, Accessed: Apr. 12, 2022. [Online]. Available: http://arxiv.org/abs/1911.03912
[75] H. Jégou, M. Douze, and C. Schmid, “Product Quantization for Nearest Neighbor Search,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, Jan. 2011, doi: 10.1109/TPAMI.2010.57.
[76] E. Jang, S. Gu, and B. Poole, “Categorical Reparameterization with Gumbel-Softmax,” *ArXiv161101144 Cs Stat*, Aug. 2017, Accessed: Apr. 12, 2022. [Online]. Available: http://arxiv.org/abs/1611.01144
[77] A. van den Oord, Y. Li, and O. Vinyals, “Representation Learning with Contrastive Predictive Coding,” *ArXiv180703748 Cs Stat*, Jan. 2019, Accessed: Apr. 11, 2022. [Online]. Available: http://arxiv.org/abs/1807.03748
[78] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A Simple Framework for Contrastive Learning of Visual Representations,” *ArXiv200205709 Cs Stat*, Jun. 2020, Accessed: Apr. 11, 2022. [Online]. Available: http://arxiv.org/abs/2002.05709
[79] M. Ott *et al.*, “fairseq: A Fast, Extensible Toolkit for Sequence Modeling,”

ArXiv190401038 Cs, Apr. 2019, Accessed: Apr. 12, 2022. [Online]. Available: http://arxiv.org/abs/1904.01038

[80] A. Paszke *et al.*, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems*, 2019, vol. 32. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf

[81] A. Hannun, "Sequence Modeling with CTC," *Distill*, vol. 2, no. 11, p. 10.23915/distill.00008, Nov. 2017, doi: 10.23915/distill.00008.

[82] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning - ICML '06*, Pittsburgh, Pennsylvania, 2006, pp. 369–376. doi: 10.1145/1143844.1143891.

[83] D. S. Park *et al.*, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," *Interspeech 2019*, pp. 2613–2617, Sep. 2019, doi: 10.21437/Interspeech.2019-2680.

[84] A. Fan, E. Grave, and A. Joulin, "Reducing Transformer Depth on Demand with Structured Dropout," *ArXiv190911556 Cs Stat*, Sep. 2019, Accessed: Apr. 14, 2022. [Online]. Available: http://arxiv.org/abs/1909.11556

[85] T. Wolf *et al.*, "HuggingFace's Transformers: State-of-the-art Natural Language Processing," *ArXiv191003771 Cs*, Jul. 2020, Accessed: Apr. 16, 2022. [Online]. Available: http://arxiv.org/abs/1910.03771

[86] T. Fushiki, "Estimation of prediction error by using K-fold cross-validation," *Stat. Comput.*, vol. 21, no. 2, pp. 137–146, Apr. 2011, doi: 10.1007/s11222-009-9153-8.

[87] L. W. Johnston, "Student's *t* -Test," *J. Qual. Technol.*, vol. 2, no. 4, pp. 243–245, Oct. 1970, doi: 10.1080/00224065.1970.11980443.

[88] T. G. Dietterich, "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms," *Neural Comput.*, vol. 10, no. 7, pp. 1895–1923, Oct. 1998, doi: 10.1162/089976698300017197.

[89] B. Everitt, *The analysis of contingency tables*, 2nd ed. London ; New York: Chapman & Hall, 1992.

Appendices

I. Source Code and Instructions for Experimental Validation

The machine learning code used for data preprocessing, model pre-training, fine-tuning and evaluation in this thesis along with installation and execution instructions can be accessed at https://github.com/rootskar/speech-recognition.

II. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Karel Roots**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

"**Semi-Supervised Automatic Speech Recognition for Low Resource Languages**", supervised by Mark Fišel.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe on other persons' intellectual property rights or rights arising from the personal data protection legislation.

Karel Roots

17/05/2022