university of groningen

campus fryslân

Improving the State-of-the-Art Frisian ASR by fine-tuning Large-Scale Cross-Lingual Pre-Trained Models

Dragoș Alexandru Bălan

university of groningen

campus fryslân

University of Groningen - Campus Fryslân

Improving the State-of-the-Art Frisian ASR by fine-tuning Large-Scale Cross-Lingual Pre-Trained Models

Master's Thesis

To fulfill the requirements for the degree of
Master of Science in Voice Technology
at University of Groningen under the supervision of
Assoc. Prof. Dr. M. Coler (Voice Technology, University of Groningen)
with the second reader being
Asst. Prof. Dr. S. Nayak (Voice Technology, University of Groningen)

Dragoș Alexandru Bălan (S3944867)

July 21, 2023

Acknowledgements

I am deeply grateful to Matt Coler, my first supervisor and mentor, for his exceptional guidance, meticulous supervision, and invaluable feedback throughout the development of my research proposal and thesis. His prompt responses to my queries and his detailed critique of the drafts I submitted played an instrumental role in ensuring the timely delivery and high quality of the thesis.

I would also like to express my sincere appreciation to Tan Phat Do, my second supervisor, for providing me with crucial technical support and expertise that greatly contributed to the success of my research. I am also grateful to Shekhar Nayak with whom I brainstormed thesis ideas since the beginning of the Master's programme and for evaluating my thesis.

In addition, I am indebted to my friends and family for their unwavering love and support throughout this journey. Their encouragement and presence made this endeavor possible. I would like to extend special recognition to Golshid Shekoufandeh, my colleague and friend, who not only offered daily support but also engaged in fruitful brainstorming sessions, resulting in significant advancements in my research.

Furthermore, I would like to convey my immense gratitude to my parents, whose unwavering support has been a constant source of strength in every aspect of my life, including the completion of this thesis. Their belief in me and their encouragement have been invaluable.

I acknowledge the Center for Information Technology of the University of Groningen for their technical support and for providing access to the Hábrók high-performance computing cluster.

Lastly, I would like to express my gratitude to all the individuals who have played a part, no matter how small, in shaping my academic journey and the successful completion of this thesis.

Abstract

Frisian is a West Germanic language recognized as an official language in the Netherlands and used extensively in the province of Fryslan. Despite its official status, Frisian lacks technological support and resources, especially in the field of automatic speech recognition (ASR). Thus, it is considered a low-resource language, and low-resource language speech recognition requires alternative approaches. To enhance Frisian ASR performance and address the challenges posed by its low-resource status, this research focuses on fine-tuning the XLS-R model, a large-scale cross-lingual pre-trained model. XLS-R, built upon the wav2vec 2.0 architecture, has shown promising results in multilingual ASR tasks. It will be compared with the state-of-the-art XLSR-53 model, which has been widely used for Frisian speech recognition, to assess its potential for achieving improved word error rates and surpassing the existing performance benchmarks. Specifically, my research answers the following question: Can fine-tuning the XLS-R model on Frisian speech achieve a word error rate (WER) below 20% and outperform the state-of-the-art XLSR-53 model? Comparisons were made using a baseline WER score of 15.19%. Training the XLS-R model with the same data as the baseline (5 hours of speech) yielded a WER of 14.13%, improving the current state-of-the-art by 1.06% absolutely and 7% relatively. Further fine-tuning with approximately 8 times more data (41 hours) achieved an impressive WER of 4.11% that sets a new milestone in Frisian speech recognition and even surpasses the performance of XLS-R fine-tuned on high-resource languages. Additional experiments were conducted using 10 minutes, 1 hour, and 10 hours of training speech, resulting in word error rates of 62.25%, 25.4%, and 8.83% respectively, underscoring the importance of more data when fine-tuning large-scale models. XLS-R models with 0.3B and 2B parameters have also been fine-tuned with 41 hours of data. XLS-R with 1B is the most balanced out of the three, scoring the best WER and consuming more resources than the 0.3B model, but less than the 2B model. Future work involves using a newer version of the dataset, using the FAME! corpus, comparing the results with a different large-scale model, Whisper, as well as using language model rescoring and other metrics such as character error rate and phoneme error rate.

Contents

1 Introduction

Imagine a world where every language, no matter how small or under-resourced, could have its own automatic speech recognition system. A world where technology breaks down language barriers while preserving linguistic diversity. In a landscape where less than 1% of languages have sufficient speech resources for ASR systems (Eberhard, Simons, & Fennig, 2021), the quest to empower low-resource and zero-resource languages becomes even more crucial. Developing ASR systems for these languages is difficult because they require hundreds to thousands of hours of speech data to train models that can perform well and be robust to new input. Such languages lack the resources of higher-resourced ones and collecting more data is a tedious process that involves approvals from ethics committees to conduct the process, finding a wide range of speakers to record, as well as transcribing the speech using linguist experts. This thesis addresses this issue by developing ASR models for Frisian, a small language struggling to find its voice amidst limited speech data.

There are numerous reasons why researchers should develop more ASR models for low-resource languages. Some of them include preserving endangered languages or enhancing access to technology for people who speak low-resourced languages. Potential applications encompass initiatives such as the development of speech-enabled language learning applications for mobile devices. These mobile applications have the potential to greatly enhance language learning opportunities for individuals in low-resource language communities, particularly among younger learners who are increasingly engaged with mobile technologies. Moreover, by enabling automatic transcription and translation services, individuals who speak low-resourced languages can gain better access to online content, educational resources, and information in their native languages. This facilitates greater participation, integration, and empowerment within their communities and society at large.

A language that is considered to be low-resource and the one that will be covered by this research, as previously mentioned, is Frisian, or Frysk as the speakers call it. Frisian is a West Germanic language which is recognized as the official language in the Netherlands and spoken widely in the province of Fryslân in the Netherlands. It is also known as West Frisian, in order to distinguish it from other dialects that are spoken in certain areas in the north of Germany. Although Frisian is taught in schools and used in different media outlets like radio or TV, it still lacks support when it comes to recorded and labeled speech, which makes it difficult to develop ASR systems for it. Thus, different approaches other than data collection need to be employed in order to develop speech recognition models for such cases of under-resourced languages. An alternative is data augmentation, a process that involves the generation of additional data from pre-existing sources to enhance the quality and quantity of training data available. By applying data augmentation, a diverse range of variations can be introduced, such as modifying pitch, speed, or adding background noise, which can effectively simulate real-world scenarios and improve the model's robustness and generalization capabilities.

The focus of this research, however, is the use of transfer learning, which refers to using a model that was trained on a higher-resourced language or context and adapting the knowledge of that by fine-tuning the model on a lower-resourced language or context. Some approaches that use transfer learning are monolingual, as in training on a single high-resource language, usually English since it has the highest amount of speech data available, and then fine-tuning on a low-resource language. These models perform better than architectures trained from scratch on the low-resource language since knowledge on a phonetic level is shared from a higher-resourced language. However, the better-performing approaches involve training a model on larger data from multiple languages, thus learning how to recognize speech in a multilingual or cross-lingual context. Cross-lingual architectures manage to perform better than their monolingual counterparts due to the usage of larger amounts of data from a wider context, unrestricted to a singular language. Thus, such models are more generalizable and more robust to new data. Cross-lingual models achieve state-of-the-art performances when fine-tuned on languages with a significantly smaller number of hours of data (under 100 hours).

Two notable cross-lingual pre-trained models are XLSR-53 (Conneau, Baevski, Collobert, Mohamed, & Auli, 2020) and XLS-R (Babu et al., 2021), which are built on top of wav2vec 2.0, a self-supervised architecture that learns language-independent speech representations (Baevski, Zhou, Mohamed, & Auli, 2020). XLSR-53 is pre-trained on 56,000 hours of speech from 53 different languages and it managed to achieve state-of-the-art results in various high- and low-resource contexts at the time the paper was published. XLS-R improved upon XLSR-53 by using more parameters in its architecture (2 billion parameters versus 300 million parameters of XLSR-53) and by pre-training on 436,000 hours of data from 128 languages, thus attaining knowledge from a wider context. The two models are compared in this research since the baseline is fine-tuned on XLSR-53, whereas I fine-tuned XLS-R.

The prior work done in developing ASR for Frisian has been done for the FAME! (Frisian Audio Mining Enterprise) dataset, a corpus of bilingual Frisian-Dutch radio broadcasts (Yılmaz, Andringa, et al., 2016). However, a monolingual and larger dataset exists for Frisian, which is Mozilla's Common Voice project, a crowdsourced corpus containing over 100 languages (Ardila et al., 2020). There have been models that have been trained on this corpus; they mainly use XLSR-53 as the underlying model (Conneau et al., 2020). The best-performing Frisian ASR models on Common Voice Frisian are determined to be the works of de Vries (2021) and Crang (2021). Both of the models were fine-tuned in parallel by their respective authors in 2021, but de Vries (2021) has managed to achieve a lower word error rate (WER), the metric used in speech recognition evaluation. The developer's WER was 16.25%, which corresponds to the current state-of-the-art in Frisian ASR. I plan to improve upon the current state-of-the-art by fine-tuning the larger-scale XLS-R model, analyzing the amount of data needed to achieve significant performance gains and considering what number of parameters in the XLS-R model determines the perfect balance between the duration of training or evaluation and the model's performance. To that end, I articulate the scope of this thesis with a research question with four subquestions and a hypothesis below.

Now that a brief motivation for this research has been presented, the structure of the thesis is the following: subsection 1.1 introduces the research question posed along with a hypothesis on the outcome of the research. Section 2 provides an extensive literature review that frames the research question and hypothesis in the state-of-the-art. In section 3, the methodology is covered and the underlying models used are explained. Then, section 4 describes the experimental setup developed to answer the research questions and validate the hypothesis. Section 5 describes the results obtained and compares them to the baseline. In section 6, I discuss the previously-mentioned results in detail. Lastly, section 7 summarizes the thesis and presents the conclusions drawn, along with recommended future work.

1.1 Research Question and Hypothesis

In light of the preceding discussion, the research question at the core of this study can be formulated as follows:

> **Can fine-tuning the XLS-R model on Frisian speech achieve a WER below 20% and outperform the state-of-the-art XLSR-53 model?**

From which the following subquestions are derived:

- What is the baseline WER achieved by fine-tuning XLSR-53 on Frisian speech?
- Can the XLS-R model with 1B parameters achieve a lower WER than XLSR-53 on Frisian speech?
- How does the size of the training data used for fine-tuning affect the performance of the model on Frisian speech?
- What number of parameters of the XLS-R model achieves the best balance between performance and duration of training or inference?

My hypothesis is that fine-tuning the XLS-R model with 1B parameters on Frisian speech using specific hyperparameters and a sufficient amount of training data will result in a WER below 20% and, at the same time, outperform the state-of-the-art XLSR-53 model. Furthermore, I hypothesize that the training data size will significantly impact the model's performance. The more data that will be used for training, the better the performance of the model will be. In addition, the XLS-R model with 1 billion parameters is the one expected to be the most balanced when it comes to duration and performance.

Word Error Rate, or WER, is a metric commonly used in ASR for evaluation. It measures the error rate of the sentence predicted by an ASR model compared to a reference sentence and it calculates the error rate at a word level. A lower WER results in a better-performing model that predicts words with very few errors.

The latest developments in low-resource transfer-learning ASR involve models trained on multiple languages, also known as multilingual or cross-lingual models, which leverage massive amounts of data (tens to hundreds of thousands of hours) and use it effectively for low-resource languages (Babu et al., 2021; Conneau et al., 2020; Radford et al., 2022; Y. Zhang et al., 2023). The most relevant latest research found is that of XLS-R, a set of models trained on 436,000 hours of speech in 128 different languages, both high- and low-resourced (Babu et al., 2021). The quantity and variety of data used to pre-train the models prove to outperform the previous version of this model trained on fewer data and fewer languages, XLSR-53 (Conneau et al., 2020). The higher number of parameters used in the XLS-R model, 2 billion or 1 billion parameters, leads to better performance compared to the 300 million parameters models (Babu et al., 2021), but no benchmarking has been done in the initial paper regarding the time and resources it would take to train or evaluate models with higher numbers of parameters. However, given that XLS-R with 2B parameters performs slightly better than XLS-R with 1B parameters, it is expected that the version with 1B parameters would significantly reduce training/evaluation times with the only drawback being slightly worse performance.

There are already several XLSR-53 models fine-tuned on Frisian that manage to achieve results below 20% WER (Crang, 2021; de Vries, 2021), therefore I expect to also achieve a comparable, if not lower, error rate that would also fall under the 20% threshold set by the previous work of de Vries (2021) and Crang (2021). Furthermore, the current state-of-the-art for Frisian ASR is the work of de Vries (2021), who achieves a WER of 16.25%. I anticipate outperforming that score by fine-tuning XLS-R on Frisian, with an absolute WER improvement of 1-2% from the baseline, similar to the results in the XLS-R paper (Babu et al., 2021).

The falsification of the hypothesis would suggest that fine-tuning the XLS-R model is not an effective approach for improving Frisian ASR and that the XLSR-53 model may be a more suitable choice for this. It could also partly contest the work of Babu et al. (2021); Radford et al. (2022); and Y. Zhang et al. (2023), implying that pre-training larger and larger models does not necessarily translate to better results.

2 Literature Review

This section is dedicated to providing a comprehensive review of the existing research pertaining to the fine-tuning of large-scale cross-lingual models for automatic speech recognition (ASR), with a specific focus on addressing the unique challenges posed by the low-resource Frisian language. By conducting a thorough and critical analysis of the literature in this field, this review aims to offer valuable insights into the use of large-scale cross-lingual models for ASR in low-resource language settings, with Frisian serving as an illustrative case study.

To those ends, the section is structured as follows. To begin, I will delineate the keywords used during the comprehensive literature search described above and describe the inclusion/exclusion criteria used in selecting the literature. After that, I offer a succinct overview of the key findings and contributions of the selected papers (in subsections 2.1-2.11).

I have grouped the keywords according to the topic they are related to. The topics are highlighted in bold, after which the keywords for that topic are mentioned. The search has been mainly conducted on Google Scholar, as well as on Hugging Face to find the baseline Frisian ASR models. Most of the XLSR models outputted by various research groups have been uploaded to Hugging Face. Thus, the topics and their corresponding keywords are:

- **Datasets:** Frisian speech, Frisian speech corpus, Frisian speech dataset;

- **Transfer learning:** transfer learning ASR, transfer learning speech recognition;

- **Cross-lingual models:** cross(-)lingual speech (recognition), cross(-)lingual ASR, multilingual speech (recognition), multilingual ASR;

- **Frisian ASR:** Frisian ASR, Frisian (automatic) speech recognition.

To streamline the paper selection process, I organized the papers based on their relevance to specific topics and keywords. However, not all retrieved literature was directly related to Frisian ASR. Therefore:

1. To maintain coherence, I excluded papers that pertained to different tasks such as speech emotion recognition or speech synthesis;

2. To ensure the inclusion of recent research, the publication dates were limited to papers from 2012 onwards. This decision was made to reflect the latest advancements and methodologies in the field of ASR using neural networks in a cross-lingual context since the earliest works date back to 2012;

3. To prioritize the most influential works, I selected the top 20 articles according to their relevancy on Google Scholar.

By applying these filters and exclusion criteria, I aimed to ensure the inclusion of the most pertinent and up-to-date literature directly related to Frisian ASR, aligning with the research objectives and scope of this study

The literature review is organized into different subsections, based on the general topic of which they are part. Subsection 2.1 discusses the literature regarding Frisian speech datasets. Transfer learning for different contexts, particularly for children ASR, is presented in subsection 2.2. Moving towards low-resource language speech recognition, subsection 2.3 covers different monolingual models that have been fine-tuned for other languages. Further on, the early research into cross-lingual models using deep neural networks and later on recurrent and sequence-to-sequence networks are discussed in subsections 2.4 and 2.5. In subsection 2.6, the more recent literature regarding cross-lingual models is considered. The speech feature extraction framework underlying the models used in the methodology, wav2vec 2.0, and its predecessors are surveyed in subsection 2.7. The models used in experiments, as well as other state-of-the-art large-scale cross-lingual models, are analyzed in subsections 2.8 and 2.9. Alternative optimized fine-tuning of large models is discussed in subsection 2.10. Lastly, the relevant literature and models regarding Frisian speech recognition are reflected upon in subsection 2.11.

For simplicity and readability, tables 1 and 2 provide a full list of references appended with some notes, sorted by order of appearance in the following subsections of the literature review.

2.1 Frisian Speech Datasets

Firstly, the literature I have found regarding the available speech corpora for Frisian consists of mainly two references: the Frisian Audio Mining Enterprise (FAME!) corpus (Yılmaz, Andringa, et al., 2016) and Common Voice (Ardila et al., 2020). FAME! is a Frisian-Dutch bilingual corpus of annotated speech collected from radio broadcasts. The quantity of data available in the corpus is 18.5 hours and most of the recordings involve code-switching between Frisian and Dutch. The other corpus, Common Voice, is an open-source, crowdsourced multilingual project with over 100+ languages available designed for speech recognition purposes. The users can contribute to the project by recording themselves speaking a given prompt or by validating other users' recordings. The downside is that the recordings can be noisy and the validation depends on the users that contribute to the corpus rather than trained experts. However, the availability and quantity of data make it a corpus worth considering. Therefore, I will work with Common Voice since it has more speech available than FAME! and it contains exclusively monolingual Frisian recordings. More information about the corpus will be provided in section 4.

2.2 Transfer Learning for Different Contexts

Transfer learning for the task of speech recognition is a powerful and robust tool that can be applied in various contexts. One such context is developing ASR for children, as illustrated by the works of R. Tong, Wang, and Ma (2017), Matassoni, Gretter, Falavigna, and Giuliani (2018), and Shivakumar and Georgiou (2020).

R. Tong et al. (2017) compare acoustic adaptation and multi-task learning techniques to develop Mandarin and English ASR for both adult and children's speech. Acoustic adaptation refers to training a baseline acoustic model on adult Mandarin data in this case, then tuning the obtained acoustic model on the children's data. Multi-task refers to training a set of shared hidden layers with data from both adults and children, but fine-tuning separate output layers for each type of task (adult ASR or children ASR). The authors demonstrated that using either of the techniques provides significant improvements over the baseline trained on either adult Mandarin speech or children American English speech. What is also important is that multi-task learning performs better compared to acoustic adaptation. Multi-task learning also manages to improve over the baseline Mandarin model for adult speech in contrast to acoustic adaptation which performs worse in that category. However, both transfer learning techniques are useful for developing ASR for children compared to training only on children speech data.

Matassoni et al. (2018) used different transfer learning techniques to develop ASR for non-native Italian children speaking German and non-native Italian and German children speaking English. They have proved that any sort of model adaptation or multilingual context helps with improving the WER for non-native speech, although, in doing so, native speech performed worse. This shows that there are trade-offs that one should consider when attempting to develop a multilingual system that performs well for both native and non-native speakers, but if we are to focus on non-native speech recognition only, then transfer learning offers the best results and is a more robust option.

Finally, Shivakumar and Georgiou (2020) conducted a thorough analysis of different adaptation methods, based on age, for children ASR. They first show that transfer learning a model trained on adult data is more robust and provides better results, as also confirmed by R. Tong et al. (2017) and Matassoni et al. (2018). What they also add as observation is that bottom-layer adaptation proves to be more efficient than top-layer adaptation because the acoustic characteristics of a child's voice are different than that of an adult's and the bottom layers of a deep neural network usually capture acoustic variability rather than linguistic content. Since the language of both adult and children datasets is the same, but the voice characteristics are different, bottom layer adaptation has better performance. The authors also state that more data is required the smaller the age is, therefore the closer a child is to adult age, the fewer data is required for fine-tuning to obtain good results due to the vocal tract being more developed than that of, for example, a five-year-old.

2.3 Monolingual Transfer Learning

Since this research focuses on low-resource ASR, it is important to review the literature regarding this task. Research has been done on transferring a monolingual ASR to a low-resource target language. One example is the work of Kunze et al. (2017) where transfer learning is used to develop a German ASR system in constrained GPU memory by converting a pre-trained English model. The authors demonstrate that an English ASR model fine-tuned on German performs better than a model trained from scratch only on German data. Furthermore, they observe that the outer layers of a model need to change more than the inner layers when the model is learning a new language and that freezing the bottom layers when training allows for computations with a GPU memory as low as 5.5 GB available. However, the error rates obtained are quite high compared to the latest developments, which shows how rapidly the field has evolved in the past 2 years. The authors report a WER of 42.49% and a letter error rate (LER) of 15.05% when using a language model.

Three years later, J. Huang et al. (2020) manage to fine-tune different English models for different tasks, obtaining impressive results. The authors fine-tune models pre-trained on different large English corpora, such as LibriSpeech or Wall Street Journal. The tasks concern different English accents (Singapore or African American), as well as different languages (German, Spanish, and Russian). In all experiments, the fine-tuned versions performed better than the baselines trained from scratch on the respective datasets with absolute WER improvements ranging from 5% (German and Spanish fine-tuning) to 34% (African American English fine-tuning). In contrast to Kunze et al. (2017), the authors do not freeze any layers when fine-tuning but rather continue the training process after the model has been trained on the English datasets, without any other modifications.

Research has been done to investigate the influence of the source language from which we transfer knowledge. Hjortnæs, Partanen, Rießler, and Tyers (2021) trained two different models for Zyrian Komi speech recognition which, according to them, the language is ”an endangered, low-resource Uralic language spoken in Russia”. The first model is pre-trained on English data from Common Voice (Ardila et al., 2020) and then fine-tuned on Komi, whereas the second model is pre-trained on Russian from the same dataset and fine-tuned on Komi. The English dataset has over 1400 hours of speech whereas the Russian dataset has 103 hours available. They used Russian because Komi is more related to Russian than to English. The results obtained, however, indicate that the amount of data available in the source language is more relevant than relatedness to the target language, with the English model scoring a WER of 76.5% and the Russian model scoring 83%.

Monolingual transfer learning work has been done for Czech ASR as well, in the paper of Polák and Bojar (2021). The authors experiment with three different configurations. The first one is a pre-trained English model fine-tuned on Czech, where the encoder is frozen and the decoder is rapidly trained for the first 1500 steps, then the encoder is unfrozen and the whole model is subsequently trained. The second configuration involves training on a simplified Czech vocabulary that matches the English one for 39000 steps, then doing the same steps as in the first configuration. This configuration is coined as ”a simple version of coarse-to-fine training”. Coarse-to-fine refers to the vocabulary that is used for the first step of training (a ”coarse” vocabulary consisting of a smaller set of characters) which is then refined for the latter steps of the training (use the full vocabulary rather than a simplified set). The last setup uses the same simplified Czech vocabulary but is instead trained on a randomly-initialized network rather than the English model previously mentioned, for 39000 steps. Then, the same steps are followed as in the first configuration. All types of transfer learning have shown promising results, with the last coarse-to-fine configuration performing the best in terms of WER (an absolute WER improvement of $\approx 3.5\%$ over the baseline).

All of the research done above is key to highly-performant low-resource ASR. However, even more research has been conducted in cross-lingual ASR where one model pre-trained on a set of different languages can be fine-tuned for any low-resource language to achieve remarkable results by leveraging a wider area of linguistic knowledge.

2.4 Early DNN-Based Works in Cross-Lingual ASR

Cross-lingual ASR is a subfield that has shown more promise recently with the developments of more powerful hardware and, subsequently, end-to-end models that can encode speech in meaningful representations. However, work has been done in this subfield as early as 2012 by Swietojanski, Ghoshal, and Renals (2012). In this research, two types of configurations that involve deep neural networks (DNNs) are used: a tandem configuration, where the output of a DNN is used in concatenation with other features as the input to a GMM-HMM model, and a hybrid DNN-HMM configuration where the DNN outputs tied triphone states. In both configurations, the DNN used is pre-trained using unsupervised restricted Boltzmann machines (RBMs), which have been shown to be language-independent, as well as significantly improve the WER by $\approx 5\%$ for hybrid systems. Hybrid DNN-HMMs performed better than the tandem systems by a difference of $\approx 2\%$. This research has motivated further work to be done on cross-lingual speech recognition using hybrid DNN-HMM systems and replace the previous state-of-the-art GMM-HMM models.

In the next year, J.-T. Huang, Li, Yu, Deng, and Gong (2013) have achieved to develop one model that can be used for speech recognition in different languages. In this paper, the authors employ a deep neural network with five hidden layers, 2048 units each, and several different softmax output layers. The hidden layers are trained on multiple languages in parallel (the hidden layers are shared), but the several output layers used are fine-tuned on different languages (one layer for each language). The architecture was coined Shared-Hidden-Layer Multilingual Deep Neural Network or SHL-MDNN. In the paper, the authors train the model on four different European languages (French, German, Spanish, and Italian) and then fine-tuned on a corpus closely related to the aforementioned languages (American English), as well as a corpus more distant from the European languages (Chinese). For both datasets, the results improve over the baseline (an absolute WER difference of 4.6% on American English and an absolute CER difference of 2.4% on Chinese).

Other models that have been researched in parallel, in the same year, are a DNN-HMM hybrid system where the bottom three hidden layers of the DNN were trained on data from multiple languages (Heigold et al., 2013) and a DNN architecture developed by the same team as Swietojanski et al. (2012) where the hidden layers are trained sequentially, one language after another (Ghoshal, Swietojanski, & Renals, 2013). This research from three different groups has set the groundwork for cross-lingual ASR.

2.5 LSTM- and Seq2seq-Based Cross-Lingual Models

Based on J.-T. Huang et al. (2013), Zhou, Zhao, Xu, Xu, et al. (2017) developed a cross-lingual model related to SHL-MDNN named Shared-Hidden-Layer Multilingual Long Short-Term Memory or SHL-MLSTM. As it can be observed from the name, authors replace the DNN element with long short-term memory (LSTM) networks. LSTMs are powerful due to their memory blocks that allow them to remember long-term dependencies, as well as containing several types of gates that control the flow of past and future information. Due to these blocks and gates, LSTMs are more suitable for use for speech recognition purposes than DNNs since speech is a context-dependent process. Aside from using LSTMs, the authors also employ residual learning. Two configurations are used in the experiments of the paper: one without residual learning, and the other with residual learning. Both configurations outperform the SHL-MDNN baseline. The setup with residual learning obtains the best scores, improving over the setup without residual learning by $\approx 1\%$ WER and over the baseline by $\approx 4\%$ WER. The error rates are still quite high but, given that the setups were trained and tested in a low-resource context, they show the potential of LSTMs in contrast to DNNs for cross-lingual and low-resource ASR models.

In the following year, Dalmia, Sanabria, Metze, and Black (2018) employed an end-to-end, sequential model using a Connectionist Temporal Classification (CTC) loss. A speech recognition model trained with CTC loss attempts to align an input sequence of acoustic features to an output sequence of labels or characters by finding the best CTC path to match the 2 sequences. Bidirectional LSTM layers are used as the shared encoder that is trained on all of the multilingual data and language-dependent softmax output layers, along with the CTC loss, decode the encoded features. What they observe is that using large data from multiple languages or large clean data from one language, namely English, provides better results in very low-resource contexts compared to monolingual baselines.

In parallel to Dalmia et al. (2018), S. Tong, Garner, and Bourlard (2017) also developed a CTC model with bidirectional LSTMs. Their goal is to be able to learn to decode phonemes unseen in the training set. They do so by using Learning Unit Hidden Contribution (LHUC) that aids with recognizing a specific language. Their experiments show that a multilingual model without LHUC performs worse than monolingual models, but performs better when using LHUC. Moreover, by employing LHUC in all layers when fine-tuning on Portuguese, the performance of the model is significantly better than the monolingual baseline. They also show that phoneme coverage is important and the more phonemes are covered, the better the model will adapt to a language with unseen phonemes during training. Lastly, using dropout is important to avoid overfitting when adapting multilingual models. All of the experiments prove the power of multilingual models when fine-tuned on low-resource, unseen languages.

Lastly, a sequence-to-sequence (seq2seq) model using the Listen, Attend and Spell (LAS) framework has been trained on nine Indic languages by Toshniwal et al. (2018). In their paper, they experiment with jointly training the model on all data from the nine languages, without explicit language identification mention, multi-task learning where the model learns recognition of words, as well as of the language spoken, and conditioning where the model is explicitly conditioned during inference with a language ID. All experiments show improvements of the multilingual models over their monolingual counterparts, with the language-conditioned models performing the best. In most cases, the multi-task model would perform better than the joint one, but in some the joint one manages to obtain better results. Overall, their proposed cross-lingual models outperform the monolingual models and, thus, leveraging more data from a wider range of languages creates models capable of recognition in various low-resource contexts.

2.6 Recent Large-Scale Cross-Lingual Work

More recent work towards large-scale cross-lingual models has been done by Fukuda and Thomas (2021) and Z.-Q. Zhang, Song, Wu, Fang, and Dai (2021), where two networks are used during training in both cases. In Fukuda and Thomas (2021), the authors employ knowledge distillation as a method for training more robust cross-lingual models. What is meant is that several teacher networks are trained, one per language, which are then used by one simpler student model to learn the correct labels. After that, two shuffles are inserted: one shuffles the data around such that speech from a different language than a specific target network is used, and one where the language-dependent layers are switched between each other in the source model. This encourages the source student model to learn language-independent phonemes and representations which provides better adaptability to an unseen language. They show that their method outperforms conventional transfer learning where one model is trained and fine-tuned.

Aside from that, XLST is another speech representation extraction framework that uses a target network that is fixed and non-trainable and a main network that is trained based on the output of the target network (Z.-Q. Zhang et al., 2021). Instead of using a contrastive loss during training, the authors use a similarity function between the main and target networks which simplifies training and leverages prior knowledge of the fixed target network. In their experiments, the XLST models pre-trained on thousands of hours of English speech and fine-tuned for specific languages perform better than the equivalent XLSR-10 counterparts, using a tiny amount of labeled pre-training data.

2.7 Speech Feature Extraction Frameworks

The subfield has seen exponential growth in the past three years. This is mainly attributed to advancements in unsupervised and self-supervised speech feature representation, as well as to the rapid growth in data and computational power. The first unsupervised speech representation extraction model is wav2vec (Schneider, Baevski, Collobert, & Auli, 2019). Wav2vec takes as input raw speech and feeds it through two convolutional neural networks, one context network that is stacked on top of an encoder network. The authors then use the outputted representations as input to the acoustic model of an ASR. The framework is trained using a contrastive loss that tries to distinguish a future sample from a set of distractors and uses the outputs of both the context and encoder networks. They show that a model that uses representations from wav2vec instead of other feature methods achieves a lower WER.

The following iteration, vq-wav2vec (vector quantization wav2vec), improves upon the wav2vec architecture by adding a quantization module which discretizes the output of the encoder network (Baevski, Schneider, & Auli, 2020). Using the quantized representations together with the contextualized features, the model is trained with a contrastive loss function similar to the original wav2vec architecture. In addition, for the output layer, the research team employs Gumbel-softmax, as well as k-means, and compare between the 2 techniques. They show that, under no language model, the Gumbel-softmax model performs better than the baseline. Nevertheless, the authors improved upon their past attempt with a new model that produces quantized representations which result in a contrastive loss reduction when training.

The successor of wav2vec, wav2vec 2.0, obtains features from raw speech in a self-supervised manner (Baevski, Zhou, et al., 2020). It does so by replacing the context network with a transformer architecture and by using an additional quantization module, similar to vq-wav2vec, which trains the model together with the output of the transformer by attempting to minimize a contrastive loss between the two representations. Given the past success of Gumbel-softmax in the experiments of vq-wav2vec, the authors employ this output layer in wav2vec 2.0. Paired with a language model on top, it manages to achieve below 10% WER on large datasets, even when using as low as 10 minutes of labeled data.

2.8 Large-Scale Cross-Lingual Models Based on wav2vec 2.0

Wav2vec 2.0 was used in training XLSR-53, one of the first and most popular large-scale cross-lingual ASR models (Conneau et al., 2020). The authors have managed to draw several conclusions after analyzing the results obtained on different datasets, but the most important is that low-resource languages can benefit greatly from a model trained on large multilingual data. When the model is pre-trained on all data from 53 languages and then fine-tuned for the target low-resource language, it outperforms monolingual models. A downside is found for high-resource languages which suffer from interference due to the model being trained on scarce data from other languages. Since this study is aimed at improving low-resource ASR, the weakness of XLSR-53 is not as relevant as its strength.

It is worth mentioning that Conneau et al. (2020) also pre-trained a model on 10 languages, namely XLSR-10, and compared this model and XLSR-53 with monolingual models. The monolingual models manage to perform better for high-resource languages (Spanish, French, Italian) on Common Voice (Ardila et al., 2020) when compared to XLSR-10, but XLSR-10 performed better for lower-resourced languages, and XLSR-53 performed the best for all languages, no matter the amount of pre-training data available for them.

The same research team worked further to create XLS-R, an even larger model than XLSR-53 (Babu et al., 2021). Trained on almost half a million hours of data from 128 languages, it manages to achieve state-of-the-art results on speech translation, speech recognition, and language identification tasks. It outperforms the previously-mentioned model on both low- and high-resource data, which makes it a powerful model to use for developing a Frisian ASR system.

Gupta et al. (2021) extend the work of Conneau et al. (2020) by pre-training a large-scale wav2vec 2.0 base model for 23 Indic languages. The model's name is CLSRIL-23 (Cross Lingual Speech Recognition for Indic Languages). The research group uses the base wav2vec 2.0 checkpoint pre-trained on LibriSpeech (Baevski, Zhou, et al., 2020) and pre-train two models: a multilingual model trained on data from all 23 languages and a monolingual one on Hindi, the language with the most data available (over a half of the multilingual training data). In their experiments, they tested the models on four languages, one of them being Hindi. The multilingual model fine-tuned on each language performed better than the fine-tuned monolingual model, even on Hindi which was also used in the monolingual model and is significantly higher-resourced than the other 22 languages.

2.9 Other Large-Scale Cross-Lingual Models

There have been other large-scale cross-lingual models released by other research groups. Some notable examples are mSLAM, Whisper, and Google USM (Bapna et al., 2022; Radford et al., 2022; Y. Zhang et al., 2023).

Bapna et al. (2022) have developed mSLAM, a multilingual model built upon the speech-text preprocessing framework SLAM (Bapna et al., 2021). SLAM consists of several Conformer layers from which both speech and text representations are extracted then used in recognition models. In a supervised context, the authors use CTC, similar to XLS-R (Babu et al., 2021). They also train on almost the same datasets as XLS-R, except they do not include VoxLingua. Therefore, the model is pre-trained on 51 languages instead of 128. Regardless, it achieves better performance than XLS-R.

Whisper is a sequence-to-sequence model based on an encoder-decoder Transformer architecture with cross-attention (Radford et al., 2022). It is trained on 680,000 hours of data from 96 languages and it manages to improve upon both mSLAM and XLS-R. The most recent cross-lingual model released, Google USM, is trained on 12 million hours of speech from 300 languages and using a convolution-augmented Transformer (Conformer) architecture (Y. Zhang et al., 2023). It established state-of-the-art results on different high and low-resource tasks.

While all three frameworks, mSLAM, Google USM, and Whisper, demonstrate promising outcomes and exhibit enhanced performance in comparison to XLS-R, certain limitations exist when considering their applicability to the research conducted. Firstly, mSLAM and Google USM lack publicly available pre-trained models or code, which hinders reproducibility and limits their accessibility for further exploration. Moreover, Whisper, while a noteworthy framework, diverges significantly from the baseline XLSR-53, making a direct comparison less suitable. In contrast, XLS-R, the baseline model, offers extensive tutorials, robust support, and a training methodology utilizing wav2vec 2.0, aligning more closely with XLSR-53. Therefore, choosing XLS-R as the basis for comparison is a more appropriate decision for this research.

2.10 Optimized Networks for Cross-Lingual Speech Recognition

Following the release of many large-scale cross-lingual models, research has been conducted in optimizing these networks and then fine-tuning them for various tasks. The works of Hou et al. (2021), Lu, Huang, Qu, Wei, and Ma (2022), and Yang et al. (2023) illustrate this the best. In Hou et al. (2021), the authors cover parameter-efficient adaptation, a technique used during fine-tuning where only a sizably smaller number of parameters are optimized rather than the full network. Two methods are used in the paper to achieve this: MetaAdapter, based on meta-learning for fast adaptation, and SimAdapter which uses an attention mechanism to find similarities between the target low-resource language and the source language adapters the model was trained on. They also combine the two methods and name the resulting adapter SimAdapter+. Their results show that fine-tuning less than 20% of the model's parameters brings significant improvements over full fine-tuning or baseline models. MetaAdapter is the fastest in both training and inference while SimAdapter performs slightly worse than full fine-tuning, but still at an acceptable level. Therefore, adapters are one useful technique that can be used in fine-tuning optimization.

Lu et al. (2022) adopt sparse language-dependent sub-networks extracted from large-scale pre-trained models, which are then combined to produce a sparser but more efficient cross-lingual model. They denote the model as S3Net. XLSR-10 is used in the paper as the architecture from which extraction is performed. The extracted sub-networks are combined in such a way that specific connections that belong only to one sub-network will be used by speech from the specific language for which the network was extracted, whereas the other connections are shared between the different languages, thus resulting in a sparse language-adaptive model. It manages to outperform XLSR-10, used as the baseline, in all of the experiments.

Finally, neural reprogramming can be used to optimize models for mono and cross-lingual tasks (Yang et al., 2023). With the added reprogramming blocks into the neural networks, the model can be trained on less parameters and thus speed up computation time. The authors show for different tasks that attention-based reprogramming with bridged connections manages to achieve comparable results with less than 10% of parameters used during training.

2.11 Frisian Speech Recognition

The first and most well-known research in Frisian ASR has been done by Yılmaz, van den Heuvel, and Van Leeuwen (2016) whose researchers have also participated in creating the FAME! corpus (Yılmaz, Andringa, et al., 2016). In this paper, the authors train a model similar to that of J.-T. Huang et al. (2013) to recognize code-switched Dutch-Frisian speech. The setup consists of two versions: one where there are separate output layers for Frisian and Dutch respectively, the same as SHL-MDNN (language-dependent), and the other where only one shared output layer is used (language-independent). The authors obtain the best performance when using a language-dependent output layer, scoring a WER of 36.3%. The difference between language-dependent and language-independent is marginal since the language-independent output layer scores a 36.4% WER.

San et al. (2023) conducted recent research on the FAME! corpus, exploring multiple languages including Frisian, Gronings (spoken in the province of Groningen, Netherlands), and Besemah and Nasal (Malayo-Polynesian languages from Sumatra, Indonesia). Their study aligns with the objectives of my own as they also fine-tuned the XLS-R model on different languages. Notably, their findings reported a WER of 35.2% for Frisian using a 4-gram language model trained on the entire

Frisian corpus. Although this paper is closely related to the scope of this contribution, the fact that the dataset used in their study is different prevented its use as a baseline for my experiments.

The current state-of-the-art for Frisian speech recognition on the Common Voice dataset (Ardila et al., 2020), as mentioned in the introduction, is the work of de Vries (2021). De Vries manages to achieve a WER of 16.25% by fine-tuning XLSR-53 on the Frisian subset of Common Voice. A similar attempt was made in the same period by a different author. Crang (2021) performed worse by scoring a higher WER than de Vries (2021), which is 19.11%. The main difference in these two attempts is that de Vries applies a 10% activation layer dropout probability, whereas Crang does not apply any dropout. Since de Vries (2021) achieves a lower WER, I will be using his model as the reference baseline for my experiments.

This concludes the literature review section, which provides an extensive overview of the prior research in transfer learning, cross-lingual ASR, and the current state-of-the-art in Frisian ASR. We have looked at transfer learning in the context of children ASR, as well as monolingual transfer learning from one high-resource language, such as English, to a lower-resourced language. I emphasized cross-lingual transfer learning by covering its history, starting with DNN and LSTM models, and ending up with more recent work. The more recent cross-lingual work involves various large-scale models, such as mSLAM, Whisper, and Google USM, or techniques like knowledge distillation, parameter-efficient adaptation, or sparse sub-networks. Emphasis has been placed on the speech feature extractor, wav2vec 2.0, and on large-scale models that are based on it, such as XLSR-53 and XLS-R, used in the methodology. Lastly, research done on Frisian speech recognition on both FAME! and Common Voice has been discussed where the baseline has been introduced, which is XLSR-53 fine-tuned on Common Voice Frisian by de Vries (2021) that has a reported word error rate of 16.25%.

While previous studies have contributed to our understanding of Frisian speech recognition, there are still gaps in knowledge and room for improvement that need to be addressed, which is what I attempt to do. Specifically, newer and larger cross-lingual models can be used to achieve better results. Therefore, I investigate fine-tuning a larger-scale cross-lingual model, XLS-R (Babu et al., 2021), and in doing so outperforming the current state-of-the-art Frisian ASR trained on XLSR-53 (de Vries, 2021). I also analyze the impact of the amount of training data on the performance of XLS-R similar to Babu et al. (2021); Baevski, Zhou, et al. (2020); Conneau et al. (2020), as well as benchmark XLS-R with different numbers of parameters in order to find the optimal balance between training or inference durations and performance. The methodology that helps address this knowledge gap will be described in the following section.

Table 1: List of references for subsections 2.1-2.5, summarized

Reference	Brief description	Subsection
Yılmaz, Andringa, et al. (2016)	FAME!, Dutch-Frisian code-switched speech corpus	2.1
Ardila et al. (2020)	Common Voice, massive multilingual crowdsourced corpus, used in experiments	2.1
R. Tong et al. (2017)	Mandarin & English children ASR using acoustic adaptation and multi-task learning	2.2
Matassoni et al. (2018)	Speech recognition on non-native children speaking German or English	2.2
Shivakumar and Georgiou (2020)	Analysis of transfer learning for different ages of children	2.2
Kunze et al. (2017)	English ASR model fine-tuning on German under constrained GPU memory	2.3
J. Huang et al. (2020)	English ASR model fine-tuned for different accents of English or for different languages	2.3
Hjortnæs et al. (2021)	Source language relevance in monolingual transfer learning	2.3
Polák and Bojar (2021)	Coarse-to-fine monolingual transfer learning Czech ASR using simplified vocabulary	2.3
Swietojanski et al. (2012)	Earliest cross-lingual DNN models with unsupervised RBM pre-training	2.4
J.-T. Huang et al. (2013)	SHL-MDNN, early attempt at cross-lingual ASR, hidden layers shared between languages, multiple output layers for each language	2.4
Heigold et al. (2013)	Early attempt at cross-lingual ASR, DNN-HMM with bottom 3 hidden layers shared	2.4
Ghoshal et al. (2013)	Early attempt at cross-lingual ASR, trained sequentially, a language after another	2.4
Zhou et al. (2017)	SHL-MLSTM, similar to SHL-MDNN, replaces DNNs with LSTM networks	2.5
Dalmia et al. (2018)	end-to-end CTC model using bidirectional LSTMs	2.5
S. Tong et al. (2017)	end-to-end CTC model with LHUC	2.5
Toshniwal et al. (2018)	Seq2seq model using LAS, trained on 9 Indic languages	2.5

Table 2: List of references for subsections 2.6-2.11, summarized

Reference	Brief description	Subsection
Fukuda and Thomas (2021)	Cross-lingual model using knowledge distillation	2.6
Z.-Q. Zhang et al. (2021)	XLST, cross-lingual framework, uses 2 networks during training	2.6
Schneider et al. (2019)	wav2vec, unsupervised learning of speech representations	2.7
Baevski, Schneider, and Auli (2020)	vq-wav2vec, adds quantization module to wav2vec	2.7
Baevski, Zhou, et al. (2020)	wav2vec 2.0, self-supervised learning of speech representations, used as underlying model of XLSR-53 and XLS-R	2.7
Conneau et al. (2020)	XLSR-53, large-scale cross-lingual model, used in state-of-the-art Frisian ASR	2.8
Babu et al. (2021)	XLS-R, larger-scale cross-lingual model than XLSR-53, used in experiments	2.8
Gupta et al. (2021)	CLSRIL-23, large-scale cross-lingual model similar to XLSR-53, trained on 23 Indic languages	2.8
Radford et al. (2022)	mSLAM, multilingual model built upon the speech-text preprocessing framework SLAM	2.9
Radford et al. (2022)	Whisper, larger-scale cross-lingual model than XLS-R	2.9
Y. Zhang et al. (2023)	Google USM, latest large-scale cross-lingual model, state-of-the-art results	2.9
Hou et al. (2021)	SimAdapter, used for optimized training of large models	2.10
Lu et al. (2022)	S3Net, uses sparse language-dependent sub-networks for optimized large cross-lingual ASR model	2.10
Yang et al. (2023)	Attention-based neural reprogramming for fine-tuning ASR models with fewer parameters	2.10
Yılmaz, van den Heuvel, and Van Leeuwen (2016)	First Frisian ASR, WER of 36.3% on FAME!	2.11
San et al. (2023)	Latest Frisian ASR, WER of 35.2% on FAME!	2.11
de Vries (2021)	State-of-the-art Frisian ASR, WER of 16.25% on Common Voice 8.0	2.11
Crang (2021)	Similar attempt to state-of-the-art Frisian ASR, WER of 19.11% on Common Voice 8.0	2.11

3 Methodology

In this section, I will outline the methodology used to address the research question and validate the hypothesis on a high-level. First, in subsection 3.1, I will discuss the dataset utilized for training and testing the models. Next, subsection 3.3 will focus on the feature extractor employed in the models, namely wav2vec 2.0. Following that, in subsection 3.2, I will delve into the XLSR models and provide a comparative analysis. Subsection 3.4 will then elaborate on the evaluation method and metric employed, specifically the word error rate. Finally, in subsection 3.5, I will reflect on the ethical considerations inherent in this research.

3.1 Dataset - Common Voice

As mentioned in section 2, there are two main corpora available for Frisian: FAME!, a bilingual and code-switched Frisian-Dutch corpus consisting of radio recordings of speech (Yılmaz, Andringa, et al., 2016), and Common Voice, a massive multilingual crowdsourced corpus, freely accessible and where everyone can contribute to it (Ardila et al., 2020). FAME! contains 18.5 hours of speech collected from Omrop Fryslân, the regional public broadcaster of the province of Fryslân. The data has been manually annotated by 2 native Frisian experts and the main purpose of the dataset collected is to be used in code-switching research.

Common Voice, on the other hand, is not restricted to only Frisian since it is a project designed to incorporate as many languages as possible for the development of inclusive speech recognition. Recordings are made by the community, where they are asked to read given prompts. Contributors can also validate the recordings of others which alleviates the need of experts in transcription, especially where none may be available due to the low population of speakers for it. However, recordings can be made in any environment which can introduce a certain level of noise to the recording. Despite that, I will be utilizing this corpus as it contains much more data than FAME!, as well as speech from a wider range of speakers and that focuses exclusively on Frisian. The number of hours of speech can vary, depending on the version of the corpus downloaded. The version used in experiments, 8.0, contains 50 hours of validated speech, whereas the latest release, 13.0, has 67 hours. Common Voice has also been used in prior large-scale cross-lingual ASR work for training and evaluation, including the XLSR models I used (Babu et al., 2021; Conneau et al., 2020), which makes it a suitable choice for my own research.

3.2 Large-Scale Cross-Lingual Models - XLSR-53 & XLS-R

XLSR is a cross-lingual architecture based on wav2vec 2.0 that learns discrete tokens across different languages instead of a monolingual context (Conneau et al., 2020). As it can be seen in figure 1, the feature extractor on the left-hand side learns discrete speech representations in a multilingual context due to the raw speech input originating from several multilingual datasets. In such a manner, wav2vec 2.0 discretizes the latent representations independent of the language which enables it to learn robust features that can be used in an unrestricted linguistic context.

In Conneau et al. (2020), several models are compared: monolingual ones trained on either English or different languages from the Common Voice dataset, XLSR-10 trained on 10 high- and low-resource languages from Common Voice, or XLSR-53, the largest model trained on the most amount of data (56,000 hours) from 53 languages. XLSR-53 has been trained on several datasets,

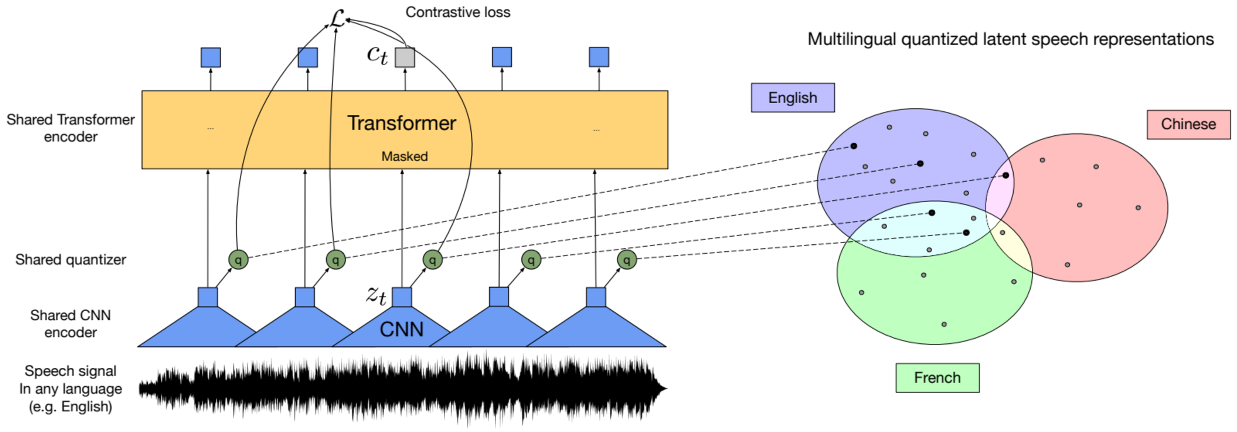

Figure 1: The XLSR approach, based on wav2vec 2.0. Reprinted from Conneau et al. (2020) (p. 2).

including Common Voice, Babel, and Multilingual Librispeech. The baseline model discussed in the previous sections and used in the experiments is a fine-tuned version of the pre-trained XLSR-53 model on Frisian speech from Common Voice (de Vries, 2021).

The same research team worked on an even larger model for speech recognition, named XLS-R (Babu et al., 2021). The researched models are all trained on the same data, which consists of 436,000 hours of speech from five different public datasets (VoxPopuli and VoxLingua107 in addition to the ones used in XLSR-53, as it can also be seen in figure 2) that cover 128 languages. The models differ in terms of the number of parameters: a 0.3B parameters model, the same number of parameters as XLSR-53, a 1B parameters model, and a 2B parameters model. Both the 1B and 2B parameters models perform better than XLSR-53 in all speech recognition tasks, whereas the 0.3B model performs comparably to XLSR-53.

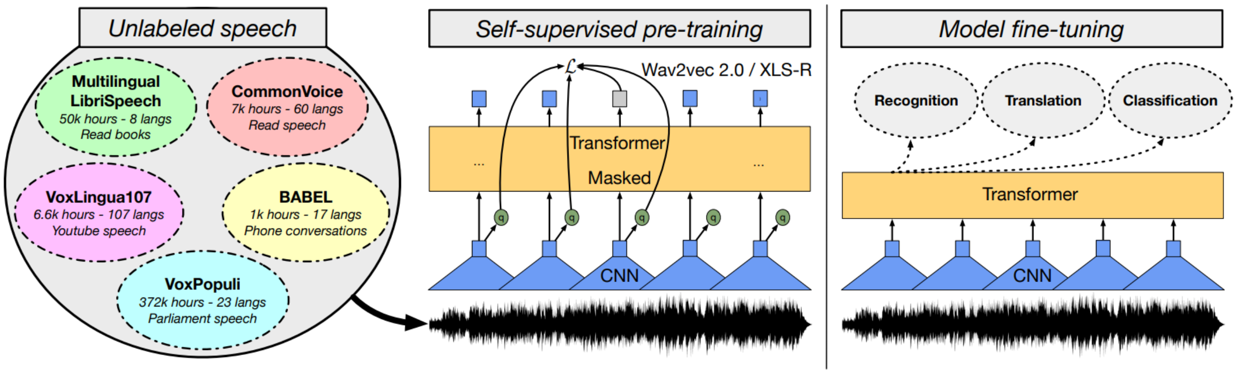

Figure 2: The XLS-R approach, based on wav2vec 2.0 and larger than the previous attempt, XLSR. In addition to speech recognition, this model has been tested on speech translation and language identification tasks as well. Reprinted from Babu et al. (2021) (p. 2).

Aside from speech recognition, the model has been fine-tuned for speech translation and identification as well, as indicated in figure 2. The authors show that the model can be applied to those tasks as well, achieving state-of-the-art results. Since XLS-R leverages more information compared to XLSR and obtains better results in various tasks, I have decided to use it and observe whether it improves over the baseline or not, thus confirming the findings of Babu et al. (2021). Since the 0.3B parameters model performed similar to XLSR-53 and the 2B parameters model would be expensive to fine-tune for each of the experiments, I decided to use the XLS-R 1B parameters pre-trained model and fine-tuned on Common Voice Frisian, similar to de Vries (2021), for most of the experiments. The 0.3B and 2B models have also been used in two experiments to benchmark their performance in relation to XLS-R with 1B parameters.

3.3 Feature Extractor Framework - wav2vec 2.0

The crucial component that connects the XLSR models mentioned above is the feature extractor. The feature extractor is represented by the wav2vec 2.0 framework (Baevski, Zhou, et al., 2020). An illustration of it can be seen in figure 3. Wav2vec 2.0 takes as input raw speech X and feeds it through a multi-layer convolutional feature encoder $f : \mathcal{X} \rightarrow \mathcal{Z}$ to output latent representations $z_1, ..., z_T$ for T time steps, where T is smaller than the length of the input. It effectively reduces the number of samples so that the next block in the architecture can process it in an easier manner. After that, the Transformer $g : \mathcal{Z} \rightarrow \mathcal{C}$ takes the latent representations outputted by the encoder and converts them to context representations $c_1, ..., c_T$ that capture information about the entire sequence rather than being localized representations that contain details only about the current timestep. In addition, there is also a quantization module $\mathcal{Z} \rightarrow \mathcal{Q}$ that discretizes the latent representations and outputs quantized representations, used as targets in the self-supervised task.

The input to the feature encoder is normalized to zero mean and unit variance due to the activation function used by this module, Gaussian Error Linear Unit (GELU). GELU is defined as:

$$GELU(x) = xP(X \leq x) = x\phi(x)$$

where $\phi(x)$ is the cumulative distribution of the standard normal function (Hendrycks & Gimpel, 2020). Since the standard normal function is a part of this activation function, normalization is needed in order for the function to activate as expected. After the normalized input is fed through temporal convolutions, the output is normalized through layer normalization and then activated by the GELU activation function. The stride of the encoder determines the number of timesteps T. The Transformer block uses a convolutional layer which applies a relative positional embedding to the input. Then, the output is activated using GELU and added to the input of the Transformer, followed lastly by layer normalization.

As previously mentioned, the quantization module is responsible for discretizing the output of the encoder. It does so by using product quantization, which consists of choosing entries from multiple codebooks G and concatenating them. Given the G codebooks with V entries each, one entry is chosen from each codebook, then all the selected entries are concatenated into a vector $e_1, ..., e_G$ on which a linear transformation is applied in order to obtain vector $\mathbf{q}$. In order to select the codebook entries in a differentiable way, a Gumbel-softmax function is used (Jang, Gu, & Poole, 2017). It works similarly to an average softmax function, except it is computed over samples drawn from a Gumbel distribution. The probability of choosing entry v in codebook g is:

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

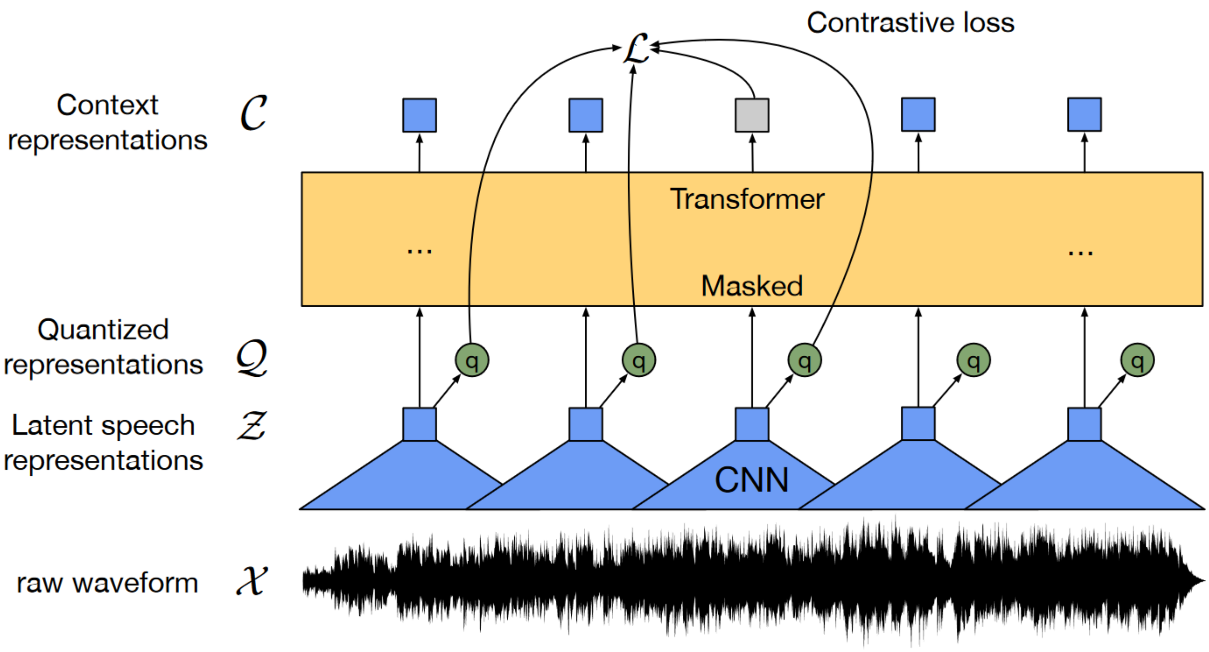

Figure 3: The wav2vec 2.0 framework used in XLSR-53 and XLS-R. Reprinted from Baevski, Zhou, et al. (2020) (p. 2).

where l is the logit to which the feature encoder is mapped, $n = -\log(-\log(u))$, u are samples drawn from a uniform distribution $U(0,1)$, and τ is a non-negative temperature that, if closer to zero, the sample becomes a one-hot encoding and the further the value is from zero, the more uniform the sample and conversely the Gumbel-softmax distribution become. In forward passes, codeword i is chosen by $i = argmax_j(p_{g,j})$. In backward passes, the gradient of the Gumbel-softmax outputs is used.

The context and quantized representations are used during training in a loss function which is defined as:

$$\mathcal{L} = \mathcal{L}_m + \alpha \mathcal{L}_d$$

where $\mathcal{L}_m$ is a contrastive loss, $\mathcal{L}_d$ is a diversity loss, and α is a hyperparameter that balances the contrastive and diversity losses. The goal of the contrastive loss is to maximize the contrast between the true quantized representation and a set of K distractors so that the true representation is the most likely one to be chosen. It is defined as:

$$\mathcal{L}_m = -\log \frac{\exp(sim(c_t, q_t)/\kappa)}{\sum_{\tilde{q} \sim Q_t} \exp(sim(c_t, \tilde{q})/\kappa)}$$

where c_t is the context representation at timestep t, q_t is our true quantized representation, κ is the number of distractors, $\tilde{Q}_t$ includes q_t and the κ distractors, and $sim(a,b) = a^T b/\|a\|\|b\|$ refers to the cosine similarity of two vectors. Essentially, the resulting loss value will always be positive and the goal is to minimize it such that there is a clear contrast between the true quantized representation and the distractors.

The diversity loss, on the contrary, increases the chances of using each entry in a codebook, thus increasing the diversity of the choices so that the model does not overfit on one entry. It is defined as:

$$\mathcal{L}_d = \frac{1}{GV} \sum_{g=1}^{G} \sum_{v=1}^{V} \tilde{p}_{g,v} \log(\tilde{p}_{g,v})$$

where G is the number of codebooks, V is the number of entries in each codebook, and $\tilde{p}_{g,v}$ corresponds to the Gumbel-softmax probability of entry v from codebook g. Since the probability is always a value between 0 and 1, the logarithm of such a number will be negative in value, which makes the loss function negative. The closer to zero the number is, the less diverse the selection of entries is and the higher the loss. If the value is further from zero, then it will minimize the overall loss function.

To fine-tune the framework for speech recognition, a linear projection is added on top which is trained using tokens, and a CTC loss which outputs sequences of tokens. Tokens in this case correspond to characters. Employing the CTC method, we make predictions of tokens for every frame in our audio features. These tokens are selected from a predefined vocabulary. The prediction of each token is influenced by the preceding frame. In the current frame, if the previous token is not a special token, the current token will either be a special token or the same as the previous one. Conversely, if the previous token is a special token, the current token can be any non-special token.

The special tokens consist of the whitespace token, a blank token that helps with decoding words that contain two letters next to each other (such as ”namme”), or an unknown token for characters that are out of the vocabulary. After tokens have been predicted for each frame, the non-space tokens that appear consecutively are combined into a single character, resulting in a character sequence for each group of tokens separated by a space token. The CTC loss function’s role is to minimize the differences between the predicted and the target character sequences so that the model outputs accurate transcriptions.

In order to make the model more robust, a modified version of SpecAugment is implemented to add artificial degradation to the audio in certain frequency and time ranges (Park et al., 2019). To optimize the model, the authors use a modified version of the Adam optimizer (Kingma & Ba, 2017) that decouples the weight decay from the optimization steps, named AdamW (Loshchilov & Hutter, 2019). AdamW manages to generalize better than Adam due to the decoupling, which is a crucial element for any deep learning model.

3.4 Evaluation - Word Error Rate

The Common Voice Frisian dataset is split into three: train and development/validation subsets for training and optimizing the XLS-R model, and a test subset used for evaluating the performance of the XLSR-53 baseline and the final XLS-R model.

I analyze the results of the test set using Word Error Rate, or WER, as the performance metric. I also compare the models of each experiment using absolute and relative WER differences, metrics that are commonly used in the field of automatic speech recognition. More details about the formulas used for the metrics can be found in Appendix A.

3.5 Ethical considerations

While the research aimed to develop an ASR system that benefits the Province of Fryslân, there is still a possibility that the technology may have unforeseen consequences. In order to mitigate the risks, the research team will communicate the study's results and implications in an accessible and transparent manner.

I have not collected any sort of data from human participants. Instead, I used a previously-recorded dataset, which is Mozilla's Common Voice project (Ardila et al., 2020). It is a multilingual, open-, and crowdsourced corpus that is constantly updated, with support for over 100 languages. The participants in the Common Voice project are informed about their data being collected and they do so voluntarily. The recordings are also validated by the community. The corpus is licensed under CC0[1], therefore any distribution, adaptation, or otherwise may be made freely, without having to credit or mention in any way.

Most of the dataset contains data from speakers whose characteristics are unknown. Therefore, a certain degree of bias might be present in the models trained and evaluated which I mitigate by disclosing clearly and precisely that bias is present.

Objective metrics were used for evaluation which are relevant to the field. Therefore, subjective evaluation methods involving human participants have not been used and are mostly not significant to use in the field of speech recognition. Therefore, there are no concerns regarding the ethics of involving human participants or any other issues that do not align with the ethics of the faculty.

As when it comes to the replicability of the research, the code is available via GitHub[2] and the fine-tuned models are available via Hugging Face Hub[3]. URLs to each model can be found in table 4 of section 5. All steps and details on how to reproduce the experiments to be described in the thesis can be found under section 4. The dataset is publicly available to download and use. The outcomes should be more or less similar, but they may not be exactly the same due to certain elements that introduce randomness in the trained models. The hardware used may also impact the performance of the models since the experiments have been conducted on the University of Groningen's high-performance cluster, Hábrók.

This concludes the methodology section which explains at a high-level the methods employed during this research. In the next section, the experimental setup will be presented which will include more low-level details about the dataset used and the parameters of the models.

[1] Information about the CC0 license: `https://creativecommons.org/share-your-work/public-domain/cc0/`

[2] `https://github.com/greenw0lf/MSc-VT-Thesis/`

[3] `https://huggingface.co/greenw0lf`

4 Experimental Setup

In order for the research to be fully reproducible, the experimental setup must be explained in detail. The first experiment is built in order to answer the main research question, as well as the subsequent four subquestions, whereas the other experiments will mainly answer the third and fourth subquestions pertaining to the relation between the amount of training data and the performance of XLS-R, as well as the relation between the resources required to conduct the experiments and the performance obtained. The experiments are arranged such that there is no overlap between the train, development, and test sets and the hardware environment is identical. Hyperparameters have been tuned in each experiment in order to ensure the most optimal models.

I will start by examining the different subsets of data used from Common Voice in subsection 4.1. Then, the data preprocessing step is explained in subsection 4.2. Next, the XLS-R hyperparameters that have been tuned for my experiments are presented and analyzed in subsection 4.3. Lastly, subsection 4.4 covers the hardware setup as well as the duration of the training and evaluation of the experiments.

4.1 Data Splitting of Subsets

To answer the third subquestion of the research question, I experiment with different amounts of training data. The development and test splits remain identical for all experiments. For the first five experiments, the 1B parameters version of the pre-trained XLS-R model is used. The first experiment uses the same 5-hour training split predefined in Common Voice 8.0 that the baseline model by de Vries (2021) uses, then for experiment 2 I use all of the validated data available except for the development and test splits, which rounds up to 41 hours of speech. Experiments 3, 4, and 5 follow a similar structure to the research done by Babu et al. (2021); Baevski, Zhou, et al. (2020); Conneau et al. (2020), which is training using approximately 10 hours, 1 hour, and 10 minutes of speech respectively. The last three experiments are employed in a comparative analysis to assess the quantity of data required for achieving high-performance systems and the results of these experiments could be generalized for different situations of data available in a low-resource language to understand the quantity of data required for good performance. The fourth and final subquestion is answered by experiments 6 and 7, where I employ the same training set as experiment 2 on XLS-R with 0.3B and 2B parameters respectively.

Recordings for experiments 3-5 have been randomly selected by myself from experiment 2's train split such that they add up to the number of hours required. No recording has been split into smaller segments and the random selection was done regardless of the speaker, therefore some bias might be present for a specific group of speakers for each of the last three experiments. All of the tables containing information about the different splits according to the sex and age of the speaker can be found in Appendix B.

4.1.1 Development & Test Subsets

As previously mentioned, I use the same development and test subsets of Common Voice 8.0. These subsets are already defined in the corpus and each one of them corresponds to approximately 9% of the total validated data, which is around 4.5 hours. A more detailed analysis of the development set can be found in table 5, whereas for the test set the study can be found in table 6.

Most of the data comes from speakers whose main characteristics, age and sex, are unknown which unfortunately means that, due to the lack of metadata about speakers overall, it is not feasible to conduct more thorough research into the bias of the subsets used for evaluating either during training or testing phases. This trend is observed throughout all of the data in each experiment.

4.1.2 Experiment 1: 5 Hours of Training Data

For the first experiment, the same training set as the baseline by de Vries (2021) is used, which contains approximately five hours of data. This training set corresponds to the train split predefined in the corpus. This was done in order to have a straightforward comparison to the baseline and showcase that, in the same conditions with only the model itself being different, the performance is improved. More details about the training data can be found in table 7.

4.1.3 Experiment 2: 41 Hours of Training Data

For the second experiment, I use a training set of 41 hours which corresponds to all of the validated data available in Common Voice 8.0 excluding the test and development sets. This is done in order to set a state-of-the-art performance by employing all of the validated data available, as well as to compare the difference in performance when using 8x more data to show how the scarcity of data can affect the performance of the models. A detailed breakdown of the data can be seen in table 8.

4.1.4 Experiment 3: 10 Hours of Training Data

In the third experiment, data has been randomly selected from the training subset of experiment 2 such that a subset containing 10 hours of speech is created. This experiment is directly linked with the third subquestion, whereas the previous two experiments answer all of the subquestions presented in the introduction. More information about metadata of the speakers can be found in table 9.

4.1.5 Experiment 4: 1 Hour of Training Data

The fourth experiment investigates a lower-resource context compared to the previous paragraph. In this setup, 1 hour of speech has been randomly selected from the same train subset as experiment 2. As with experiment 3, experiment 4 seeks to clarify the impact of ten times less training data on the performance of the model. A table of statistics about this subset can be seen in table 10.

4.1.6 Experiment 5: 10 Minutes of Training Data

Experiment 5 covers an extremely low-resource context where a language only has 10 minutes of data available for training. The subset has been randomly extracted similarly to experiments 3 and 4. A detailed breakdown of the data can be observed in table 11.

4.1.7 Experiment 6: XLS-R with 0.3B Parameters

The setup of experiment 6 is identical to experiment 2 in terms of training data used. Therefore, the model was fine-tuned using 41 hours of training data. As for the number of parameters, this experiment evaluates XLS-R with the same number of parameters as XLSR-53, 0.3 billion parameters, with the emphasis being placed on the score and its relation to how much time it took to train or evaluate the model.

4.1.8 Experiment 7: XLS-R with 2B Parameters

Similar to experiment 6, experiment 7 is fine-tuned using the same amount of training data, which is 41 hours, but on XLS-R with 2B parameters. This is the model with the highest number of parameters, therefore it is expected to perform the best, but training resources and time required doubles compared to XLS-R with 1B parameters.

4.2 Data Preprocessing

All subsets used in the experiments have several metadata columns available. Those columns contain information about the ID of the recording, the path to the audio file, the spoken sentence, as well as information about the speaker like age, gender, and accent. Since the main task is to recognize Frisian speech regardless of the speaker, I remove all metadata except for the audio path and the sentence spoken. After that, I preprocess the remaining columns.

The text labels associated with our speech samples need to be simplified since we align the speech features obtained from wav2vec 2.0 with characters using a CTC loss function. If we were to use a language model, then we could keep the transcriptions as they are. Thus, the first step in the text normalization phase is to remove special characters that are not linked to a specific sound, such as punctuation („.?!;:). Then, we make all characters lowercase since lowercase and uppercase represent the same acoustic features when decoding speech. The resulting preprocessed sentences will contain only the characters from the Frisian alphabet and whitespaces to separate the words. Some examples of normalized transcriptions and a list of the Frisian orthography extracted from the transcripts are provided in Appendix C.

As for the audio, the only preprocessing that has been applied is resampling from the original sample rate of 48000 Hz to 16000 Hz because wav2vec was trained using audio of the latter frequency.

4.3 XLS-R Hyperparameters

When it comes to all experiments, most of the hyperparameters used during fine-tuning of the large-scale XLS-R models are shared. Some hyperparameters correspond to the default values of the framework used for training and evaluating, which is Hugging Face's Transformers[4]. I will be majorly covering the hyperparameters I modified and experimented with.

The hyperparameter that has the most impact on the performance of the model is the learning rate of the AdamW optimizer. I have evaluated several values for the learning rate in the range of $[2e-5, 3e-4]$, as mentioned in Babu et al. (2021), and selected the best ones according to the validation set scores, for each experiment individually. Unfortunately, I could not retrieve the values I experimented with other than the ones that corresponded to the best validation WER for each experiment, so only the hyperparameters that offered the best results are mentioned. The concrete

[4] https://huggingface.co/docs/transformers/index

values can be found in table 3. The other parameters related to the optimizer have been set the same for all experiments. The values are $\beta_1 = 0.9$ and $\beta_2 = 0.98$, the same values as in the GitHub page of the XLS-R model[5]. I use a linear schedule with a warmup for the learning rate. In my experiments, I utilize a warmup ratio of 10% of the total training steps, which corresponds to linearly increasing the learning rate until 10% of the training steps have been executed, then linearly decaying the remaining 90% of the total steps. Aside from the optimizer, the SpecAugment probabilities for the feature and time masking are both set to 0.3 or 30%, which corresponds to the lower end of the 30%-75% range suggested in Babu et al. (2021). To further optimize training, fp16 16-bit (mixed) precision training is utilized instead of 32-bit training, as well as an evaluation batch size of 8 which corresponds to the default of the Transformers framework.

In addition to the optimizer, I have also fine-tuned different hyperparameters that impact the training time for each experiment. Specifically, I tuned the number of epochs, batch size, as well as how often the model should checkpoint and evaluate, expressed in the number of steps. A summary of the values chosen for each subset can be seen in table 3. The aim was to strike a balance between the time and resources required for fine-tuning the models and achieving optimal performance, resulting in the optimization of these hyperparameters. However, given the limitations imposed by time and resources during the thesis project, the selected hyperparameters may not be the most optimal choice, and further fine-tuning is advised.

Overall, the smaller the number of steps, the more often evaluation and checkpointing are done, and the number of epochs is directly proportional to the training time. A higher batch size reduces computational time. However, what has been observed is that the amount of training data used determines the overall expected performance of the model, regardless of the hyperparameters tuned. Therefore, tuning hyperparameters will only provide minor improvements to the model and, ultimately, the amount of data still has the largest impact on the performance, regardless of whether the hyperparameters used are optimal or not.

Table 3: The hyperparameter values chosen for each experiment.

Experiment #	**Learning rate**	**# of epochs**	**# of steps**	**Batch size**
Experiment 1	5e-5	50	200	32
Experiment 2	3e-5	40	400	36
Experiment 3	5e-5	50	300	32
Experiment 4	6e-5	80	100	32
Experiment 5	7e-5	80	50	16
Experiment 6	5e-5	40	400	32
Experiment 7	3e-5	20	400	16

[5] https://github.com/facebookresearch/fairseq/blob/main/examples/wav2vec/xlsr/config/finetune.yaml

4.4 Hardware and Training Time

The experiments were conducted on the Hábrók high-performance computing cluster of the University of Groningen. The GPU used is an Nvidia A100 GPU accelerator card with 40 GB of VRAM available. Testing the models of the first five experiments takes around 3.5 hours. For experiment 6, testing took 1.5 hours to complete. For experiment 7, testing took approximately 6.5 hours. The training time on average for each experiment is the following:

- **Experiment 1 (5 hours of training data)**: 4 hours
- **Experiment 2 (41 hours of training data)**: 22 hours
- **Experiment 3 (10 hours of training data)**: 7.5 hours
- **Experiment 4 (1 hour of training data)**: 1.5 hours
- **Experiment 5 (10 minutes of training data)**: 45 minutes
- **Experiment 6 (XLS-R with 0.3B parameters)**: 16 hours
- **Experiment 7 (XLS-R with 2B parameters)**: 1 day

5 Results

The results of each experiment compared to the baseline on the validation and test sets of the Frisian Common Voice 8.0 subset can be found in table 4.

Table 4: Results of the baseline compared to each experiment on the validation and test sets of Frisian Common Voice 8.0. For the baseline, the underlined WER is the one used as reference. Clicking on a cell in the ”Experiment” column will lead to its Hugging Face model page.

Model	Experiment	Validation WER	Test WER
XLSR-53	Baseline model (de Vries, 2021) (CV 8 train split, 5 hours)	-	reported: 16.25% actual: <u>15.19%</u>
XLS-R (1B)	Experiment 1 (CV 8 train split, 5 hours)	14.29%	14.13%
	Experiment 2 (41 hours)	4.21%	**4.11%**
	Experiment 3 (10 hours)	9.61%	8.83%
	Experiment 4 (1 hour)	23.73%	25.4%
	Experiment 5 (10 minutes)	52.62%	62.25%
XLS-R (0.3B)	Experiment 6 (41 hours)	7.24%	7.1%
XLS-R (2B)	Experiment 7 (41 hours)	**4.05%**	4.22%

Examining table 4, there is a negative correlation between the word error rate and the amount of data used for training. The best-performing model is the XLS-R model trained on the most amount of Frisian data, which is 41 hours. The absolute WER improvement from the baseline is 11.08%, and the relative WER improvement is 73%. Conversely, the worst-performing model is the one fine-tuned on only 10 minutes of speech which scores a WER of 62.25% on the test set. Compared to the baseline, it is an absolute WER difference of 47.06% and a relative WER increase of 310%.

In addition, the number of parameters also impacts the performance of the model. By looking at experiments 2 and 6, there is an improvement of XLS-R with 1B parameters compared to 0.3B in terms of WER. The absolute improvement of experiment 2 over experiment 6 when using the same hyperparameters is 2.99%, and the relative improvement is 42%. However, for experiment 7, a decrease in performance is observed. It outperforms the model with 0.3B parameters, but it fails to outperform the 1B model, scoring an absolute WER difference of 0.11% and a relative decrease in performance of 3%. What can also be observed is that for experiments 1-3 and 6 the validation WER is larger than the test WER, whereas for experiments 4, 5, and 7, the contrary is true.

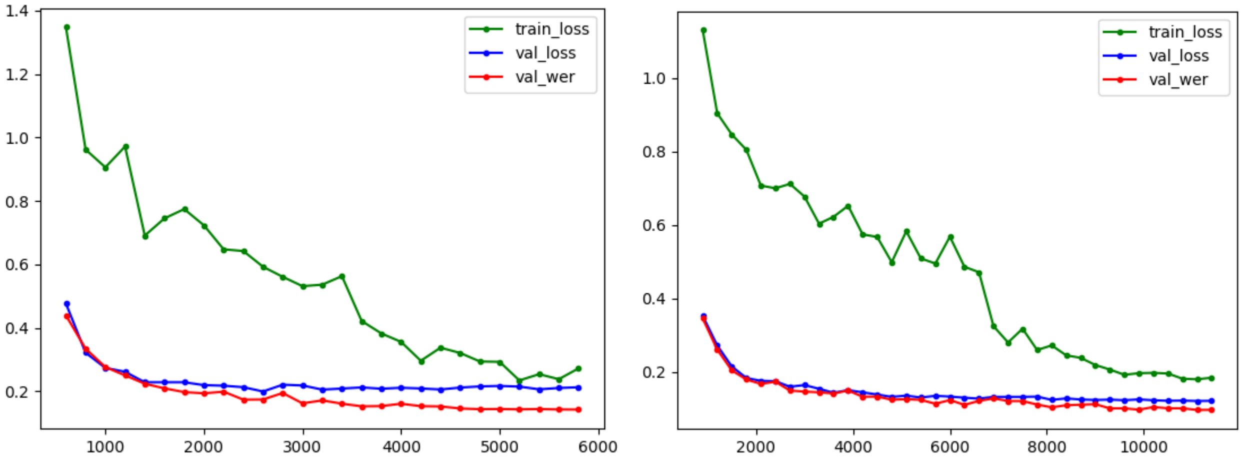

Figure 4: Train, validation losses and validation WER for experiment 1 (5 hours of speech, left) and experiment 3 (10 hours of speech, right).

Observations can also be made for figure 4. The models, which underwent fine-tuning using 5 and 10 hours of data respectively, exhibit comparable patterns in terms of train and validation losses. The train loss initially begins at notably higher values and then descends rapidly over the course of the steps. Meanwhile, the validation loss commences with lower initial values compared to the initial train loss values, gradually converging towards a range of values that are close to each other by the end. The validation WER and loss have similar tendencies when it comes to plot trajectories. The gap in train and validation losses is also reduced towards the end with the train loss still being higher than the validation loss in both plots. It is worth mentioning that the starting points of the plots are at step 600 for the model on the left and at step 900 for the model on the right in order to have better visibility of the tendencies of the losses.

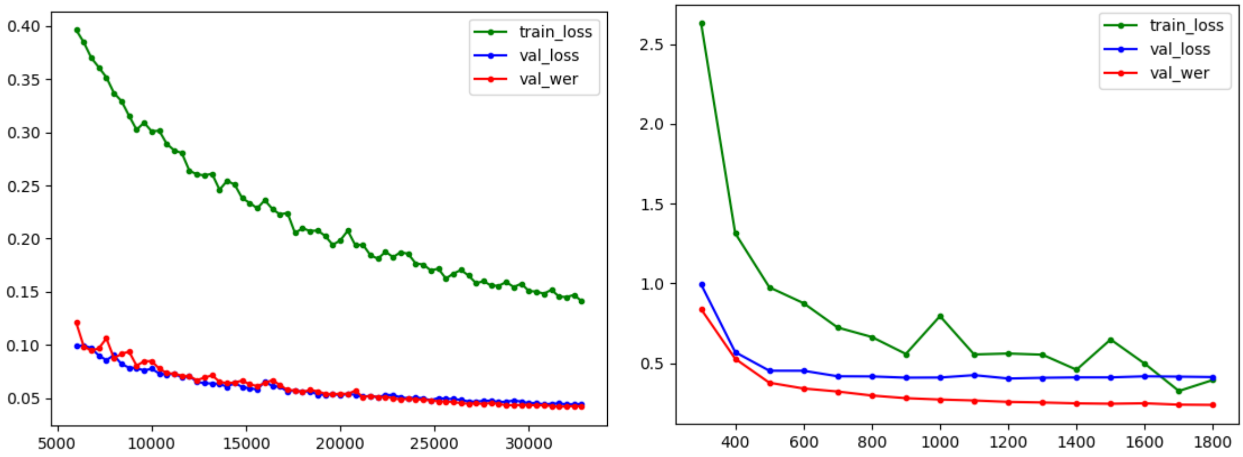

Figure 5: Train, validation losses and validation WER for experiment 2 (41 hours of speech, left) and experiment 4 (1 hour of speech, right).

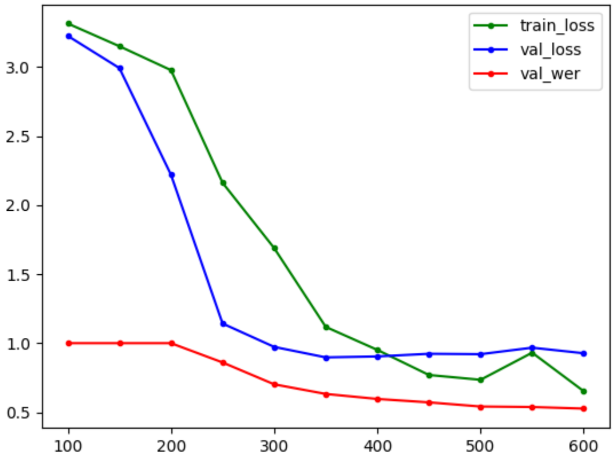

Figure 6: Train, validation losses and validation WER for experiment 5 (10 minutes of speech).

For figures 5 and 6, similar trajectories can be found in the models trained on 41 hours of data, 1 hour of data, and 10 minutes of data. There are discrepancies, however, in the differences between training and validation losses. Specifically, for the model trained on 41 hours of Frisian speech, the gap is much larger between the two losses, whereas for the models trained on 1 hour and 10 minutes, the training loss has a smaller value than the validation loss towards the end. Looking at figure 6 in particular, the training loss becomes smaller than the validation loss around midway of training the experiment 5 model whereas the training loss of experiment 4 in figure 5 is less than the validation loss towards the last few steps. The plots of the experiments discussed in this paragraph start at step 6000 for experiment 2, step 200 for experiment 4, and step 100 for experiment 5, for better readability.

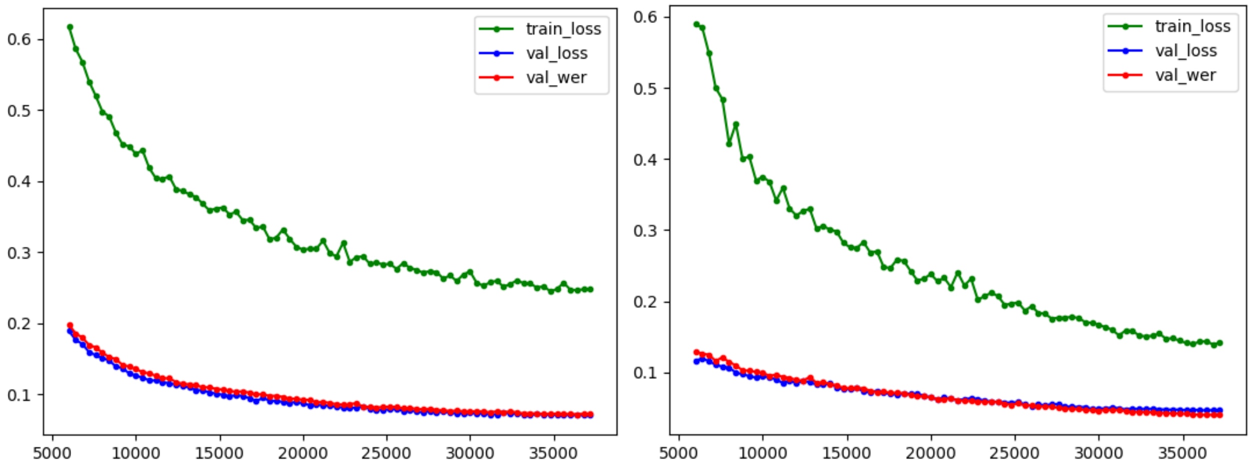

Figure 7: Train, validation losses and validation WER for experiment 6 (XLS-R with 0.3B parameters, left) and experiment 7 (XLS-R with 2B parameters, right).

In figure 7, the loss and WER values during training can be observed for experiments 6 and 7. When comparing to experiment 2, the plots have comparable trajectories. However, the train losses have different starting points in each plot, with experiment 2 starting at 0.4, whereas the last two experiments have higher initial train losses. What can be noticed is that the train loss of experiment 7 converges towards the end to a range similar to experiment 2 whereas experiment 6 converges to a loss value larger than the final loss values of experiments 2 and 7. The plots of the experiments discussed in this paragraph start at step 6000, the same as experiment 2, for readability reasons.

This section covered the results obtained in the different experiments compared to the baseline, as well as some plots regarding the train and validation losses and validation WER, with some preliminary observations that can be made by analyzing these results. In the next chapter, the results will be thoroughly discussed and scrutinized and it will be shown how they answer the research question and validate the hypothesis.

6 Discussion

Upon analyzing the results presented in table 4 from the previous section, it is evident that XLS-R confirms the hypothesis outlined in Subsection 1.1. By utilizing an equivalent amount of training data as the baseline (experiment 1), XLS-R achieves a remarkable WER of under 20% and surpasses the current state-of-the-art XLSR-53 model on the Frisian subset of Common Voice 8.0, which answers the main research question.

6.1 Validation of the First Hypothesis

The baseline has been evaluated in order to validate the reported score by de Vries (2021). The result obtained after the evaluation is better than the reported value, achieving a WER of 15.19% versus the reported 16.25%, thus answering the first subquestion of the research question. Therefore, when conducting comparisons, the word error rate I obtained will be used instead of the reported error rate.

In experiment 1, XLS-R obtains a WER of 14.13%, showcasing an enhancement compared to the baseline established by de Vries (2021). This improvement can be quantified as an absolute WER difference of 1.06% and a relative WER improvement of 7%. Although the relative WER improvement is not in a similar range to the scores obtained in Babu et al. (2021) when comparing XLSR-53 with XLS-R, it is still significant as it shows that a larger number of parameters and hours used during pre-training can improve the performance of an ASR model in low-resource contexts and languages. This experiment confirms the first hypothesis that XLS-R with 1 billion parameters fine-tuned on Frisian performs better than the XLSR-53 baseline and achieves a WER below 20%. This experiment also addresses the main research question and the first two subquestions of the research question.

6.2 Validation of the Second Hypothesis

The second hypothesis about the relation between the quantity of data and the performance of the model which answers the third research subquestion is validated by the results of the first five experiments. The more hours of speech are used during training, the better the models perform in terms of WER. Experiment 2 uses 41 hours of Frisian during training and achieves a WER of 4.11% which is better than the performance of languages with a higher number of hours used during training in Babu et al. (2021), such as those from the VoxPopuli dataset. A direct comparison can not be made with Common Voice as the authors of the paper utilize phoneme error rate as a metric. Nevertheless, the performance of the XLS-R fine-tuned in experiment 2 is noteworthy and sets a new state-of-art in Frisian speech recognition. It could be investigated whether this performance is as remarkable on the FAME! corpus discussed in subsection 2.1.

The experiments conducted show a consistent improvement in word error rates as the number of hours used for training increases, ranging from 37.5% to 59.2%. These results provide strong evidence to support the second hypothesis that more data leads to better performance and increased robustness in fine-tuned models. Additionally, the more considerable amount of training data helps mitigate overfitting issues by incorporating the variability of available speech data.

The relation between the performance of the model and the amount of training data used can also be observed in the vocabulary size. For experiments 1 and 2, the vocabulary size is 41, whereas for experiments 3, 4, and 5 the sizes are 40, 39, and 38 respectively. The characters that are missing from the vocabulary for experiments 3-5 can be found in Appendix C. Therefore, the word error rate is greatly increased in cases where the word is transcribed correctly with the exception of one character that happens to not be in the vocabulary of the model. In those cases, the word altogether is considered wrong, and, therefore, the word error rate increases.

However, it is important to note that the absolute differences in WER improvement vary significantly across different training durations. For instance, when comparing one hour of training data to 10 minutes, there is a substantial absolute WER improvement of 36.85%. On the other hand, the improvement from fine-tuning with 10 hours of speech to 41 hours is relatively modest, with only a 4.72% absolute WER reduction.

The impact of data sparsity on model performance cannot be understated. When using a mere 10 minutes of Frisian data, the resulting WER of 62.25% yields transcriptions that are often unintelligible. Conversely, the difference in WER between fine-tuning with 10 hours and 41 hours might not be as perceptible to users of the ASR system, as both models achieve an acceptable word error rate below 10%.

All of the experiments discussed above showcase the importance of the number of hours of speech available for fine-tuning a model. The results obtained and observations formulated are not only applicable to Frisian but can also be generalized for other low-resource languages, similar to what Babu et al. (2021) and Conneau et al. (2020) have also observed.

6.3 Validation of the Third Hypothesis

The last two experiments, along with experiment 2, are compared not only in performance but also in their time and hardware requirements. The model with 0.3B parameters from experiment 6 requires the least amount of time and hardware, as expected due to its smaller parameter count. It takes approximately 16 hours to train and 1.5 hours to evaluate, achieving a WER of 7.1% on the test set. This shows a considerable difference compared to experiment 2's 1B parameter model, which scores a WER of 4.11%. Experiment 2's relative WER improvement over experiment 6 is 42%.

On the other hand, XLS-R with 2B parameters demands the most time and hardware resources for training and testing. Training the model takes around a day, and evaluation requires 6.5 hours, totaling 30.5 hours to fine-tune it with 41 hours of Frisian speech data. This is significantly higher than the resources needed for the model in experiment 6. Despite the larger parameter count, XLS-R with 2B parameters does not perform better than experiment 2. It scores a WER of 4.22%, indicating a 0.11% decrease compared to XLS-R with 1B parameters, resulting in a relative decrease of 3%. The model of experiment 7 seems to overfit since the validation WER is smaller than the test WER, even though the train loss is higher than the validation loss. To address memory overflow issues, the training batch size and number of epochs had to be significantly decreased during fine-tuning.

Overall, experiment 2 strikes a balance between performance and the time and hardware required. It takes 25.5 hours in total and outperforms both the 0.3B and 2B models, which take 17.5 hours and 30.5 hours, respectively. This confirms the validity of the third hypothesis and answers the fourth subquestion about the model that balances resources and performance the best. However, if fewer resources are available, it may be recommended to use the 0.3B model as it can still perform decently well.

6.4 Limitations

By inspecting the validation and test word error rates of experiments 1-3 and 6, it can be observed that validation WER is larger than test WER. This indicates that the models underfit, which is further confirmed by the validation loss maintaining lower values than the training loss until the end of training. Conversely, experiments 4, 5, and 7 show signs of overfitting as the test WER is larger than the validation WER and the validation loss is larger than the training loss, indicating that the models do not contain enough training data to be able to generalize and perform well. Therefore, more hyperparameter tuning can be executed since the models can potentially improve performance. Some examples include the learning rate or the number of epochs, which can be increased for experiments 1-3 in order to train the model for longer and reduce underfitting, or reduced for experiments 4 and 5 to reduce overfitting.

Another limitation is the hardware used. Training time could have been severely reduced if more hardware resources could have been employed, such as multiple GPUs with more video RAM. This is especially applicable to experiment 7 since the number of epochs and training batch size had to be downgraded in order to be able to train the model. Removing this limitation would have also allowed for more hyperparameter tuning and consequently finding the optimal parameters to use for the models of each experiment.

Finally, no extensive bias analysis can be conducted on the datasets used for training and evaluating due to the lack of information on the speakers for the majority of the recordings. In a more realistic setting, the model might prove to perform inaccurately for a certain group of Frisian speakers. Therefore, if there would be a Frisian dataset comparable in size to Common Voice and contains information about the speakers for most, if not all the recordings, then it would be relevant to investigate that dataset and analyze where bias occurs.

Related to bias, the recordings of the training sets for experiments 3-5 have been randomly selected. It may be more relevant to instead select the recordings for experiments 3-5 in such a way that the vocabulary sizes are the same across all experiments and to balance the duration for each group of speakers present in the corpus.

In summary, the research question has been addressed and my initial hypothesis has been validated. As the research objectives have been met and the study's contributions have been established, the subsequent section will serve as the concluding chapter, encapsulating the key findings and their implications.

7 Conclusion

A brief summary of the contributions made with this research will be presented alongside future plans and possibilities, ending with a subsection on the impact and relevance of my thesis in the field of low-resource speech recognition and in the Frisian-speaking community.

7.1 Summary of the Main Contributions

To sum up, this thesis addresses the issue of automatic speech recognition systems for low-resource languages by developing models capable of recognizing Frisian. This is achieved by fine-tuning XLS-R (Babu et al., 2021), which is a large-scale cross-lingual model using wav2vec 2.0 (Baevski, Zhou, et al., 2020) as a feature extractor that has been pre-trained on 436,000 hours of data from 128 different languages. In addition, this research analyzes the impact of the number of hours of speech used during fine-tuning to understand better how much data would be needed to develop a highly-performant ASR system.

The XLS-R pre-trained version I used is the 1 billion parameters one. The experiments conducted during this research are compared to a baseline (de Vries, 2021) fine-tuned on XLSR-53, a model similar to XLS-R based on wav2vec 2.0 but pre-trained using only 56,000 hours of speech from 53 languages and using 0.3 billion parameters (Conneau et al., 2020).

Analyzing the results, the model fine-tuned on the same data as the baseline performs better, achieving a WER of 14.13%. This is a significant advancement over the actual baseline result of 15.19%. The relative WER improvement observed is 7% and the absolute WER improvement is 1.06%. This validates my hypothesis that a model fine-tuned on XLS-R can achieve a WER below 20% and, at the same time, improve over the state-of-the-art in Frisian ASR. Adding 8x more data, the relative gain is heavily improved. The XLS-R model fine-tuned on 41 hours of Frisian speech achieves a WER of 4.11% on the test set, improving over the baseline by 73% relative WER and 11.08% absolute WER.

Aside from using the same data as the baseline and 41 hours of training data, I have experimented with various numbers of hours, specifically, I fine-tuned XLS-R with 10 hours, 1 hour, and 10 minutes of Frisian speech randomly selected from Common Voice. The results obtained validate my second hypothesis which relates to the amount of data used during fine-tuning and performance of the model. The performance significantly drops the fewer hours are used during fine-tuning. The worst-performing model, fine-tuned on 10 minutes of Frisian speech, scores a WER of 62.25% which would be noticeable to the users of such a system and would negatively impact the experience.

Lastly, the comparison of the last two experiments with experiment 2 involves evaluating their performance and time and hardware requirements. The 0.3B parameter model from experiment 6 proves to be the most efficient, with the least time and hardware resources required for training and evaluation. However, it achieves a WER of 7.1% on the test set, showing a significant difference from the 1B parameter model in experiment 2, which scores a WER of 4.11%. The XLS-R model with 2B parameters in experiment 7 demands the most time and resources, performing slightly worse than the 1B parameter model with a WER of 4.22%. Experiment 2, therefore, achieves a balance between performance and resources, outperforming both the 0.3B and 2B models.

7.2 Future Work

One of the ideas for future work involves using data from the latest iteration of Common Voice, 13.0, and evaluating the performance in relation to the quantity of data used throughout the experiments in this paper. This could lead to an improvement over the XLS-R model fine-tuned with 41 hours of data given the validated hypothesis about the relation of data and performance. Aside from Common Voice, FAME! (Yılmaz, Andringa, et al., 2016) has been briefly discussed in subsection 2.1 as another Frisian speech database that has been widely used in different research covered in subsection 2.11. It could be investigated whether FAME! can be combined with Common Voice in order to develop a more extensive and robust ASR model.

Other metrics may also be considered for evaluation, such as character error rate (CER) or phoneme error rate (PER) which measure errors at a character or phoneme level respectively. This would offer an introspective into the more subtle differences between words and sources of confusion for the trained models. CER can also capture granularity since it evaluates errors on a smaller scale compared to WER which invalidates an entire word that may have a few characters that are different. PER can provide a language-independent evaluation by looking at the phonemes recognized instead of characters or words and thus evaluating the ability of the ASR system to link specific audio characteristics to phonemes. Another improvement that could be made is to rescore the models using a language model in order to generate improved transcriptions that could involve names, punctuation, and capitalized letters.

More detailed benchmarking should also be conducted for experiments 2, 6, and 7 in order to create a more detailed comparison between the models in terms of resources and performance. For example, an average time per utterance can be calculated when evaluating the models, or using a smaller set of carefully selected sentences may amplify the model's actual performance given certain contexts. In this way, researchers would be more informed about the choice of the number of parameters in a large-scale model given the same amount of training data available.

Furthermore, in addition to XLS-R and XLSR-53, several other large-scale cross-lingual models have emerged in the field of automatic speech recognition, as reviewed in subsection 2.9. Models such as Google USM (Y. Zhang et al., 2023) and mSLAM (Bapna et al., 2022) show promise as potential candidates for fine-tuning and evaluation in low-resource scenarios. While these models exhibit certain advantages, such as their architecture and training methodologies, they lack open-source pre-trained models or accessible source code. This limitation hinders their widespread adoption and limits their usability in research and practical applications.

On the other hand, Whisper, developed by OpenAI, offers an opportunity for experimentation and comparison with XLS-R (Radford et al., 2022). Whisper provides pre-trained models and comprehensive tutorials, making it an accessible resource for researchers. This facilitates the exploration of fine-tuning techniques and the evaluation of performance on low-resource languages like Frisian.

In future work, it would be valuable to conduct a detailed comparison between XLS-R and Whisper, assessing their respective strengths and weaknesses in the context of Frisian speech recognition. By analyzing factors such as recognition accuracy, computational efficiency, and model adaptability, researchers can gain insights into the optimal model choice for low-resource ASR tasks. Additionally, investigating the potential of Google USM and mSLAM once open-source models become available would contribute to a more comprehensive evaluation of state-of-the-art cross-lingual models.

Expanding the comparative analysis to include these models not only enriches the research landscape but also provides a broader understanding of the capabilities and limitations of different approaches in advancing low-resource speech recognition.

7.3 Impact & Relevance

My research has a significant impact on the Frisian community. Firstly, I achieved a groundbreaking advancement in Frisian speech recognition, setting a new state-of-the-art benchmark. This accomplishment demonstrates the effectiveness and reliability of the developed system in accurately transcribing Frisian speech.

Moreover, my work validates the recent advancements in cross-lingual models for low-resource automatic speech recognition (ASR) tasks. By exploring the fine-tuning of larger models using diverse datasets and fine-tuning them specifically for low-resource tasks, I have demonstrated that this approach leads to improved performance. This validation highlights the potential of leveraging these state-of-the-art transfer learning techniques to enhance ASR capabilities for languages with limited resources, such as Frisian.

Furthermore, the potential applications for my research are diverse and far-reaching. One such application involves integrating the developed model into a virtual assistant, which could prove invaluable in various settings. For instance, it could be employed at museums and information booths to provide multilingual assistance to visitors, enhancing their overall experience. Additionally, the virtual assistant could be utilized in care homes to support the elderly, offering them language-based guidance and companionship.

Another significant application of the research lies in assisting language learners interested in reading, speaking, and understanding Frisian. By leveraging the developed model, learners would have access to a powerful tool that facilitates their language acquisition journey. This, in turn, contributes to the preservation of the unique language and culture of the province of Fryslân. Encouraging individuals to communicate in Frisian in their daily lives rather than relying solely on Dutch helps combat the potential extinction of Frisian and fosters a stronger connection to its rich heritage.

References

Ardila, R., Branson, M., Davis, K., Kohler, M., Meyer, J., Henretty, M., … Weber, G. (2020, May). Common Voice: A Massively-Multilingual Speech Corpus. In *Proceedings of the twelfth language resources and evaluation conference* (pp. 4218–4222). Marseille, France: European Language Resources Association.

Babu, A., Wang, C., Tjandra, A., Lakhotia, K., Xu, Q., Goyal, N., … others (2021). XLS-R: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*.

Baevski, A., Schneider, S., & Auli, M. (2020). *vq-wav2vec: Self-supervised learning of discrete speech representations.*

Baevski, A., Zhou, H., Mohamed, A., & Auli, M. (2020). *wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations.* arXiv. Retrieved from `https://arxiv.org/abs/2006.11477` doi: 10.48550/ARXIV.2006.11477

Bapna, A., an Chung, Y., Wu, N., Gulati, A., Jia, Y., Clark, J. H., … Zhang, Y. (2021). *Slam: A unified encoder for speech and language modeling via speech-text joint pre-training.*

Bapna, A., Cherry, C., Zhang, Y., Jia, Y., Johnson, M., Cheng, Y., … Conneau, A. (2022). *mslam: Massively multilingual joint pre-training for speech and text.*

Conneau, A., Baevski, A., Collobert, R., Mohamed, A., & Auli, M. (2020). *Unsupervised Cross-lingual Representation Learning for Speech Recognition.* arXiv. Retrieved from `https://arxiv.org/abs/2006.13979` doi: 10.48550/ARXIV.2006.13979

Crang, M. (2021). *Crang/WAV2VEC2-large-XLSR-53-frisian · hugging face.* Retrieved from `https://huggingface.co/crang/wav2vec2-large-xlsr-53-frisian`

Dalmia, S., Sanabria, R., Metze, F., & Black, A. W. (2018). Sequence-based multi-lingual low resource speech recognition. In *2018 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 4909–4913).

de Vries, W. (2021). *WIETSEDV/WAV2VEC2-large-XLSR-53-frisian · hugging face.* Retrieved from `https://huggingface.co/wietsedv/wav2vec2-large-xlsr-53-frisian`

Eberhard, D. M., Simons, G. F., & Fennig, C. D. (2021). Ethnologue: Languages of the World. 24th edition. SIL International.

Fukuda, T., & Thomas, S. (2021). Knowledge distillation based training of universal asr source models for cross-lingual transfer. In *Annual conference of the international speech communication association.*

Ghoshal, A., Swietojanski, P., & Renals, S. (2013). Multilingual training of deep neural networks. In *2013 ieee international conference on acoustics, speech and signal processing* (pp. 7319–7323).

Gupta, A., Chadha, H. S., Shah, P., Chhimwal, N., Dhuriya, A., Gaur, R., & Raghavan, V. (2021). Clsril-23: cross lingual speech representations for indic languages. *arXiv preprint arXiv:2107.07402*.

Heigold, G., Vanhoucke, V., Senior, A., Nguyen, P., Ranzato, M., Devin, M., & Dean, J. (2013). Multilingual acoustic models using distributed deep neural networks. In *2013 ieee international conference on acoustics, speech and signal processing* (pp. 8619–8623).

Hendrycks, D., & Gimpel, K. (2020). *Gaussian error linear units (gelus).*

Hjortnæs, N., Partanen, N., Rießler, M., & Tyers, F. M. (2021). The relevance of the source language in transfer learning for asr. In *Proceedings of the workshop on computational methods for* *endangered languages* (Vol. 1, pp. 63–69).

Hou, W., Zhu, H., Wang, Y., Wang, J., Qin, T., Xu, R., & Shinozaki, T. (2021). Exploiting adapters for cross-lingual low-resource speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *30*, 317–329.

Huang, J., Kuchaiev, O., O'Neill, P., Lavrukhin, V., Li, J., Flores, A., ... Ginsburg, B. (2020). Cross-language transfer learning, continuous learning, and domain adaptation for end-to-end automatic speech recognition. *arXiv preprint arXiv:2005.04290*.

Huang, J.-T., Li, J., Yu, D., Deng, L., & Gong, Y. (2013). Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers. In *2013 ieee international conference on acoustics, speech and signal processing* (pp. 7304–7308).

Jang, E., Gu, S., & Poole, B. (2017). *Categorical reparameterization with gumbel-softmax.*

Kingma, D. P., & Ba, J. (2017). *Adam: A method for stochastic optimization.*

Kunze, J., Kirsch, L., Kurenkov, I., Krug, A., Johannsmeier, J., & Stober, S. (2017). Transfer learning for speech recognition on a budget. *arXiv preprint arXiv:1706.00290*.

Loshchilov, I., & Hutter, F. (2019). *Decoupled weight decay regularization.*

Lu, Y., Huang, M., Qu, X., Wei, P., & Ma, Z. (2022). Language adaptive cross-lingual speech representation learning with sparse sharing sub-networks. In *Icassp 2022-2022 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 6882–6886).

Matassoni, M., Gretter, R., Falavigna, D., & Giuliani, D. (2018). Non-native children speech recognition through transfer learning. In *2018 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 6229–6233).

Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019, sep). SpecAugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019*. ISCA. Retrieved from `https://doi.org/10.21437%2Finterspeech.2019-2680` doi: 10.21437/interspeech.2019-2680

Polák, P., & Bojar, O. (2021). Coarse-to-fine and cross-lingual asr transfer. *arXiv preprint arXiv:2109.00916*.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*.

San, N., Bartelds, M., Billings, B., de Falco, E., Feriza, H., Safri, J., ... Jurafsky, D. (2023). Leveraging supplementary text data to kick-start automatic speech recognition system development with limited transcriptions. *arXiv preprint arXiv:2302.04975*.

Schneider, S., Baevski, A., Collobert, R., & Auli, M. (2019). wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*.

Shivakumar, P. G., & Georgiou, P. (2020). Transfer learning from adult to children for speech recognition: Evaluation, analysis and recommendations. *Computer speech & language*, *63*, 101077.

Swietojanski, P., Ghoshal, A., & Renals, S. (2012). Unsupervised cross-lingual knowledge transfer in dnn-based lvcsr. In *2012 ieee spoken language technology workshop (slt)* (pp. 246–251).

Tong, R., Wang, L., & Ma, B. (2017). Transfer learning for children's speech recognition. In *2017 international conference on asian language processing (ialp)* (pp. 36–39).

Tong, S., Garner, P. N., & Bourlard, H. (2017). Multilingual training and cross-lingual adaptation on ctc-based acoustic model. *arXiv preprint arXiv:1711.10025*.

Toshniwal, S., Sainath, T. N., Weiss, R. J., Li, B., Moreno, P., Weinstein, E., & Rao, K. (2018). Multilingual speech recognition with a single end-to-end model. In *2018 ieee international* *conference on acoustics, speech and signal processing (icassp)* (pp. 4904–4908).

Yang, C.-H. H., Li, B., Zhang, Y., Chen, N., Prabhavalkar, R., Sainath, T. N., & Strohman, T. (2023). From english to more languages: Parameter-efficient model reprogramming for cross-lingual speech recognition. *arXiv preprint arXiv:2301.07851*.

Yılmaz, E., Andringa, M., Kingma, S., Dijkstra, J., Kuip, F., Velde, H., … van Leeuwen, D. A. (2016). A longitudinal bilingual Frisian-Dutch radio broadcast database designed for code-switching research.

Yılmaz, E., van den Heuvel, H., & Van Leeuwen, D. (2016). Investigating bilingual deep neural networks for automatic recognition of code-switching frisian speech. *Procedia Computer Science*, *81*, 159–166.

Zhang, Y., Han, W., Qin, J., Wang, Y., Bapna, A., Chen, Z., … Wu, Y. (2023). *Google USM: Scaling Automatic Speech Recognition Beyond 100 Languages.*

Zhang, Z.-Q., Song, Y., Wu, M.-H., Fang, X., & Dai, L.-R. (2021). Xlst: Cross-lingual self-training to learn multilingual representation for low resource speech recognition. *arXiv preprint arXiv:2103.08207*.

Zhou, S., Zhao, Y., Xu, S., Xu, B., et al. (2017). Multilingual recurrent neural networks with residual learning for low-resource speech recognition. In *Interspeech* (pp. 704–708).

Appendices

A Metrics

WER is commonly used in evaluating automatic speech recognition systems and is defined as:

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

where S is the number of words substituted by the ASR, D is the number of deletions (words omitted by the ASR), I is the number of insertions (words recognized by the ASR that were not in the original transcript), and N is the total number of words in the reference/label. The lower the WER is, the better the performance of the model. It is usually expressed as a percentage, and it can reach a value over 100% (the metric has only a lower boundary, 0%).

By convention, the WER results obtained per utterance are averaged and a single score is reported, which is the approach I will also employ in my evaluation. Most of the large-scale models I mentioned in section 2 manage to achieve WERs below 10% for high-resource languages. As for low-resource, scores can range from 10% up to 50%.

Based on WER, the relative word error rate (WER) is also used as a metric. It is calculated as follows:

$$Relative\,WER = \frac{Reference\,WER - Actual\,WER}{Reference\,WER}$$

where $Reference\,WER$ is the WER score that is compared and $Actual\,WER$ corresponds to the WER score measured in a different experiment than the reference. This metric is commonly utilized for comparing various models or experiments, as it takes into account the relative nature of the results in relation to a reference point. This relativity factor makes it particularly valuable when conducting such comparisons.

Absolute WER difference is another metric based on WER results. The formula is:

$$Absolute\,WER = |Reference\,WER - Actual\,WER|$$

Although the difference is not relative to the performance of the model we compare to, it is as important as relative WER in comparisons of performance between models as it reports the real difference in two WER scores.

B Data Analysis

In this appendix, detailed tables covering every training subset for all experiments, as well as the development and test subsets, can be found.

Table 5: Analysis of the **development** data, based on age and sex.

Sex Age	Unknown	Female	Male	**Total**
Unknown	3h15m37s	0.0	0.0	3h15m37s
18-19	0.0	1m8s	0.0	1m8s
20-29	0.0	4m6s	3m27s	7m33s
30-39	0.0	13m8s	4m35s	17m43s
40-49	0.0	10m55s	2m6s	13m1s
50-59	1m6s	8m12s	7m4s	16m22s
60-69	37s	15m15s	3m26s	19m18s
70-79	0.0	0.0	1m46s	1m46s
80-89	0.0	0.0	1m6s	1m6s
Total	3h17m20s	52m44s	23m30s	**4h32m34s**

Table 6: Analysis of the **test** data, based on age and sex.

Sex Age	Unknown	Female	Male	**Total**
Unknown	3h27m7s	0.0	0.0	3h27m7s
20-29	0.0	4m55s	4m13s	9m8s
30-39	0.0	10m43s	2m19s	13m2s
40-49	0.0	10m32s	4m10s	14m42s
50-59	1m14s	7m36s	4m59s	13m49s
60-69	0.0	9m1s	3m17s	12m18s
80-89	0.0	0.0	13s	13s
Total	3h28m21s	42m47s	19m11s	**4h30m19s**

Table 7: Analysis of the **train** data of **experiment 1**, based on age and sex.

Sex Age	Unknown	Female	Male	**Total**
Unknown	3h15m54s	0.0	0.0	3h15m54s
20-29	0.0	1m56s	8m51s	10m47s
30-39	0.0	14m1s	2m15s	16m16s
40-49	0.0	32m19s	0.0	32m19s
50-59	0.0	6m4s	15m58s	22m2s
60-69	0.0	24m16s	1m51s	26m7s
70-79	0.0	0.0	26s	26s
Total	3h15m54s	1h18m36s	29m21s	**5h3m51s**

Table 8: Analysis of the **train** data of **experiments 2, 6, and 7**, based on age and sex.

Sex Age	Unknown	Female	Male	**Total**
Unknown	23h8m11s	0.0	0.0	23h8m11s
18-19	0.0	1m54s	0.0	1m54s
20-29	0.0	28m57s	18m1s	46m58s
30-39	0.0	1h33m34s	20m39s	1h54m13s
40-49	0.0	3h39m33s	5m46s	3h45m19s
50-59	10s	2h35m51s	1h47m6s	4h23m7s
60-69	18m1s	6h10m59s	24m47s	6h53m47s
70-79	0.0	0.0	3m59s	3m59s
80-89	0.0	0.0	49s	49s
Total	23h26m22s	14h30m48s	3h1m7s	**40h58m17s**

Table 9: Analysis of the **train** data of **experiment 3**, based on age and sex.

Sex Age	Unknown	Female	Male	**Total**
Unknown	5h40m4s	0.0	0.0	5h40m4s
18-19	0.0	23s	0.0	23s
20-29	0.0	8m7s	4m30s	12m37s
30-39	0.0	22m34s	4m12s	26m46s
40-49	0.0	54m35s	1m39s	56m14s
50-59	0.0	36m26s	27m2s	53m28s
60-69	4m43s	1h28m10s	6m25s	1h39m18s
70-79	0.0	0.0	1m9s	1m9s
80-89	0.0	0.0	21s	21s
Total	5h44m47s	3h30m15s	45m18s	**10h20s**

Table 10: Analysis of the **train** data of **experiment 4**, based on age and sex.

Sex Age	Unknown	Female	Male	**Total**
Unknown	36m36s	0.0	0.0	36m36s
60-69	16s	7m46s	53s	8m55s
50-59	0.0	3m18s	2m34s	5m52s
40-49	0.0	4m59s	17s	5m16s
30-39	0.0	2m4s	13s	2m17s
20-29	0.0	32s	20s	52s
70-79	0.0	0.0	12s	12s
Total	36m52s	18m39s	4m29s	**60m**

Table 11: Analysis of the **train** data of **experiment 5**, based on age and sex.

Sex Age	Unknown	Female	Male	**Total**
Unknown	5m28s	0.0	0.0	5m28s
20-29	0.0	8s	5s	13s
30-39	0.0	17s	12s	29s
40-49	0.0	57s	4s	1m1s
50-59	0.0	40s	33s	1m13s
60-69	0.0	1m28s	9s	1m37s
Total	5m28s	3m30s	1m3s	**10m1s**

C Text Preprocessing

The Frisian orthography, as extracted from the text labels, is:

a b c d e f g h i j k l m n o p q r s t u v w x y z à á â ä è é ê ë ï ô ú û ü

Some characters are not part of the Frisian alphabet and appear in transcripts due to certain loanwords that contain them.

For experiment 3, the character missing from the orthography above is 'à'. For experiment 4, the characters missing are 'à' and 'ü'. For experiment 5, the characters missing are 'à', 'ü', and 'x'.

Table 12: Examples of preprocessed sentences

Base sentence	**Normalized sentence**
Hoe witte jim dêr no krekt goed it paad yn te finen?	hoe witte jim dêr no krekt goed it paad yn te finen
Ik tink net dat it wat oplost, mar fan my mei it.	ik tink net dat it wat oplost mar fan my mei it
Sneontemiddei is in bekende reisboekeskriuwer te gast yn de nije biblioteek.	sneontemiddei is in bekende reisboekeskriuwer te gast yn de nije biblioteek
De Nederlânske wettersektor leit ynternasjonaal op in tige goede namme.	de nederlânske wettersektor leit ynternasjonaal op in tige goede namme
As bern siet er altyd al op kop en earen yn de boeken.	as bern siet er altyd al op kop en earen yn de boeken
De behanneling betsjut ek dat libbenslang medisinen jûn wurde moatte.	de behanneling betsjut ek dat libbenslang medisinen jûn wurde moatte
De hinnereis is mei de nachttrein en de weromreis mei de deitrein.	de hinnereis is mei de nachttrein en de weromreis mei de deitrein
Oan de foarstelling dogge ferskate keunstriders mei.	oan de foarstelling dogge ferskate keunstriders mei
Lokwinske mei dyn nije baan.	lokwinske mei dyn nije baan

D Research Proposal

The proposal can be found on the next page; it was pushed there due to the pdf import.

Research proposal: Improving the State-of-the-Art Frisian ASR using Large-Scale Cross-Lingual Pre-Trained Models

Dragoș Alexandru Bălan

April 7, 2023

Abstract

Frisian is a West Germanic language recognized as an official language in the province of Fryslan in the Netherlands. Despite its official status, technological support and resources for Frisian are scarce to non-existent, especially in the field of automatic speech recognition (ASR). Thus, it is considered a low-resource language, and approaches different from high-resource speech recognition need to be employed. This research aims to contribute to the field of low-resource language speech recognition by improving upon the current state-of-the-art Frisian ASR performance of 16.25% word error rate (WER) by fine-tuning a large-scale cross-lingual pre-trained model. Specifically, my research will answer the following question: Can fine-tuning the XLS-R model on Frisian speech achieve a WER below 20% and outperform the state-of-the-art XLSR-53 model? I hypothesize that I will be able to achieve a WER under 20% and set a new state-of-the-art, as further validated by my pilot study in which I managed to fine-tune a preliminary model and achieved a 15.99% WER. In case my hypothesis is invalidated, it will challenge the research done on large-scale cross-lingual models and encourage research into different approaches for low-resource context speech recognition.

Keywords: Frisian speech recognition, transfer learning, XLS-R, cross-lingual, low resource speech, fine-tune, XLSR-53

Contents

1 Introduction

There are 7608 languages in the world, out of which only 7168 are still actively spoken (Eberhard et al., 2021). Only less than 1% of them have enough speech resources to build automatic speech recognition (ASR) systems. Such languages that have large corpora of speech available are called high-resource. However, many other languages in the world have limited or no speech data available, making them low-resource or zero-resource languages. Developing ASR systems for these languages is difficult because it requires hundreds to thousands of hours of speech data to train a model that can perform well and be robust to new input. Such languages lack the resources of higher-resourced ones and collecting more data is a tedious process that involves approvals from ethics committees to conduct the process, finding a wide range of speakers to record, as well as transcribing the speech using linguist experts.

There are numerous reasons as to why researchers should develop more ASR models for low-resource languages. Some of them include preserving endangered languages or enhancing access to technology for people who speak low-resourced languages. Applications can range from teaching people from a young age how to speak these languages to creating robots that could communicate with humans via voice, something that would be particularly useful to people of age since their computer literacy is lower than that of other age groups.

A language that is considered to be low-resource and the one that will be covered by this research is Frisian, or Frysk (IPA: [frisk]) as the speakers call it. Frisian is a West Germanic language which is recognized as the official language of the province of Fryslân in the Netherlands. It is also known as West Frisian, in order to distinguish it from other dialects that are spoken in certain areas in the north of Germany. Although Frisian is taught in schools and used in different media outlets like radio or TV, it still lacks support when it comes to recorded and labeled speech.

Thus, different approaches other than data collection need to be employed in order to develop speech recognition systems for such cases of under-resourced languages. Some examples are data augmentation, where more data is generated from already-existing sources, or transfer learning. Transfer learning refers to using a model that was trained on a higher-resourced language or context and adapting the knowledge of that by fine-tuning the model on a lower-resourced language or context.

The first models that were used in transfer learning were monolingual models, pre-trained on a single source language, then fine-tuned on a target language. Newer approaches involve training a model on larger data from multiple languages, thus learning how to recognize speech in a multilingual or cross-lingual context. Such models manage to achieve impressive performances when fine-tuned on languages with a significantly smaller number of hours of data (under 100 hours).

As for the reference baseline for Frisian speech recognition that I will be using, there have been some attempts at fine-tuning XLSR-53, a large-scale cross-lingual model that learns language-independent speech representations (Conneau et al., 2020). The best-performing models are determined to be the works of de Vries (2021) and Crang (2021). Both of the models were fine-tuned in parallel by their respective authors in 2021, but de Vries has managed to achieve a lower word error rate (WER), the metric used in speech recognition evaluation. The developer's WER was 16.25%, which corresponds to the current state-of-the-art in Frisian ASR and which I plan to improve upon. More specifically, my research aims to improve Frisian speech recognition by fine-tuning a larger cross-lingual model and analyzing the amount of data needed to achieve significant performance gains.

This proposal is structured as follows: section 2 provides a brief literature review. Section 3 presents the research question, along with subquestions derived from it and a falsifiable hypothesis. Then, section 4 describes the execution of the research and outlines the timeline. Section 5 discusses risks and solutions for mitigating them. In section 6, I describe the ethical issues regarding my research. Lastly, section 7 covers the impact and relevance of my study.

2 Literature review

In this section, I will briefly mention the keywords used throughout my literature search. Then, I will explore the criteria used for the selection of papers to be reviewed in this proposal. Lastly, I will provide a short synthesis of each of the chosen academic papers.

I have grouped the keywords according to the topic they are related to. The topics are highlighted in bold, after which the keywords for that topic are mentioned. The search has been mainly conducted on Google Scholar. Thus, the topics and their corresponding keywords are:

Datasets: Frisian speech, Frisian speech corpus, Frisian speech dataset;
Cross-lingual models: cross(-)lingual speech (recognition), cross(-)lingual ASR, multilingual speech (recognition), multilingual ASR;
Transfer learning: transfer learning ASR, transfer learning speech recognition;
Frisian ASR: Frisian ASR, Frisian (automatic) speech recognition.

Aside from the literature found using the keywords above, I also investigated the references section of each paper and extracted more relevant research. Other filters I have used are for the publications to be as recent as possible (from 2010 onwards) and I have picked results from the top 20 with the most citations.

Since the number of references I found is quite large and some references are older or not as relevant to the proposal, I decided to select only a subset of them. The literature described in this proposal is the most recent (published after 2019) or describes the groundwork for the latest research. The references relate to the datasets and the cross-lingual models that I considered, as well as the underlying architecture of the model I have chosen.

Reference	Brief description
Yilmaz et al., 2016	FAME!, Dutch-Frisian code-switched speech corpus
Ardila et al., 2020	Common Voice, massive multilingual crowdsourced corpus, used in methodology
Huang et al., 2013	SHL-MDNN, early attempt at cross-lingual ASR, hidden layers shared between languages, multiple output layers for each language
Heigold et al., 2013	Early attempt at cross-lingual ASR, DNN-HMM with bottom 3 hidden layers shared
Ghoshal et al., 2013	Early attempt at cross-lingual ASR, trained sequentially, a language after another
Schneider et al., 2019	wav2vec, unsupervised learning of speech representations
Baevski et al., 2020	wav2vec 2.0, self-supervised learning of speech representations, used as underlying model of XLSR-53 and XLS-R
Conneau et al., 2020	XLSR-53, large-scale cross-lingual model, used in state-of-the-art Frisian ASR
Babu et al., 2021	XLS-R, larger-scale cross-lingual model than XLSR-53, used in methodology
Radford et al., 2022	Whisper, larger-scale cross-lingual model than XLS-R
Zhang et al., 2023	Google USM, latest large-scale cross-lingual model, state-of-the-art results
de Vries, 2021	State-of-the-art Frisian ASR, WER of 16.25%
Crang, 2021	Similar attempt to state-of-the-art Frisian ASR, WER of 19.11%

Table 1: List of references, summarized

For simplicity and readability, table 1 provides a list of references appended with some notes, sorted by order of appearance in the following subsections of the literature review.

2.1 Frisian speech datasets

Firstly, the literature I have found regarding the available speech corpora for Frisian consists of mainly two references: the FAME! corpus (Yilmaz et al., 2016) and Common Voice (Ardila et al., 2020). FAME! is a Frisian-Dutch bilingual corpus of annotated speech collected from radio broadcasts. The quantity of data available in the corpus is 18.5 hours and most of the recordings involve code-switching between Frisian and Dutch. The other corpus, Common Voice, is an open-source, crowdsourced multilingual project with over 100+ languages available designed for speech recognition purposes. The users can contribute to the project by recording themselves speak a given prompt or by validating other users' recordings. The downside is that the recordings can be noisy and the validation depends on the users that contribute to the corpus rather than trained experts. However, the availability and quantity of data makes it a corpus worth considering. Therefore, I will work with Common Voice since it has more speech available than FAME! and it contains exclusively monolingual Frisian recordings. More information about the corpus will be provided in the methodology section.

2.2 Early works in cross-lingual ASR

Cross-lingual ASR is a relatively new subfield. One of the very first works that achieved one model that can be used for speech recognition in different languages is by Huang et al. (2013). In this paper, the authors employ a deep neural network where the hidden layers are trained on multiple languages in parallel (the hidden layers are shared), but there are several output layers, each one of them fine-tuned on a different language. The architecture was coined Shared-Hidden-Layer Multilingual Deep Neural Network, or SHL-MDNN.

Other models that have been researched in parallel, in the same year, are a DNN-HMM hybrid system where the bottom three hidden layers of the DNN were trained on data from multiple languages (Heigold et al., 2013) and a DNN architecture where the hidden layers are trained sequentially, one language after another (Ghoshal et al., 2013). This research from three different groups has set the groundwork for cross-lingual ASR.

2.3 wav2vec 2.0

The subfield has seen exponential growth in the past two years. This is mainly attributed to advancements in unsupervised and self-supervised speech feature representation, as well as to the rapid growth in data and computational power. The first unsupervised speech representation extraction model is wav2vec (Schneider et al., 2019). Wav2vec takes as input raw speech and feeds it through two convolutional neural networks, one context network that is stacked on top of an encoder network. The authors then use the outputted representations as input to the acoustic model of an ASR. They show that a model that uses representations from wav2vec instead of other feature methods achieves a lower WER.

Its successor, wav2vec 2.0, obtains features from raw speech in a self-supervised manner (Baevski et al., 2020). It does so by replacing the context network with a transformer architecture and by using an additional quantization module which, together with the output of the transformer, trains the model by attempting to minimize a contrastive loss between the two representations. Paired with a language model on top, it manages to achieve below 10% WER on large datasets, even when using as low as 10 minutes of labeled data.

2.4 Cross-lingual models based on wav2vec 2.0

Wav2vec 2.0 was used in training XLSR-53, one of the first and most popular large-scale cross-lingual ASR models (Conneau et al., 2020). The authors have managed to draw several conclusions after analyzing the results obtained on different datasets, but the most important is that low-resource languages can benefit greatly from a model trained on large multilingual data. When the model is pre-trained on all data from 53 languages and then fine-tuned for the target low-resource language, it outperforms monolingual models. A downside is found for high-resource languages which suffer from interference due to the model being trained on scarce data from other languages. Since this study is aimed at improving low-resource ASR, the weakness of XLSR-53 is not as relevant as its strength.

The same research team worked further to create XLS-R, an even larger model than XLSR-53 (Babu et al., 2021). Trained on almost half a million hours of data from 128 languages, it manages to achieve state-of-the-art results on speech translation, speech recognition, and language identification tasks. It outperforms the previously-mentioned model on both low- and high-resource data, which makes it a powerful model to use for developing a Frisian ASR system.

2.5 Other cross-lingual models

There have been other large-scale cross-lingual models released by other research groups. Some of them are Whisper, trained on 680,000 hours of data from 96 languages and using the Transformer architecture (Radford et al., 2022), and the most recent model released, Google USM, trained on 12 million hours of speech from 300 languages and using a convolution-augmented Transformer (Conformer) architecture (Zhang et al., 2023). They both show promising results and have improved performance compared to XLS-R, but the research group of XLS-R provides extensive tutorials and support for their model, which makes it a more fitting choice for the research I plan to conduct.

2.6 State-of-the-art Frisian ASR

The current state-of-the-art for Frisian speech recognition, as mentioned in the introduction, is the work of de Vries (2021). The author manages to achieve a WER of 16.25% by fine-tuning XLSR-53 on the Frisian subset of Common Voice (Ardila et al., 2020). A similar attempt was made in the same period by a different author. Crang (2021) managed to obtain a higher WER than de Vries, which is 19.11%. The main difference in these two attempts is that de Vries applies a 10% activation layer dropout probability, whereas Crang does not apply any dropout. Since de Vries achieves a lower WER, I will be using his model as the reference baseline for my experiments.

This concludes the literature review section, which provides an extensive overview of the early works in cross-lingual ASR and the current state-of-the-art in Frisian ASR. While previous studies have contributed to our understanding of Frisian speech recognition, there are still gaps in knowledge and room for improvement that need to be addressed, which is what I attempt to do. Specifically, I will be looking into fine-tuning a larger-scale cross-lingual model and, in doing so, establishing a new state-of-the-art in Frisian ASR.

3 Research question and hypothesis

The research question is the following:

> **Can fine-tuning the XLS-R model on Frisian speech achieve a WER below 20% and outperform the state-of-the-art XLSR-53 model?**

From which the following subquestions are derived:

- What is the baseline WER achieved by fine-tuning XLSR-53 on Frisian speech?
- Can the XLS-R model with 1B parameters achieve a lower WER than XLSR-53 on Frisian speech?
- How does the size of the training data used for fine-tuning affect the performance of the model on Frisian speech?

My hypothesis is that fine-tuning the XLS-R model on Frisian speech using specific hyperparameters and a sufficient amount of training data will result in a WER below 20%, outperforming the state-of-the-art XLSR-53 model. The latest developments in low-resource transfer-learning ASR involve models trained on multiple languages, also known as multilingual or cross-lingual models, which leverage massive amounts of data (tens to hundreds of thousands of hours) and use it effectively for low-resource languages (Babu et al., 2021; Conneau et al., 2020). The latest and most relevant research found is that of XLS-R, a set of models trained on 436,000 hours of speech in 128 different languages, both high- and low-resourced (Babu et al., 2021). The quantity and variety of data used to pre-train the models prove to outperform the previous version of this model trained on fewer data and fewer languages, XLSR-53 (Conneau et al., 2020).

There are already several XLSR-53 models fine-tuned on Frisian that manage to achieve results below 20% WER (Crang, 2021; de Vries, 2021), therefore I expect to achieve a comparable, if not lower, error rate that would also fall under the 20% threshold. Furthermore, the current state-of-the-art for Frisian ASR is the work of de Vries, 2021, who achieves a WER of 16.25%. I anticipate outperforming that score by fine-tuning XLS-R on Frisian.

In case the hypothesis is falsified, it can be implied that XLS-R is not a suitable choice for fine-tuning a Frisian ASR model and that XLSR-53 is a more suitable choice. It could also partly contest the work of Babu et al. (2021), implying that pre-training larger and larger models does not necessarily translate to better results.

4 Execution

This section will talk about how the research question will be answered and the hypothesis, validated. First, I will concretely describe the methodology to be employed to achieve my goal. Then, a timeline of tasks that I plan to do is presented, along with deliverables and deadlines.

4.1 Methodology

I will take the following steps to answer my research question:

1. **Find and use a public dataset of Frisian speech;**

   I plan to use the Common Voice corpus (Ardila et al., 2020). As mentioned in section 2, Common Voice is a multilingual, open-source, and crowdsourced dataset recorded and validated by its users. I will be using only the Frisian subset, labeled as 'fy-NL' in the dataset. Common Voice has multiple versions available to download on their website[1] since they release the dataset frequently. Each version adds more data on top of the previous release. Therefore, it is important to mention that I will be using two versions of the Frisian subset: the release that was available at the time when de Vries developed their model (de Vries, 2021), which corresponds to version 8.0, and the most recent release, which is version 13.0 at the time of writing this. I will provide a more extensive explanation for choosing these two versions in step 3.

2. **Split the data into train, development, and test sets;**

   Each language contains several '.tsv' files, which represent the metadata about the recordings and their respective speakers. Common Voice provides train, evaluation, and test splits, which simplifies the data preprocessing step for my experiments. It also contains a file for invalidated speech samples which I will not be working with since they might contain noisy data, speech in a different language, or no speech at all and this will negatively impact the fine-tuning step.

3. **Preprocess the data;**

   For preprocessing, I need to normalize the labels, as well as resample the audio, such that both are compatible with the models I plan to experiment with. More details about the process will be provided in the thesis.

[1] https://commonvoice.mozilla.org/en/datasets

4. **Choose a suitable metric to measure the performance of the models during the experiments;**

   I plan to analyze the results using word error rate, or WER, as the performance metric. WER is commonly used in evaluating automatic speech recognition systems and is defined as:

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

   Where S is the number of words substituted by the ASR, D is the number of deletions (words omitted by the ASR), I is the number of insertions (words recognized by the ASR that were not in the original transcript), and N is the total number of words in the reference/label. The lower the WER is, the better the performance of the model. It is usually expressed as a percentage, and it can reach a value over 100% (the metric has only a lower boundary, 0%).

   All of the large-scale models I mentioned in section 2 manage to achieve WERs below 10% for high-resource languages. As for low-resource, scores can range from 10% up to 50%.

5. **Replicate the results of the XLSR-53 baseline model;**

   In order to have a baseline reference for my experiments, I will evaluate the XLSR-53 model fine-tuned on Frisian by de Vries (2021). The model is publicly available on Huggingface[2]. I will first replicate the experiment on version 8.0 of the dataset in order to confirm the reported result by de Vries. Then, I will evaluate the model on the latest version, 13.0, since this will be the version that I will use for fine-tuning the XLS-R model. Thus, the latter will serve as the reference baseline because the same test set will have been used for both de Vries' model, as well as my fine-tuned one.

6. **Fine-tune the larger-scale 1B XLS-R model on all of the Frisian Common Voice data;**

   How I plan to improve upon the baseline is to fine-tune the pre-trained 1B parameters XLS-R model. I will implement this using Huggingface. I will conduct several experiments with different hyperparameter settings. The hyperparameters will not be covered in much detail in this proposal since they will be adapted based on the outputs of the model, but a detailed description will be provided in the thesis. The current plan is to use the default hyperparameters mentioned in the paper of the model (Babu et al., 2021). After this, I hope to get results that outperform the current state-of-the-art or achieve a WER of under 20%.

7. **Fine-tune for lower numbers of hours of Frisian speech;**

   One of my subquestions mentions the size of the data used for fine-tuning the model and its impact on the performance. To answer this, I will be fine-tuning and evaluating the model with two more settings aside from using all of the data:

   - Using only 10 hours of data
   - Using only 1 hour of data

   The main reasoning behind choosing these values is because the researchers behind wav2vec 2.0 also used these amounts of data when conducting their experiments (Baevski et al., 2020). By fine-tuning with different numbers of hours of speech, I plan to show that more data is needed and will provide significantly better results, even when we fine-tune a large-scale model such as XLS-R.

8. **(OPTIONAL) Fine-tune the 2B XLS-R model;**

   Given that there is already a large list of parameters and experiments to conduct, it may not be feasible to do furthermore. However, if time allows it, I plan to also experiment more using a larger model. I can fine-tune the 2B parameter version of the pre-trained XLS-R models

[2] https://huggingface.co/wietsedv/wav2vec2-large-xlsr-53-frisian

available. This model provides state-of-the-art results on most tasks and I expect it to perform even better than the 1B parameter model (Babu et al., 2021). However, due to the number of parameters being double the size, it may not be as feasible to conduct this experiment.

9. **(OPTIONAL) Use more metrics for evaluation and analysis.**

   Other two metrics that can be used alongside WER are phone error rate (PER) and character error rate (CER). They work in a similar fashion to WER, but on different levels. PER checks the error rate of the phonemes identified, whereas CER checks the error rate of the characters outputted by the system. Since the baseline model I will use does not have CER or PER reported by default and since these metrics are less often used than WER, it may not be as viable to evaluate the models this way.

4.2 Timeline

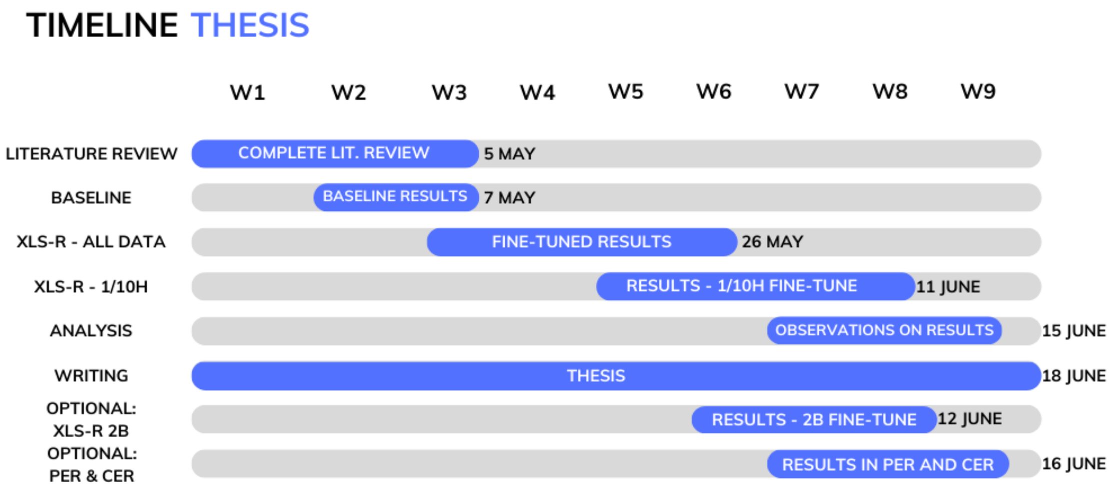

Figure 1: Gantt chart of the thesis timeline.

A Gantt chart illustrating the timeline of my activities during the thesis period can be found in figure 1. On the left of the progress bars, the tasks are listed. Above the bars, the weeks of the thesis term can be seen. Each task has a blue progress bar that represents the duration of the task; inside of it is the deliverable expected. To the right of the blue bars, we can see some dates, which correspond to the exact deadlines for each of the tasks. I grouped all of the writing into one process since several sections of it will be revised constantly throughout the thesis.

5 Risk mitigation

In this section, I will cover the risks I could face during my thesis work, as well as solutions to mitigate them. Then, I will describe the pilot study conducted to demonstrate the feasibility of my research proposal.

5.1 Risks and contingencies

An overview of the three most important risks that this project may face is summarized in table 2, with each risk categorized by severity and likelihood. In this section, I will address each risk respectively.

Risks	Very likely	Likely	Not likely
Very severe			
Severe	Issues with replicating baseline	Fine-tuning technical difficulties	
Not severe	Limited availability of data		

Table 2: Risks and contingencies

The common risk that researchers encounter when working with low-resource languages is, as the name suggests, the scarcity of data. This is also applicable to Frisian since the amount of data available in Common Voice is 67 hours of speech. However, the number of hours is not too low, and it is also one of the motivators for my research. Thus, I will mitigate this issue by fine-tuning large-scale cross-lingual ASR models on Frisian, hoping that using knowledge from a wide range of languages and many hours of data will help tremendously with achieving a WER below 20%.

Another risk that could arise is not being able to fine-tune XLS-R due to technical difficulties. I plan to mitigate this risk by starting early with my experiments, as well as contacting my supervisors in case I am confronted with issues that are very difficult to fix. In case the issues are impossible to fix, I can consider using other newly-released large-scale cross-lingual models, as covered in section 2.

As for the remaining risk, replicating the baseline model's results might pose a challenge because of older dependencies that are required, but are not fully disclosed by the author (de Vries, 2021). To mitigate this, I will backtrack to when the model was released and what were the latest versions of the dependencies at the time of the release. In case there are still issues with replicating, I will contact my supervisors for assistance. As a failsafe, I will simply use the reported score of the baseline and fully disclose my process in the thesis.

5.2 Pilot

As part of a term paper for a different course and together with Golshid Shekoufandeh, we were able to fine-tune a preliminary XLS-R model on Frisian. The hyperparameters chosen are more or less the default ones found in the Huggingface framework, except for the learning rate, which has been set at a value of 8e-5.

This preliminary model has been fine-tuned and evaluated on Common Voice 12.0, which contains 55 hours of validated Frisian speech. We chose 12.0 because, at the time of experimenting, version 13.0 was not released yet. We have achieved so by adapting a tutorial for fine-tuning XLS-R on Turkish Common Voice data, created by Patrick von Platen (2021). The preprocessing of labels and audio has been done similarly to the tutorial.

Model	**WER (%)**
Baseline XLSR-53 Model	16.25%
Fine-tuned XLS-R 1B Model	**15.99%**

Table 3: WERs of the models

The results are reported in WER. We used the reported score of the baseline, tested on Common Voice 8.0, and the score we obtained for our fine-tuned model, evaluated on Common Voice 12.0. Table 3 shows a comparison of the results.

As can be observed, the fine-tuned XLS-R model outperforms the current state-of-the-art for Frisian ASR by 0.26% which, although the improvement is small, shows promise and potential. Thus, the pilot study proves the feasibility of my project and shows potential for it to answer the research question and validate my hypothesis.

6 Ethical issues

While the study aims to develop an ASR system that will benefit the Province of Fryslân, there is a possibility that the technology may have unforeseen consequences. In order to mitigate the risks, the research team will communicate the study's results and implications in an accessible and transparent manner.

I will not be collecting any sort of data from human participants. Instead, I will be using a previously-recorded dataset, which is Mozilla's Common Voice project (Ardila et al., 2020). It is a multilingual, open-, and crowdsourced corpus that is constantly updated, with support for over 100 languages. The participants in the Common Voice project are informed about their data being collected and they do so voluntarily. The recordings are also validated by the community. The corpus is licensed under CC0, therefore any distribution, adaptation, or otherwise may be made freely, without having to credit or mention in any way.

I will not be involving human participants in evaluating the ASR models developed either since objective metrics are used that are more relevant to the field and the use of subjective evaluation methods involving human participants is not as meaningful. Therefore, there are no concerns regarding the ethics of involving human participants or any other issues that do not align with the ethics of the faculty. If at any point in time data from human participants needs to be collected, I will conduct the necessary steps for approvals from the ethics committee.

As when it comes to the replicability of the research, all of the code will be made available via GitHub and all steps and details on how to reproduce the experiments described in the proposal can be found under the Methodology section. The dataset is publicly available to download and use. The outcomes should be more or less similar, but they may not be exactly the same due to certain elements that introduce randomness in the trained models. The hardware used may also impact the performance of the models since the experiments will be conducted on the university's high-performance cluster.

7 Impact and relevance

The impact of my research will be significant to the Frisian community. If successful, not only will it set a new state-of-the-art when it comes to Frisian speech recognition, but it will also validate the recent developments in cross-lingual models for low-resource ASR tasks, that pre-training larger models on a wider variety of data then fine-tuning these models will offer better performance for low-resource tasks.

Furthermore, there are several applications where my research could be used. For example, the resulting model could be incorporated into a virtual assistant that could be used at museums, info booths, or to assist the elderly in a care home. Another application is assisting language learners who are curious to read, speak, and understand Frisian. All of these applications will help preserve the language and culture of the province of Fryslân by encouraging its speakers to communicate in it more in their daily lives rather than use Dutch and contribute further to the extinction of Frisian.

If my hypothesis will be invalidated, then it could indicate that researchers should focus more on different approaches to achieve performant low-resource speech recognition. It will also partly invalidate the recent work that has been done in pre-training larger and larger cross-lingual models and conclude that such models can not be used as an all-in-one tool.

References

Ardila, R., Branson, M., Davis, K., Kohler, M., Meyer, J., Henretty, M., Morais, R., Saunders, L., Tyers, F., & Weber, G. (2020). Common Voice: A Massively-Multilingual Speech Corpus. *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 4218–4222.

Babu, A., Wang, C., Tjandra, A., Lakhotia, K., Xu, Q., Goyal, N., Singh, K., von Platen, P., Saraf, Y., Pino, J., et al. (2021). XLS-R: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*.

Baevski, A., Zhou, H., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. https://doi.org/10.48550/ARXIV.2006.11477

Conneau, A., Baevski, A., Collobert, R., Mohamed, A., & Auli, M. (2020). Unsupervised Cross-lingual Representation Learning for Speech Recognition. https://doi.org/10.48550/ARXIV.2006.13979

Crang, M. (2021). Crang/WAV2VEC2-large-XLSR-53-frisian · hugging face. https://huggingface.co/crang/wav2vec2-large-xlsr-53-frisian

de Vries, W. (2021). WIETSEDV/WAV2VEC2-large-XLSR-53-frisian · hugging face. https://huggingface.co/wietsedv/wav2vec2-large-xlsr-53-frisian

Eberhard, D. M., Simons, G. F., & Fennig, C. D. (2021). Ethnologue: Languages of the World. 24th edition.

Ghoshal, A., Swietojanski, P., & Renals, S. (2013). Multilingual training of deep neural networks. *2013 IEEE international conference on acoustics, speech and signal processing*, 7319–7323.

Heigold, G., Vanhoucke, V., Senior, A., Nguyen, P., Ranzato, M., Devin, M., & Dean, J. (2013). Multilingual acoustic models using distributed deep neural networks. *2013 IEEE international conference on acoustics, speech and signal processing*, 8619–8623.

Huang, J.-T., Li, J., Yu, D., Deng, L., & Gong, Y. (2013). Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers. *2013 IEEE international conference on acoustics, speech and signal processing*, 7304–7308.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*.

Schneider, S., Baevski, A., Collobert, R., & Auli, M. (2019). wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*.

von Platen, P. (2021). Fine-tune XLSR-WAV2VEC2 for low-resource ASR with Huggingface transformers. https://huggingface.co/blog/fine-tune-xlsr-wav2vec2

Yilmaz, E., Andringa, M., Kingma, S., Dijkstra, J., Kuip, F., Velde, H., Kampstra, F., Algra, J., Heuvel, H., & van Leeuwen, D. A. (2016). A longitudinal bilingual Frisian-Dutch radio broadcast database designed for code-switching research.

Zhang, Y., Han, W., Qin, J., Wang, Y., Bapna, A., Chen, Z., Chen, N., Li, B., Axelrod, V., Wang, G., Meng, Z., Hu, K., Rosenberg, A., Prabhavalkar, R., Park, D. S., Haghani, P., Riesa, J., Perng, G., Soltau, H., ... Wu, Y. (2023). Google USM: Scaling Automatic Speech Recognition Beyond 100 Languages.