BACHELOR THESIS

ARTIFICIAL INTELLIGENCE



|  |
| --- |
| Adaption and Analysis of Wav2Vec2 on Air Traffic Control Communication Data |

*First supervisor:*

dr. L.F.M. ten Bosch

Taal en communicatie

[louis.tenbosch@ru.nl](mailto:louis.tenbosch@ru.nl)

*Second supervisor:*

Dr. A. B. LAST NAME

Department

E-mail

*Author:*

Karan Chand

Student number: s1033357



28 August 2022

**Abstract**

The abstract of your thesis is a brief description of the research hypothesis, scientific context, motivation, and results.

The size of an abstract is usually one paragraph or one page of text.

* Recent work
* domain shift
* data evaluation
* xlsr model – why -> focus on multilingual, so accents good?
* wer and cer reduction
* study impact of size of data on finetune performance
* impact of arpa (maybe on other finetuned models as well)

Contents

[Introduction 4](#_Toc111542462)

[Preliminaries 5](#_Toc111542463)

[Related work 6](#_Toc111542464)

[Methodology 7](#_Toc111542465)

[Experiments 8](#_Toc111542466)

[Data 8](#_Toc111542467)

[Settings 8](#_Toc111542468)

[Results 9](#_Toc111542469)

[Conclusions 10](#_Toc111542470)

[Bibliography 10](#_Toc111542471)

[Appendix 11](#_Toc111542472)

# Introduction

The introduction of your bachelor thesis introduces the research area, the research hypothesis, and the scientific contributions of your work.

The following things are usually in an introduction:

* Describe the problem and state the research question;
* Motivate why this problem must be solved;
* Demonstrate that a (new) solution is needed;
* Explain the intuition behind your solution;
* Motivate why / how your solution solves the problem (this is technical);
* Explain briefly how it compares with related work.

The introduction is closed with a paragraph in which the content in the next chapters is briefly mentioned to give an overview. A guideline is one sentence per chapter.

When travelling, the safety of the passengers depends on the drivers of the vehicle. For example, in trains and in airplanes communication between the drivers/pilots and the traffic control is crucial. Natural language processing can be used to make their job easier and less error prone by transcribing the natural language. Call signs for vehicles and directions are essential and errors should be minimized. Automatic Speech Recognition (ASR) is a perfect fit for this problem, as the transcriptions should be made in real time. There are many different models for ASR, but there is still a gap in the domain of Air Traffic Control (ATC).

Sequence-to-sequence models are commonly used for ASR applications, as they handle sequences of data well and can predict future data. Artificial Neural Networks (ANN), Recurrent Neural Network (RNN) and Hidden Markov Models (HMM) have among others been used for ASR(Trentin & Gori, 2001) (Zuluaga-Gomez et al., 2020). The state-of-the-art transformer architecture has been shown to give a lower Word Error Rate (WER) than RNN based algorithms. WER is the metric for the percentage of words that are incorrectly classified. Transformer models normally only work when a complete sentence/phrase is used as input. This is because the encoder/decoder transformer model predicts the current word using the previous and the following words. The transformer model has been adjusted(Moritz et al., 2020) to be able to use this in an online setting. Using the transformer in real-time can be done by time restricted self-attention on the encoder and triggered attention for the encoder-decoder mechanism. This way, the architecture can be used for ATC communication. To help ATC, online ASR is required to quickly process the communication between pilots and controllers. ATC can benefit from this architecture, because communication errors can be very costly. Reducing read-back errors in ATC have been researched before using an RNN and Long Short-Time Memory networks(JIA et al., 2018)(Lin et al., 2021). These models are able to get a relatively low WER, however, we should always aim for a lower WER, as it can potentially save lives.

Mistakes in Air Traffic Control ATC communication. Models for automatic speech recognition ASR exist, but constant improvement is needed. ASR models should minimize errors, which is still massively in progress. Minimizing errors in communication will reduce the risk of accidents.   
Models of ASR are being designed and updated constantly, which leads to great innovation in this field. An example of a community that specializes in Wav2Vec2 transformer models and data sharing is HuggingFace. Here, individuals can create, modify and publish their own models and datasets for others to use. Big companies such as Meta have shared their models as well, which are trained on massive amounts of data. These models are very recent and have been tested on ATC inadequately. (Has been done before though). In this paper, Wav2Vec2 transformer models are evaluated on their performance on transcribing ATC communication from the atcosim dataset. The models that are considered contain:

* A model that was pretrained on LibriSpeech data (Base)
* A model that was pretrained and fine-tuned on multiple noisy datasets as well as clean data (Robust)
* The finetuned XLS-R model in addition to a language model
* (Finetuned model on ATC data)

Technical ASR modifications on the models were not researched, as it was not the aim of paper.

**Research questions:**

* How robust are pretrained and fine-tuned automatic speech recognition models on ATC data?
* How does a language model affect the performance of a finetuned XLS-R model on ATC data?
* How does the amount of data used to finetune affect performance of a finetuned XLS-R model on ATC data?

# Preliminaries

This is an optional chapter that contains knowledge that your reader needs to know in order to understand your work. If there is not much you can also embed the preliminaries in one of the other chapters or even discard it completely.

* Transformer models
* WER and CER -> why not other metrics, like Glue or Bleu
* Huggingface

# Related work

Within this chapter you show that you have sufficient knowledge of the problem domain.

* Other models of ATC and their performance
* Wav2Vec2 Model and their performance
* Attention is all you need paper
* XLS-R on accents

# Methodology

* HuggingFace – very recent models, supposedly easy to use/learn, course, documentation
* Development - Colab and local
* ATCOSIM – choice of dataset, choice of pruning/cleaning dataset, augmentation dataset, split dataset
* Choice of models – base, robust, Hubert, xls-r
* **Transcribing**
* **Evaluation**
* **Training** **(Fine-tuning)** – hyperparameters, use of HuggingFace
* **Addition** **Language** **model** – 5-grams, low word count

**HuggingFace**  
Numerous python libraries exist for the development of ASR, such as scikit-learn and NLTK. These are standalone packages that are highly useful for tuning hyperparameters and developing NLP. For this research, complex development is not needed, rather testing of existing models is the focus. Testing of new and popular models can be more difficult to implement and test in these environments, as they are not easily accessible yet.   
The platform HuggingFace (HF) is more accessible. The HF community provides a wide range of publicly accessible models and datasets that can be modified with ease and reuploaded to the HF hub for public use. Over 20 transformer tasks are available for use, e.g. ASR, image classification and text generation. Models and datasets are constantly updated, which implies that state-of-the-art tools are attainable. HF has a course for beginners and a decent documentation of the main libraries used: Datasets and Transformers. The learning curve is high, but the models and data delivered outweigh the costs compared to other libraries.

**Development**Different platforms are used for the development of transcribing, evaluating, training and testing of ASR models. HF recommends using Google Colab for ease of programming and the absence of owning a high-end GPU for computation. For low computational power demanding tasks, such as small batch transcription, Google Colab is used, but the free to use platform has limitations that are diminishing our resources. GPUs can be either T4, P100 or a K80. These GPUs are assigned by Google Colab and are unpredictable. Fine-tuning models and transcribing full datasets regularly exceed the resources (GPU ram, disk space and system ram) made available by Google Colab, leading to resource deprivation.   
For this reason, local development in scripts is done in addition to cloud development. A Geforce GTX 1080 is used for transcribing full datasets and training of models for more robust development.

**Dataset**  
There is a distinct shortage of publicly available data for ATC that can be used for ASR, as most datasets do not include transcriptions. Websites such as ‘*liveatc.net*’ provide, as the domain name implies, live and very recent ATC communication audio. Based on airport codes and frequencies, different radio communication is selected and listened to. This data does not include transcriptions and is therefore not suitable for training our models. Audio data is abundant and publicly available; however, audio data has to be manually transcribed, which is costly and time-consuming, leading to lower accessibility of transcribed ATC data.   
In this research, the ATCOSIM dataset is used. This corpus provides ten hours of speech data, pronounced by ten non-native speakers in the English language. It includes additional information on the recording sessions and speakers, such as: Gender, speaker id, recording length, etc. This information was not used/ researched as it is out of the scope of this paper.  
The ATCOSIM dataset is a simulation of real-time ATC, which is less representative of real ATC. The same ATC lingo is used in the 10-hours of speech data. The data is considered clean, as speech recordings were made in a controlled environment (Hofbauer et al., 2018). Nonetheless, the data is noisy, as a close-talk headset microphone is used. Speaking style, language use, background noise and stress levels were kept as realistic as possible, but the representability to real ATC data has not been studied yet. This should be taken into consideration when evaluating the generalisability of the studied ASR models.

*Show dataset* **– not on HF**

The ATCOSIM dataset can be downloaded from the internet and can be obtained on DVD. The corpus is formatted in ISO, which needs to be reformatted for HuggingFace transformers use. As mentioned above, additional information on speakers and recordings are removed from the dataset and will not be used.  
Additionally, unusable rows in the dataset are removed. Unusable meaning:

* Rows containing transcriptions that exceed the maximal duration allowed for training and model transcription, as GPU usage is a limitation in this research.

*split dataset*

**Models**base, robust, Hubert, xls-r

**Transcribing**

**Evaluation**

**Fine-tuning**what is fine-tuning, hyperparameters, use of HuggingFace

**Addition** **Language** **model**5-grams, low word count

# Experiments

## Data

## Settings

# Results

* Performance of pretrained models
* Performance of finetuned models
* Performance of finetuned model with language model

Pretrained robust model performs a bit better than base model. Reason for this is:

Evaluate errors

# Conclusions

In this chapter you present all conclusions that can be drawn from the preceding chapters.

It should not introduce any new material or theories; these should have been written down earlier in the thesis. Because of this, your conclusion can be brief and to the point.

When further research?

* Fine tune on different models, was not the focus of this paper
* Try finetuned models on more data for comparison and generalisation factor
* Finetune on more data or on different training settings, as hyperparameters have not been tuned for our ATC data.
* Add LM to lower finetuned models to see the performance difference

# Bibliography

# Appendix

Appendices are optional chapters in which additional material is covered. This material is required to fully support your research, that would otherwise clutter the presentation of your research.