An evaluation and analysis fine-tuned representations for code-switched speech recognition in a zero-resource scenario

B171923

Word Count: 7839

Master of Science

Speech and Language Processing

School of Philosophy, Psychology & Language Sciences

University of Edinburgh

2022

Abstract

Recognising code-switched speech (alternating between two or more languages or varieties of language across sentences in conversation) is an important technical and social issue essential for modern society. The majority current speech recognisers are trained monolingually and therefore do not perform well on such utterances. The use of Deep Neural Network (DNN) architectures to train models allow for shared representations and provide an opportunity to level them to better handle code-switching. In the two studies contained in this work, we show multilingual fine-tuning of self-supervised speech representations can handle code-switching in a zero-resource scenario and through analysis of the latent representations, that code-switching is encoded in the model.

Keywords - Automatic Speech Recognition, Code-switching, wav2vec 2.0 XLSR, Probing

Acknowledgements

Thank you, God, for giving me the ability to do this dissertation. Thank you to Nay San, Seth Aycock and Dan Jurafsky for being so helpful!

Table of Contents

Chapter 1

Introduction

Over half of the world's population uses at least two languages regularly [2]. Two entirely natural consequences of multilingualism are code mixing (inserting a single word or phrase from another language into a sentence) and code switching (alternating between two or more languages or varieties of language across sentences in conversation). In this work, code-switching is defined by coaching at a phrase or sentence boundary and code mixing also known as inter-sentential code switching is described as utterances where more than one switch between languages takes place. When discussing both in a general sense, code-switching is used.

Yet, most automatic speech recognition (ASR) models for transribing speech waveforms are trained monolingually, meaning that they do not have any linguistic or phonetic intuition on different languages spoken, leading to very inaccurate transcriptions.

Various multilingual or shared representation models have been trained and tested to combat the code-switching phenomenon. Recently wav2vec 2.0 [3], an unsupervised pre-training of speech has proven to give very low word error rates and character error rates for English transcriptions. Although very costly to pre-train, they have released cross-lingual representations [7] that can be fine-tuned to efficiently make speech recognisers in a variety of languages.

The goal of this work is split into two studies. The first asks: *"What is the best way to build recognisers that are robust to code-switched data?"*, with our main goal being to improve recognition accuracy of code-switched data. We propose a set of experiments which include building several models with various training data combinations and analyses which setup gives the most accurate transcriptions. The second study asks: *"Can we explore the internal workings of the models to understand how they work?"*. Here, we do layer-wise analysis and run probing classifiers on monolingual

and code-switched test data to gain insight on the models' failures and successes by what we discover about the architectural properties.

To test this phenomenon, Yoruba and Nigerian accented English are chosen to study the effectiveness of multilingual models in situations such as code switching or code mixing. Most spoken Yoruba is code-switched with English and therefore systems built to recognise the speech of a Yoruba speaker must incorporate knowledge of English pronunciation of Yoruba speakers, hence the use of Nigerian Accented English as opposed to other varieties of English.

Utterance	English translation	Code-switching or mixing
(a) Mo n lọ si training	I'm going to training	Code-switching
(b) Education system ilu yi o da rara it's too bad	This country's education system is not good at all, it's too bad	Code-mixing
(c) Dada ṣe kini? Dada ni accident?	Dada did what? Dada had an accient?	Code-mixing (1 word mix)

Table 1.1: Annotated examples of code-switching and code-mixing along with English translations

Table 1.1 displays some translated instances of code-switching between Yoruba and English. Utterance (a) starts off in Yoruba and switches to English at the end. Utterance (b) has several two switches: one at "... system ilu ..." and another at "... rara it's ...". Here the phrase ilu yi o da rara (EN: this country is not good at all) is inserted into the sentence, producing multiple switches and therefore is code mixing. Code mixing often occurs with a noun or adjective only. This is the case in utterance (c) where the only switched word is "accident".

After analysing the transcriptions, we look into the model and contribute to existing work done on English identifying whether linguistic and acoustic information are included in the transformer layers of the model. We also run several probes on the model to determine whether or not it can detect code-switching.

We discover that the bilingual model without additional code-switched data give the highest transcription on the held-out test set and that although not as clearly encoded as English models, there is evidence of acoustic and phonetic information in the transformer layers. The middle transformer layers (12-15) can detect code-switching to an accuracy of 90% on average across probes.

The work begins with background on the various models used and analysis of models, introduces the data and models used in this work and then details the methods evaluation and results of study 1 the effect of training data on transcription and study to analysis of wav2vec models respectively.

Chapter 2

Background

Low-resource and zero-resource ASR have become increasingly prevalent in recent years due to novel approaches to end-to-end ASR (covered in Section 2.1). These methods, along with some adaptations for code-switching, have led to the exploration of code-switching ASR in a wide variety of language combinations and contexts (covered in Section 2.2). Deep neural nets are less interpretable than models using previous approaches to ASR, so work has been carried out to analyse layers of various DNN architectures (Section 2.3) and the networks have been probed for various tasks secondary to transcription of audio (Section 2.4).

2.1 Recent Advancements in End-to-End Automatic Speech Recognition

The rise in the use of deep neural networks (DNNs) and encoder-decoder architectures in machine learning has revolutionised novel approaches of end-to-end ASR. In contrast to the traditional Gaussian Mixture Model-Hidden Markov Model approach to transcribing audio in English and later other languages, DNNs use their parameters more efficiently, with each parameter applying to a larger proportion of data than a GMM model [14].

End-to-end ASR with DNNs can be split into two approaches: Connectionist Temporal Classification (CTC) and encoder-decoder models.

CTC models, proposed by Graves et al. predict characters from short segments of audio without the need of frame-level alignments in training using the CTC algorithm [11]. They produce monotonic, many-to-one, alignments to the input. The CTC

algorithm produces conditionally independent output, meaning that in an ideal case, language modelling is required to ensure that the outputs produced exist in the target language. The CTC criterion is usually used to predict output of deep bi-directional LSTM or RNN models via the objective of minimising CTC loss, applied in architectures such as Deep Speech [12] or attached to the decoder of encoder-decoder models.

The encoder-decoder architecture usually consists of two models: taking in the input and creating abstract vector representations of it, and the second taking in the abstract vector representations to predict the output. Examples include Google's Listen, Attend and Spell [4]. In this case, encoder-decoder models take in raw wave forms or acoustic features such as Mel filter bank coefficients or MFCCs, and predict either text or words in a lexicon. Most modern encoder-decoder models used for automatic speech recognition incorporate attention used in the decoder to incorporate information on surrounding input to better predict the output.

The prominence of transformer based, self-supervised representations in Natural Language Processing led to their applications to ASR. Architectures such as Mockingjay [18], BERTPhone [17] or DeoCAR 2.0 [16] use transformers to encode latent speech representations of audio. Such models can have CTC loss heads or other decoding methods attached to the end of them to produce transcriptions. This work narrows the evaluation and analysis to one of such models: wav2vec 2.0. The scope is narrowed to allow for in-depth analysis of model architecture to be directly tied to the experimental results.

Wav2vec 2.0 is a neural architecture that uses unsupervised pre-training to create abstract representations of speech that has state-of-the-art accuracy for speech recognition of English audio [3]. Facebook Artificial Intelligence Research (FAIR) have also released a pre-trained model with training data from 53 different languages to produce cross-lingual representations (wav2vec XLSR 53 [7]). Details on these models are found in Chapter 3. Most experiments and reported results for such models are on English data, but models have been fine-tuned for various low-resource languages, however not multilingually.

2.2 Computational Approaches to linguistic code-switching

In speech processing, work on code-switching can be divided into code-switching detection [23, 28, 26] using language identification [6] and end-to-end recognition [15]. End-to-end recognition splits into two main approaches - a multilingual modelling with

cross lingual representations and parallel modelling generating multiple transcriptions which are interpolated to result in one transcription with the highest likelihood.

In scenarios where high-quality data can be scraped from the web or other resources, language modelling is used, significantly improving performance across methods. Given that classification begins on the phonetic level, experimentation on whether shared phone sets, language-specific phone sets or newly created accent-specific phone sets improve performance.

With the rise in the use of Deep Neural Network architectures [29], cross-lingual abstract representations and transfer learning [27] have taken precedence to capture acoustic representations across languages. Before the introduction of self-supervised representations like wav2vec 2.0 [3], encoder-decoder architectures have been employed for end-to-end code-switched speech recognition across language pairs.

Models have been built to tackle domain-specific code-switching, such as in education or banking. In the same vein, many code-switching datasets are domain-specific due to its inconsistent appearance in speech and language.

To measure the amount of code-switching in an utterance, two metrics have been used: the Code Mixing Index (CMI) and Complexity Factor (detailed in Chapter 3). These allow for comparison across language pairs and datasets, providing additional insight to model performance.

In this work, we mimic a zero-resource ASR scenario, with no language modelling, pronunciation lexicons or parallel transcriptions. We use a multilingual model and CTC-predictions only.

2.3 Layer-wise analysis of abstract speech representations

The prominence of encoder-decoder architecture in modern deep neural networks has led to a field of work analysing the abstract representations. Depending on the architecture, several features can be extracted and compared to existing representations or actual representations of input or output to the model. In the case of wave2vec 2.0, Pasad et al. [21] conducted layer-wise analysis of its transformer layers. The authors study both the BASE and LARGE versions (detailed in Chapter 3.2)) of the English wav2vec 2.0 model and compare each transformer layer to linguistic representations in the form of GloVe embeddings [22], phonetic information via comparisons to phone-

level Mel filter bank features, layer-wise comparisons between representations of the BASE and LARGE model and the effect of fine-tuning models on these findings. Pasad et al. discover that irrespective of the size of the model, layer-wise representations begin to encode linguistic or phonetic information from the middle layer onwards (Layer 6 in BASE and Layer 12 in LARGE) and see to have discontinuities in the final layers. The authors conclude that the final layers are not optimal for fine-tuning and suggest performing fine-tuning to continue the training objectives with a layer lower than the final layer. Given the relative novelty of wav2vec 2.0 in comparison to other ASR models, all the work so far has been done with English, on the wav2vec 2.0 representations. In the work, we study not only another language, but a low-resource unseen language. Any similarities in findings will indicate general features of the model architecture.

2.4 Probing Self-supervised speech representations

Due to the various abstract representations in hidden layers of neural networks, work has been done to extract layers and ”probe” them - use various classifiers on them for specific tasks [1]. In ASR, tasks range from speaker detection, phoneme detection, vowel or fricative detection [19] to fluency, pronunciation, semantic and syntactic level features [24] with experiments carried out comparing results for native English speakers and non-native English speakers, but there is yet to be a comparison of such findings with the resulting fine-tuned representations of other languages. Such probes give relative confidence in a layer's ability to perform a certain task to a high degree of accuracy. Various classifiers can be chosen as probes such as Support Vector Machines (SVMs), Logistic Regression, or small neural networks. It has been shown that probing is sensitive to the classification task [13] and complexity, so most work report accuracy for a variety of classification tasks to increase confidence in conclusions. In a multilingual setting, probing for language detection is something that has yet to be explored with wav2vec 2.0.

Chapter 3

Data and Models

The majority of the data used to train the models used in monolingual in the respective languages, given that there are few code-switched corpora large enough to effectively train ASR models, let alone models for low-resource languages. Below is a detailed summary of the monolingual datasets used for the initial experiments (3.1.1), a description of the code-switched dataset created for this work 3.1.2 and an introduction to the pre-trained models used in this work (3.2) .

3.1 Data

Monolingual data in Yorùbá and Nigerian accented English were used to fine-tune these models. A dataset of 350 code-mixed and code-switched instances was handcrafted to provide a test set for this work. A summary of the monolingual datasets is illustrated in Table 3.1 and the code-switched dataset in 3.2.

3.1.1 Monolingual Datasets

Yorùbá is a language spoken by 43 million people worldwide, 2 million speaking it as a second language [8]. It is also the most broadly spoken African language outside Africa. Yorùbá is a tonal language with three tones: low, mid and high tones, donated by presence or absence tonal marks on vowels in text. The alphabet uses Latin characters with to give the alphabet: *a b d e ẹ f g gb h i j k l m n o ọ p r s ṣ t u w y* with, in the scope of this work, Yorùbá-specifc voiced labial–velar plosives gb [b] and voiceless labial–velar plosive p [kp]. The ṣ is the orthographic representation of the /ʃ/ phoneme, or the 'sh' in 'shoe', 'shirt', or 'fish'. There are seven vowels in Yorùbá, *a e*

ẹ i o ọ u with ẹ specifically referring to the e at the start of 'escalate' or 'embody' and ọ specifically referring to the o at the beginning of 'operation' or 'oscillate'. The *n* is also used to nasalise vowels.

Despite the sheer number of speakers, it is still a low-resource language, lacking enough data to train the language models that are used today for downstream tasks. There are no widely available, robust ASR systems in existence for Yorùbá. Small datasets have been created for research purposes, including the Lagos NWU corpus.The Lagos NWU corpus is a speech corpus made specifically for speech research. 33 speakers (16 female, 17 male) each recorded 130 utterances selected for phonetic coverage. Due to the main aim of the corpus being phonetic coverage, utterances are short and cover a wide variety of domains and topics. All speech is read aloud by speakers into a microphone that is connected to a laptop in a quiet environment.

Due to the range of ethnic groups, subcultures and languages spoken in Nigeria, there is a variety of English spoken in Nigeria, with phonology similar to the other languages spoken by Nigerians - Nigerian Accented English [20, 25]. When code-switching, a speaker is likely to have an accent in one or more of the languages spoken [5] and in this scenario, native Yorùbá speakers are more likely to switch to Nigerian Accented English as opposed to Received Pronunciation or any other variant of British English. To better model this code-switching scenario, a low-resource but more accurate Nigerian Accented English dataset is used.

The Nigerian Accented English dataset [10] released by Google consists of 3,350 sentences read aloud by varying native Nigerian volunteers. The sentences cover a wide variety of domains.

Datasets		
Language	Dataset	Total Hours
Nigerian accented English	Nigerian English multi-speaker speech dataset	5
Yorùbá	Lagos NWU Corpus	5
NG CS Dataset	Hand-crafted	0.5

Table 3.1: Summary of monolingual datasets used in experiments

3.1.2 Code-Switched Dataset

The Nigerian-CS dataset is a dataset prepared for this work consisting of 350 code-mixed and code-switched utterances lifted from Nollywood films, News Bulletins and political speaking engagements. Across data sources, a variety of domains are covered, but the types of speech are different: the Nollywood utterances consist of rehearsed speech, News bulletins have read aloud speech (speech similar to most speech in the datasets used to train wav2vec 2 XLSR) and the political speaking engagement speech consists of conversational speech, speech most likely to be transcribed if such a model were embedded into commercial systems.

3.1.2.1 Code Mixing Index (CMI)

The Code Mixing Index (CMI) is a metric introduced by Gambäck and Das [9] to measure the complexity of code-mixed utterances. Equation 3.1 outlines the calculation of the CMI of a text-only utterance.

$$CMI = \begin{cases} 100 \times 1 - \left[\frac{max(w_i)}{\sum_1^N w_i}\right] & \text{if } n > u \\ 0 & \text{if } n = u \end{cases} \tag{3.1}$$

Where w_i is the total number of text tokens in an utterance and $max(w_i)$ is the number of words in the dominant language of the utterance. $\sum_1^N w_i$ is also expressed as $n-u$, where n is the total number of tokens in an utterance and u is the number of language independent tokens, such as punctuation.

In the utterance "Lo tọ ni pe two heads are better than one" taken from the Nigerian-CS dataset, the CMI is 40. The utterance starts with four Yorùbá words then switches to an English saying containing 6 words. Therefore, $max(w_i) = 6$ and $\sum_1^N w_i = 10$.

$$CMI = 100 \times 1 - \left[\frac{6}{10}\right] = 100 \times (1 - 0.6) = 40$$

This metric can be measured over individual utterances, paragraphs or entire datasets as used in the NLP literature. In this work we focus on utterance level transcriptions and therefore report average utterance level CMI. CMI provides extra context to the performance of models on corpora with varying degrees of code-mixing.

Nigerian-CS Dataset	
Utterances	350
Average utterance length	6
Average CMI	25.9

Table 3.2: Summary of code-switched datasets used in experiments

3.2 Models

The multilingual wav2vec 2.0 model (XLSR-53) was fine-tuned for experiments in this work, given the resource intensiveness of training a wav2vec 2.0 model from scratch (e.g. 128 VGPUs over 5.2 days).

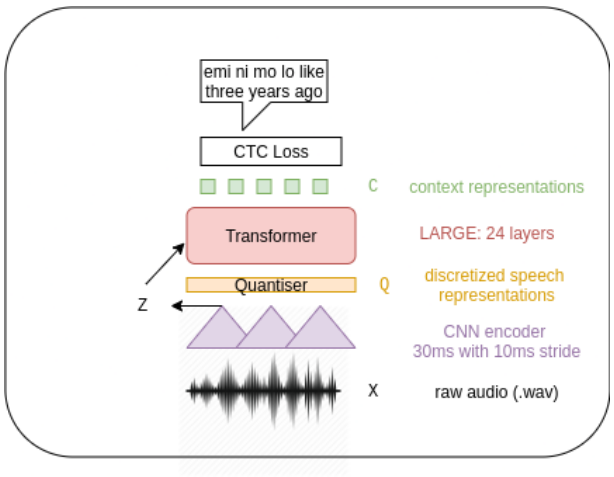

Figure 3.1: Visual representation of the wav2vec 2.0 architecture

3.2.1 wav2vec 2.0

wav2vec 2.0 is a model architecture introduced by Facebook Artificial Intelligence Research (FAIR) that learns abstract speech representations from raw waveforms. This is in stark contrast to existing HMM-DNN models that use preconstructed features e.g. Mel filter bank coefficients to transcribe audio. This architecture also so does not require a language model to provide transcriptions. wav2vec 2.0 can be split into a three-step process. It starts by obtaining a latent speech representation of the waveform with convolutional neural network (CNN) layers (Z). The new latent representations are fed into a transformer, but are quantised beforehand (Q) for use as input to other machine learning methods. The initial latent speech representations (Z) are then masked, allow-

ing the transformer to maximise the percentage of masked representations predicted correctly, resulting in the final context representations (C). Figure 3.1 provides a visualisation of the wav2vec 2.0 architecture. For the English models, FAIR released two versions of wav2vec 2.0; wav2vec 2.0 BASE with 12 transformer layers and wav2vec 2.0 LARGE with 24 transformer layers. Similar to existing hybrid (HMM-DNN) systems, the representations can be classified as characters using CTC loss. The wav2vec 2.0 models have been able to achieve a very low word error rate when transcribing English (5% on the LibriSpeech benchmark dataset).

3.2.2 wav2vec XLSR

FAIR recently released a pre-trained model optimised for cross-lingual speech representations that was trained on 53 languages across three different open source speech datasets. This model has been selected for fine-tuning because it is more likely to produce increasingly accurate speech representations for unseen low-resource African languages. This is due to the wider variety of phonological features present in the set of language used in pre-training of wav2vec XLSR 53 which has incorporated languages from almost every continent in the world. The varying phonological aspects of languages such as tone (Cantonese), other places of articulation, and articulatory combinations are present in this model, but would not be present in a model pre-trained solely on English.

Chapter 4

Study 1: Improving transcription accuracy

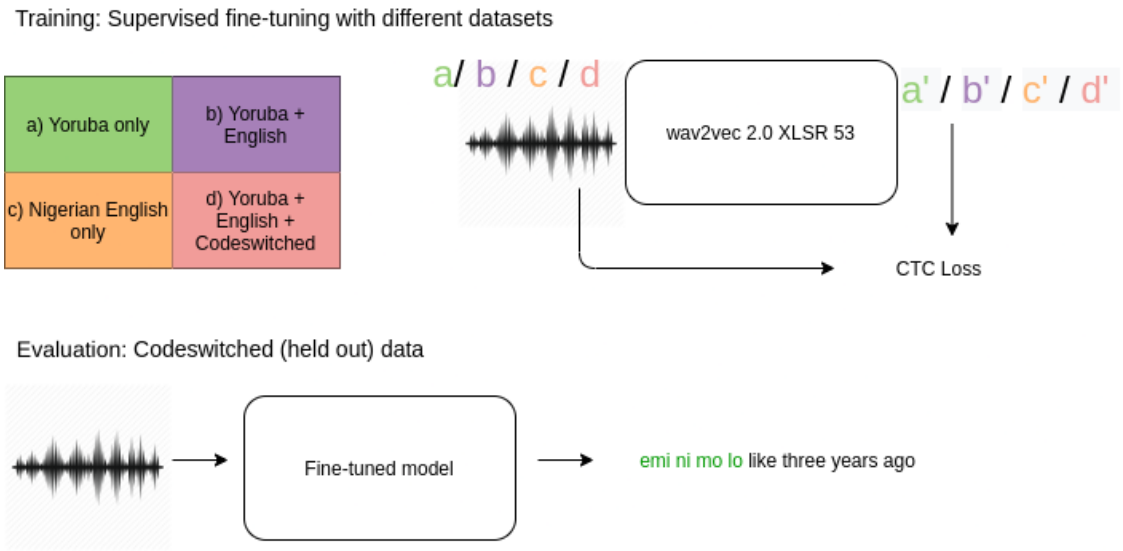

Figure 4.1: Visual representation of Study 1

4.1 Method

This study consists of fine-tuning the wav2vec 2.0 XLSR model using a variety of training datasets, keeping hyperparamters constant. Below are the details of the baseline models and multilingual models trained and an introduction to the evaluation used to compare models in this study.

4.1.1 Baseline models

To evaluate whether multilingual models lead to an improvement in transcription accuracy for code-switched utterances, two monolingual models were implemented as baseline models, one in Nigerian Accented English (NG) using the Crowdsourced Nigerian English dataset and one in Yorùbá (YO) using the Lagos NWU Corpus. Code-switched and code-mixed utterances were transcribed by the models indicating whether monolingual models can handle code-switching.

4.1.2 Multilingual model

The novel approach in this work is to fine-tune wav2vec 2.0 XLSR on multiple languages in order to improve transcription accuracy for code-switched utterances. The monolingual data for Nigerian English and Yorùbá will be combined to make the fine-tuning data for a resulting multilingual model. In an atttempt to further improve performance, 10 minutes of code-switched data are including to the monolingual fine-tuning data both at the initial fine-tuning time and after fine-tuning to obtain a bilingual model. This suite of models will form the set of models to be interpreted in later sections.

4.2 Evaluation

Transcriptions will be produced by minimising CTC loss based on the input to the model. As the transcription method is data agnostic, different fine-tuning data variations result in different model weights and therefore, varying transcription accuracy.

To measure transcription accuracy, the word error rates and character error rates will be evaluated. The word error rate and character error rates are demonstrated in Equations 4.1 and 4.2, where:

- S is the number of substitutions in the predicted transcription
- D is the number of deletions in the predicted transcription
- I is the number of insertions in the predicted transcription
- C is the number of correct words in the predicted transcription
- N is the number of words in the gold transcription

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \tag{4.1}$$

$$CER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \tag{4.2}$$

4.3 Experiments

4.3.1 The effect of training data and training procedure on transcription accuracy

As many deep learning approaches rely heavily on data, several models are fine-tuned with different amounts and types of data to investigate the effect of training data on the resulting models, irrespective of transcription method.

To investigate this, five different models are fine-tuned as illustrated in Table 4.1. Initially, monolingual Yorùbá and Nigerian Accented English models are built by fine-tuning the wav2vec 2.0 XLSR endpoint provided by fairseq, resulting in standard monolingual models (a) and (b). Since the investigation is of code-switched data, model (c), a bilingual model is trained with the combined training set of both the monolingual datasets.

Although model (c) should be able to handle code-switched data, the model seeing code-switched utterances at training time could improve transcription accuracy. 10 minutes of code-switched utterances is added to training data at the same time as the monolingual data in model (d) and after a bilingual model is trained in model (e), by fine-tuning the checkpoint of model (c).

Given the small amount of linguistically similar training data, fine-tuning the bilingual model to slightly alter the representations (model (e)) should provide the best transcription accuracy.

The results of the experiment are in Table 4.2. Unsurprisingly, the Nigerian English baseline model does not manage to transcribe the code-switched test set with a WER of 1.0 and CER of 3.8. This is expected behaviour as most Yorùbá speakers switch out of Yorùbá into English. The Yorùbá monolingual model provides a good baseline with a WER of 0.96 and a CER of 0.61. Here we can see the effect of not introducing a language model or lexicon to the architecture. The character-level prediction due to CTC loss combined with a lack of language models means that the predictions are unlikely to be words in the lexicon, let alone Yorùbá or English. We see that on a

Model	Training Data	Type
(a) Monolingual YO baseline	YO	Monolingual
(b) Monolingual NG baseline	NG	Monolingual
(c) Multilingual (YONG)	YO, NG	Multilingual
(d) Multilingual CS	YO, NG, CS	Multilingual
(e) Multilingual CS-finetuned	YO, NG + CS	Multilingual

Table 4.1: Summary of models trained to investigate the effect of training data on WER

Model	WER	CER
(a) Monolingual YO baseline	0.96	0.61
(b) Monolingual NG baseline	1.0	3.8
(c) Multilingual	0.92	0.52
(d) Multilingual CS	0.94	0.75
(e) Multilingual CS-fine-tuned	0.94	0.75

Table 4.2: Average WER and CER of models trained

character level, predictions comparable to WERs of ASR models for low-resource languages that include a language model.

Surprisingly, the bilingual model outperforms models that include code-switched data in training, having the lowest WER and CER. This may be due to the nature of the data used to train and the introduction of lower quality data negatively affecting the weights, worsening predictions. As the bilingual model is the best performing model, we will compare YONG to monolingual baselines in further experiments.

4.3.2 The effect of domain disparity on transcription accuracy

The code-switched training and test data was collected from varying domains and contains conversational speech, read-aloud speech and rehearsed speech. The monolingual datasets, providing the majority of the training data consist only of read-aloud speech. The test recordings most similar to the training data (read aloud speech) should have the highest transcription accuracy.

Table 4.3 shoes the mean WER and CER of code-switched utterances classified by domain. Read aloud data has the lowest WER and CER as suspected, but surprisingly, conversational data was easier for the model to transcribe than rehearsed data. This may be due to the rehearsed data's selection from films, adding some background

Domain	WER	CER
Read aloud	0.92	0.42
Conversational	1.08	0.70
Rehearsed	0.97	1.09

Table 4.3: Summary of models trained to investigate the effect of training data on WER

noise to the clips in some instances.

4.3.3 The effect of the code-mixing index (CMI) on the transcription accuracy

The multilingual model can transcribe monolingual utterances with the same accuracy as monolingual models. This suggests that if there is minimal code-mixing, the transcription accuracy may be higher as the utterance is more similar to a monolingual utterance than one with a lot of mixing.

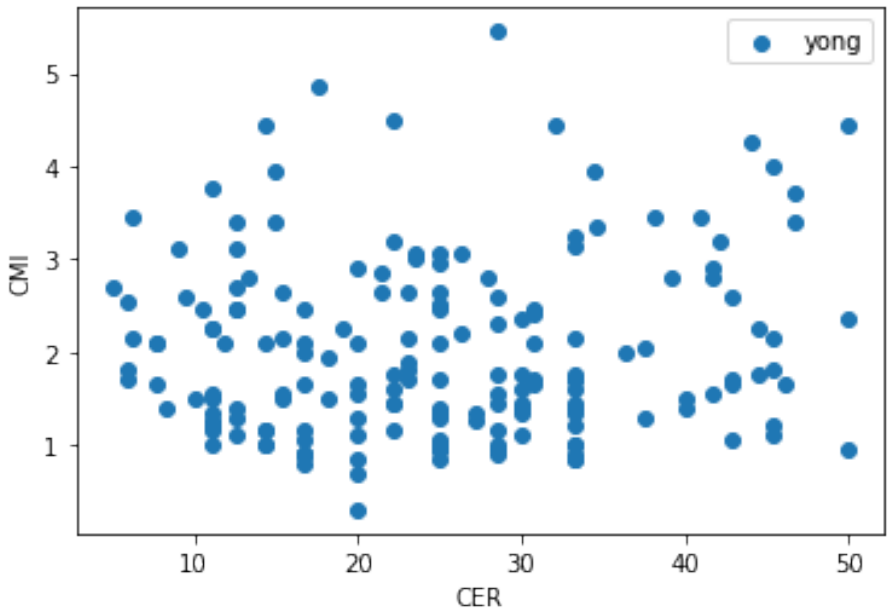

Figure 4.2: CMI of utterance against CER of transcription

The scatter plot in Figure 4.2 shows that there is no correlation between the code mixing index and the transcription accuracy of the data in the multilingual case. This is probably due to the training objectives of the model being character specific and not language-specific, therefore it is not essential for the model to capture specific

languages with such little training data. As the model does not discriminate between Yoruba and English, the degree of confusion in transcription is non-existent.

4.3.4 Error analysis

4.3.4.1 Code-mixing vs Code-switching

The test data contains code-mixed and code-switched data, sometimes in the same utterance. It is possible that such models can better transcribe code switched data as due to the local uniformity of the language and phone set. This also suggests that transcription at boundaries in both code-switching and code-mixing cases will be less accurate leading to a lower transcription accuracy.

Utterance type	WER	CER
Code-mixing	1.22	1.66
Code-switching	0.95	1.29
Both	1.42	1.50

Table 4.4: Transcription accuracy of different types of switching

The results in Table 4.4 show that code switching provided lower error than mixing. This is expected as the neighbourhood of characters in the same language is likely to be higher and this information is likely to be included in the model as the training data did not contain code-switched data and therefore correlated certain characters attributed to Yoruba or English to each other. Code mixing especially when it occurs more than once can be difficult to correctly transcribe especially in the zero resource scenario where there is no language modelling or supervised training data of the phenomenon when both occur at the same time (an utterance with two sentences one of which contains more than one switch) that gives a marked increase on the word and character error rates showing that both linguistically, and computationally it is the most difficult situation to transcribe.

4.4 The effect of shared phonetic representations

We see in Table 4.5 that the YONG model made impressive attempts at transcribing code-switched utterances in an unseen, zero-resource scenario. We can see the effects of shared representations with transcriptions such as ”ṣhe” for she in (a), where the

Gold transcription	YONG Transcription
(a) she's a changed person so ẹ need lati mock ẹ	ṣhe is a chage tos toẹniet lati mọrki
(b) mo jẹ ẹfo ti america	mo je ko ti a mẹrika
(c) emi ni mo lo like three years ago	a minimo low like three years agoou
(d) telling me e ma bother ara yin	ṣhalime e ma bother arayin
(e) you can be rest assured pe ma mun ikun wa fun yin	n you can be rest asured bamamukowafuni

Table 4.5: Transcription examples from the YONG model

model has clearly learnt the /ʃ/ phoneme and favours Yorùbá transcription, a good sign for low-resource languages. The same occurs in (b) where the "e" in America is correctly transcribed to the Yorùbá orthographic transcription. We also see in examples (c)-(e) that English words and phrases embedded into utterances can still be located and transcribed correctly. If not transcribed correctly, the errors are comparable to those of wav2vec [3]. These examples show that character level learning is taking place well. The aid of a language model will reduce such transcription errors, but experiments without a language model highlight that phonemes are learnt to some extent.

Chapter 5

Study 2: Looking at model representations

5.1 Method

5.1.1 Layer-wise analysis

As the wav2vec 2.0 architecture has 24 transformer layers, it is possible that certain transformer layers correspond to acoustic or linguistic features of the utterances. The layers will be extracted by pretrained models and compared to traditional acoustic features (MFCCs) and word embeddings. To compare the different types of vectors, Canonical Correlation Analysis (CCA), detailed in 5.2.1 will be measured.

5.1.2 Probing

To investigate whether the model can detect code-switching, pre-trained multilingual model will be probed by measuring its performance on the binary classification task of code-mixing detection layer by layer. Features will be extracted from each layer after passing through waveforms from the monolingual test sets (NG and YO) and the code-switching test set, with label 0 for monolingual waveforms and label 1 for code-switched waveforms. Due to probing tasks' sensitivity to their classifier, both Logistic Regression and SVM will be run on the extracted features.

5.2 Evaluation

5.2.1 Vector comparison: Canonical Correlation Analysis (CCA)

Canonical Correlation Analysis (CCA) measures the largest possible correlation that can be found between the linear combinations of two sets. For consistency with other work on analysing wav2vec layers [21], we use the projection-weighted CCA, but refer to it as CCA form now on.

In this case, it is used to compare two vectorised representations of the same thing: utterances and words. Word embeddings have successfully captured various textual relationships, and MFCCs have been used to model human perception of speech for decades. Comparing these well-tested representations to the transformer layers of wav2vec 2.0 can provide an indication to what the transformer layers represent.

5.2.2 Probing the representations: Classification tasks

The multilingual model is probed using simple classification tasks on each layer. Due to reported sensitivity of probing results to their methods, four methods are employed to increase confidence in conclusions: an SVM, Logistic Regression, a neural network with one hidden layer (200 units) a neural network with two hidden layers (L1: 200 units, L2: 100 units).

5.3 Experiments

5.3.1 Does multilingual training affect the resulting representations?

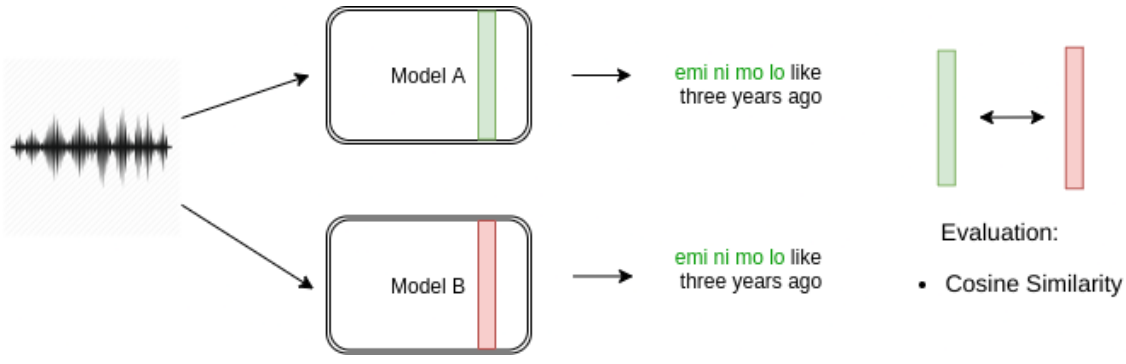

Figure 5.1: Visual representation of the experiment, titled *Does multilingual training affect the resulting representations?*

For monolingual test data, transcription accuracy is of the monolingual (YO or NG) and multilingual (YONG) model is identical. However, due to the difference in fine-tuning, the model weights are likely to differ. To investigate this, we measure the cosine similarity and CCA of the transformer layers of the respective models. The cosine similarity comparison of the model weights to each other. A score of less than 1 will confirm that encodings of monolingual test utterances are different.

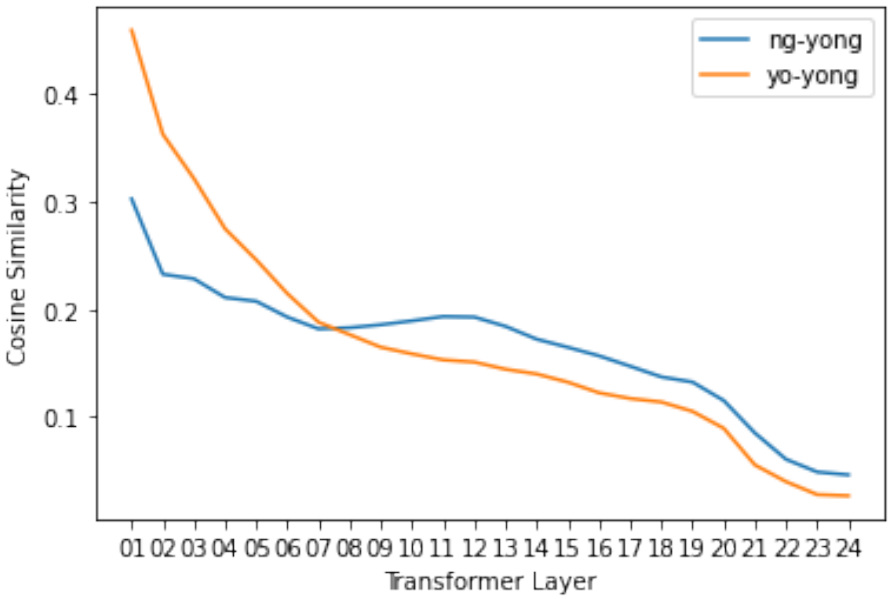

Figure 5.2: Cosine similarity of YO-YONG and NG-YONG model weights

The cosine similarity of the layers of the YO and NG models compared to the YONG model are shown in Figure 5.2. It is clear that the similarity of the monolingual and multilingual models decrease as the depth of the model increases, which is expected. The initial similarity is below 50% for both models, meaning that the way in which the weights are set is completely different, making the multilingual representations distinct from the monolingual representations. Although the earlier layers of the YO model are more similar than NG, this changes at between layers 7 and 8, with the later layers of NG being more similar to YONG than to YO. That being said, the difference is not as marked as the initial difference between YO-YONG and NG-YONG. Similarity dips in the final 3 layers for both comparisons. This is expected as those layers are used for the masked prediction task, a task completely different to transcription.

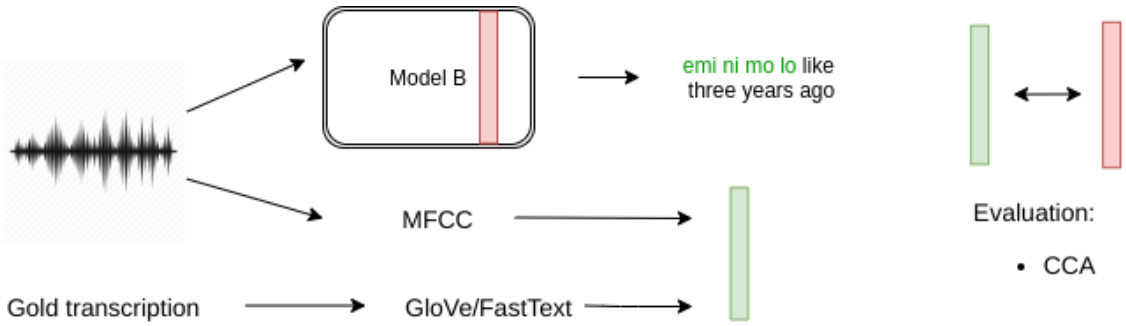

Figure 5.3: Visual representation of experiments titled *Is acoustic information encoded?* and *Is linguistic information encoded?*

5.3.2 Is acoustic information encoded?

To discover whether acoustic information is encoded in the models, we compare both the quantiser and transformer layer representations to MFCCs using CCA. The authors of the wav2vec 2.0 paper state that the representations from the quantizer can be used as input to other speech recognition systems. We run canonical correlation analysis on the quantizer representations and MFCCs. To do this we extract MFCCs from all test sets with a 30ms window and 10ms stride then perform single value decomposition on the quantizer and MFCC outputs to a dimension of 440 for comparison.

It is possible that the transformer layers may encode information similar to acoustic features found in MFCCs and so CCA is performed on the transformer layers against the MFCCs. Similar to the comparison with contents of features, singular value decomposition is performed on the transformer layers for comparison to MFCCs.

The similarity in Figure 5.4 is minuscule, therefore it is hard to conclude that acoustic representations are included in transformer layers when fine-tuned on low-resource languages. However, there are some differences between layers. Layers 2 and 15 seem to be very similar to acoustic features. The layers are very far from each other, implying that possibly later to encodes information from the encoder and layer 15 has supplementary information similar but not necessarily linked to Mel coefficients.

5.3.3 Is linguistic information encoded?

Studies of English pre-trained wav2vec show that some transformer layers have much higher similarity to GloVe embeddings to others. Here we compare the transformer layers to the GloVe and FastText word embeddings for 50 Yoruba words. Due to the lack of resources in this scenario, we take a small sample size of word dictations from a

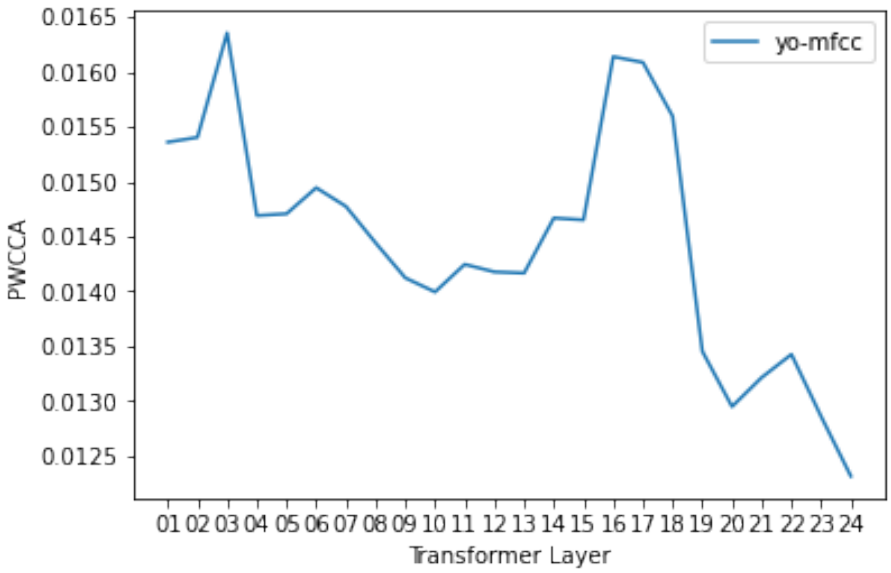

Figure 5.4: PWCCA of YO Transformer Layers and MFCCs

word list and compare them to FastText and GloVe representations of the word respectively. Provide pre-trained Fasttext embeddings of Yoruba text sourced from common crawl and Wikipedia. We train glove embeddings with the same data to produce comparable with the representations of the Yoruba words. FastText encodes words on both upward and individual word level, however glove word vectors do not. In the list of 50 words sampled for these experiments, five of the words were not found in the glove vocabulary and therefore the word vectors were estimated using the authors' knowledge of synonyms and averaging those embeddings. To make comparisons with transformer layers, singular value decomposition of the layers was done to select the 300 singular values to compare to the 300 dimensional first text and glove embeddings.

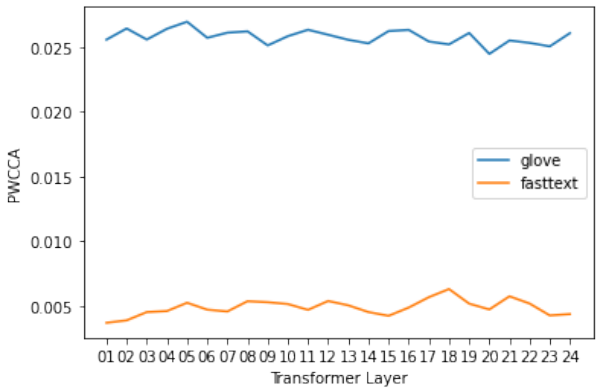

Figure 5.5: Cosine similarity of YO-YONG and NG-YONG model weights

The PWCCA of YO transformer layers with GloVe and FastText vectors is shown in Figure 5.5. It is important to note that none of the layers have a PWCCA of greater than 0.03 for either type of word embedding. We can conclude that in the low-resource fine-tuning case, none of the transformer layers of the wav2vec 2.0 architecture encode linguistic information. This result is in contrasts to experiments done on English as there is a high chance the model is exposed to each word in the vocabulary only once and with training objectives focused on very short snippets of audio, there isn't enough repeat exposure for the model to learn sub-word and word units. That being said, GloVe embeddings are 5 times as similar as FastText embeddings on average. This could be due to FastText inaccurately learning subwords as the model was trained without linguistic knowledge of Yorùbá.

5.3.4 Is code-switching encoded in the model?

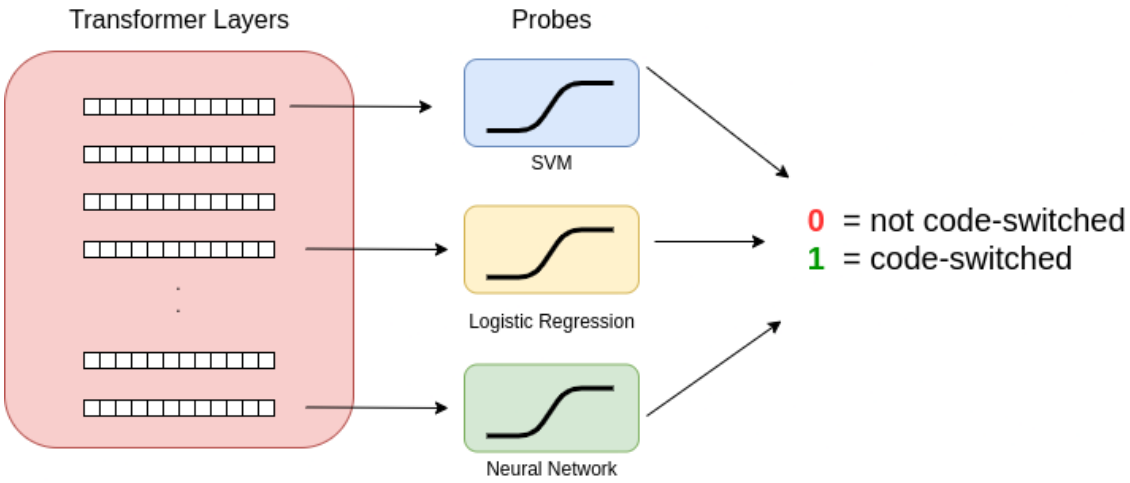

Figure 5.6: Visual representation of the experiment titled *Is code-switching encoded in the model?*

To investigate whether code switching is encoded in the multilingual model, we run a set of classification tasks on test data from the bilingual model and code switched data. Given that none of the codes which data was used as training data for this model we use the entire code switch dataset as positive labels for the experiments to balance the 350 instances we take 175 instances of Yoruba monolingual data and 175 instances of Nigerian English monolingual data to balance the dataset. We then run a variety of classification tasks on the data with an 80/20 training test split. An SVM logistic regression classifier neural network with one layer and your network with two layers

are run on this data and macro F1 scores from the test data are reported.

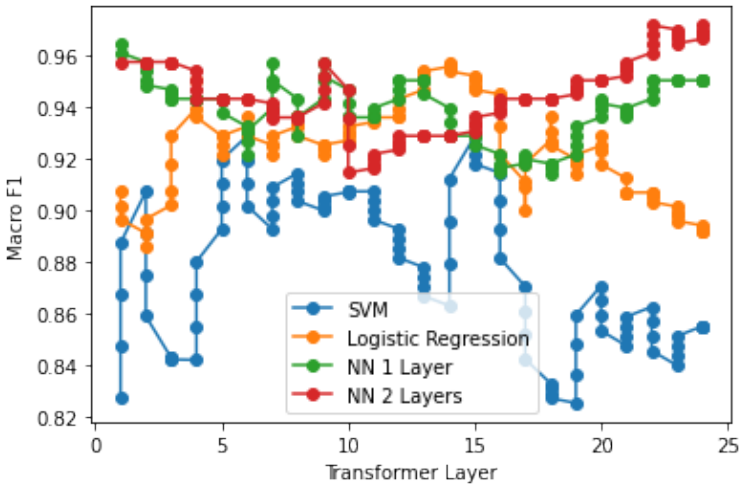

Figure 5.7: F1 scores of various classifier probes

F1 scores of various program classifiers on the code-switch detection dataset in Figure 5.7 show that most layers do and encode differing information for code-switched and monolingual utterances, as all scores are above 0.82. With peaks and troughs varying across classifiers it is consistent between logistic regression and neural network classifiers that F1 score reaches 0.94 between layers 12 and 15 suggesting that search layers encode more information than other layers but due to the differing schools across classifiers it is difficult to conclude that any particular layer can be extracted and used to classify code-switched utterances in future.

Chapter 6

Discussion

6.1 Implications of the Results

The experiments show that multilingual fine-tuning of pre-trained cross-lingual representations is a great way to build initial speech recognisers in low-resource languages. For code switching, the models handle this phenomenon in the zero resource setting better than when there is a small amount of supervised data to fine-tune the model on.

Is it surprising that the zero-resource setting works best for code-switching transcription, but such results may highlight that the training data or fine-tuning data for such models is extremely important in dictating the results.

Results from the second study nicely compliment the results from the first, as it is shown that the multilingual representations do not encode linguistic naughty they encode acoustic information very well in the transformer layers, therefore complexity factors or code mixing types are transcribed with very similar accuracy.

Another link across studies is seen in the effectiveness of code-switching detection in the multilingual model. Although it would be nice to conclude that switching is effectively detected in the model, it is clear from the results in the first study that there may be a significant difference in the code-switching dataset, to the monolingual datasets (possibly quality of microphones used) leading to such an accurate prediction of code-switched data.

From study 1 as a whole, we can conclude that training data for fine-tuning is incredibly important 2 to the transcription accuracy of such a model. From study 2 it can be concluded that linguistic and acoustic information only have the opportunity to be encoded in such models when there are very large amounts of data in languages such as English. Also, multilingual representations of the same training data are distinct from

the monolingual counterparts. Whether it's due to the nature of the data to the linguistic phenomenon itself, such motor Dingle models can detect whether an utterance has been code switched through its representations.

6.2 Limitations of the work

It is clear from this work that wav2vec 2.0 with CTC predictions performs the conditionally independent character based predictions as well as it can with low-resource fine-tuning data. Unlike English where there are vast amounts of data in low-resource and zero-resource settings the model is reduced to focusing only on the task of predicting characters for very short segments of input and therefore any other information acoustic, linguistic, or otherwise is not able to be encoded in the abstract representations of the model.

6.3 Future Work

This work calls for further experiments using a lexicon or a language model to map CTC predictions to words in either Nigerian English or Yoruba. Such an addition to the model will diverge from a zero-resource scenario, but can greatly improve transcription accuracy of models. Training with data of a similar type, possibly all data from the same genre or recording type including code mixing, code-switching, and monolingual sentences will also bring clarification as to whether the results from this work are data dependent or linguistically informed.

To make general conclusions on the code switching phenomenon in multilingual models, it is also important to run experiments on varying language pairs and possibly even language trios. We use a relatively difficult situation where both languages use the same character set with the only difference being the addition of diacritics to Yoruba. Language pairs such as Hindi-English or Arabic-English will give insight to the difference in representations when there are increasingly diverging phonetic and textual representations of data, and whether this can be taken advantage of in the case of code switching to naturally improve transcriptions due to the abstract representations of the training data.

In order to improve transcription accuracy irrespective of language, feeding the layers below the final transformer layer into a decoder may increase transcription accuracy given that the final layers are tuned for a mask prediction task.

Chapter 7

Conclusions

Code-switching is a widely occurring linguistic phenomenon that must be handled by ASR systems to ensure accurate transcription of multilingual utterances. Through experimentation of fine-tuning procedure, Study 1 reveals that a bilingual model without code-switched data performs best on unseen code-switched data, indicating that the quality of fine-tuning data supersedes the suitability of the data to the task. Analysis of the latent representations produced at inference demonstrate that multilingual models have completely different representations to monolingual models, despite identical transcription accuracy. In the low-resource, unseen language case, Study 2 shows that linguistic and acoustic information are not encoded as clearly as they are for English models. This suggests that multilingual ASR models should be pretrained models, but in the low resource case the self-supervised representations are still a black box.

Bibliography

[1] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net, 2017.

[2] Ana Inés Ansaldo, Karine Marcotte, Lilian Scherer, and Gaelle Raboyeau. Language therapy and bilingual aphasia: Clinical implications of psycholinguistic and neuroimaging research. *Journal of Neurolinguistics*, 21(6):539–557, 2008. Acquisition, Processing and Loss of L2: Functional, cognitive and neural perspectives.

[3] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460. Curran Associates, Inc., 2020.

[4] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4960–4964, 2016.

[5] Urmi Chana and Suzanne Romaine. Evaluative reactions to panjabi/english code-switching. *Journal of Multilingual and Multicultural Development*, 5(6):447–473, 1984.

[6] Monojit Choudhury, Kalika Bali, Sunayana Sitaram, and Ashutosh Baheti. Curriculum design for code-switching: Experiments with language identification and language modeling with deep neural networks. In *Proceedings of the 14th Inter-*

national Conference on Natural Language Processing (ICON-2017), pages 65–74, 2017.

[7] Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. Unsupervised cross-lingual representation learning for speech recognition, 2020.

[8] David M Eberhard, Gary F Simons, and Charles D Fennig. Yoruba.

[9] Björn Gambäck and Amitava Das. On measuring the complexity of code-mixing. In *Proceedings of the 11th International Conference on Natural Language Processing, Goa, India*, pages 1–7, 2014.

[10] distributed by Open Speech Google and Language Resources (OpenSLR). Crowdsourced highquality nigerian english speech data set by google.

[11] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006.

[12] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. Deep speech: Scaling up end-to-end speech recognition, 2014.

[13] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China, November 2019. Association for Computational Linguistics.

[14] Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[15] Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. Towards end-to-end automatic code-switching speech recognition. *arXiv e-prints*, pages arXiv–1810, 2018.

[16] Shaoshi Ling and Yuzong Liu. Decoar 2.0: Deep contextualized acoustic representations with vector quantization, 2020.

[17] Shaoshi Ling, Julian Salazar, Yuzong Liu, and Katrin Kirchhoff. Bertphone: Phonetically-aware encoder representations for speaker and language recognition. In *Speaker Odyssey*. ISCA, 2020.

[18] Andy T Liu, Shu-wen Yang, Po-Han Chi, Po-chun Hsu, and Hung-yi Lee. Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6419–6423. IEEE, 2020.

[19] Danni Ma, Neville Ryant, and Mark Liberman. Probing acoustic representations for phonetic properties. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 311–315, 2021.

[20] Oladimeji Olaniyi and Ubong Josiah. Nigerian accents of english in the context of world englishes. *World Journal of English Language*, 3, 01 2013.

[21] Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. Layer-wise analysis of a self-supervised speech representation model, 2021.

[22] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[23] SaiKrishna Rallabandi, Sunayana Sitaram, and Alan W Black. Automatic detection of code-switching style from acoustics. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 76–81, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[24] Jui Shah, Yaman Kumar Singla, Changyou Chen, and Rajiv Ratn Shah. What all do audio transformer models hear? probing acoustic representations for language delivery and its structure, 2021.

[25] Inyang Udofot. Stress and rhythm in the nigerian accent of english. *English World-Wide*, 24:201–220, 12 2003.

[26] Qinyi Wang, Emre Yılmaz, Adem Derinel, and Haizhou Li. Code-switching detection using asr-generated language posteriors. *Proc. Interspeech 2019*, pages 3740–3744, 2019.

[27] Genta Indra Winata, Samuel Cahyawijaya, Zhaojiang Lin, Zihan Liu, Peng Xu, and Pascale Fung. Meta-transfer learning for code-switched speech recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3770–3776, Online, July 2020. Association for Computational Linguistics.

[28] Emre Yılmaz, Henk van den Heuvel, and David van Leeuwen. Code-switching detection using multilingual dnns. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 610–616. IEEE, 2016.

[29] Emre Yılmaz, Henk van den Heuvel, and David van Leeuwen. Code-switching detection using multilingual dnns. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 610–616, 2016.