A Deep Learning Model for Western Neo-Aramaic Speech Recognition

Philipp Burlakov
School of Linguistics
National Research University Higher School of Economics
Bachelor Thesis
Research Advisor: Eduard Klyshinsky

Moscow 2024

Table of Contents

Abstract

The automatic speech recognition (ASR) task has recently become one of the crucial branches of the natural language processing field owing to the wide range of such models' applications. Many companies develop voice assistants, audio translators, and transcribers for major languages. Still, there is no doubt that ASR systems are just as useful for low-resource languages, particularly for fieldwork purposes. In the framework of this project, we propose an ASR deep learning model for the Maaloula dialect of the Modern Western Aramaic language (MWA). MWA is a minor language spoken in the suburbs of Damascus, Syria. There are three villages where the language can be found: Bakh'a, Jubb'adin, and Maaloula, each having its dialectal differences. Current work focuses on the latter for it being best documented so far. However, we would also consider the other two in future work. Developing the system includes choosing an initial model to fine-tune and prepare for our language, collecting the data, the learning process, and evaluation. Our work involves data augmentation, i. e. artificial generation of additional data and training a language model to enhance the model's performance.

1. Introduction

Automatic speech recognition (ASR) is a machine and deep learning (ML and DL) task that aims to process audio, containing speech and return text as an output. ASR systems are included in voice assistants and audio translators for major languages, that have numerous data sources available and are demanded by companies and their clients. Nevertheless, ASR is just as useful for minor and low-resource languages: during scientific fieldwork on one hand and machine translation on the other.

ASR for low-resource languages is slightly different from the classic ASR systems development as we constantly face a lack of data. Therefore, the main objective of researchers is to find ways to use all reachable resources and make the most of them. Later in this work, we will discuss such approaches to the problem as fine-tuning, continuous pre-training (CPT or PT), which makes use of untranscribed audio, using language models (LMs) on the decoder step for presumably better quality of output texts, and data augmentation, which implies adding artificially generated information to the training set.

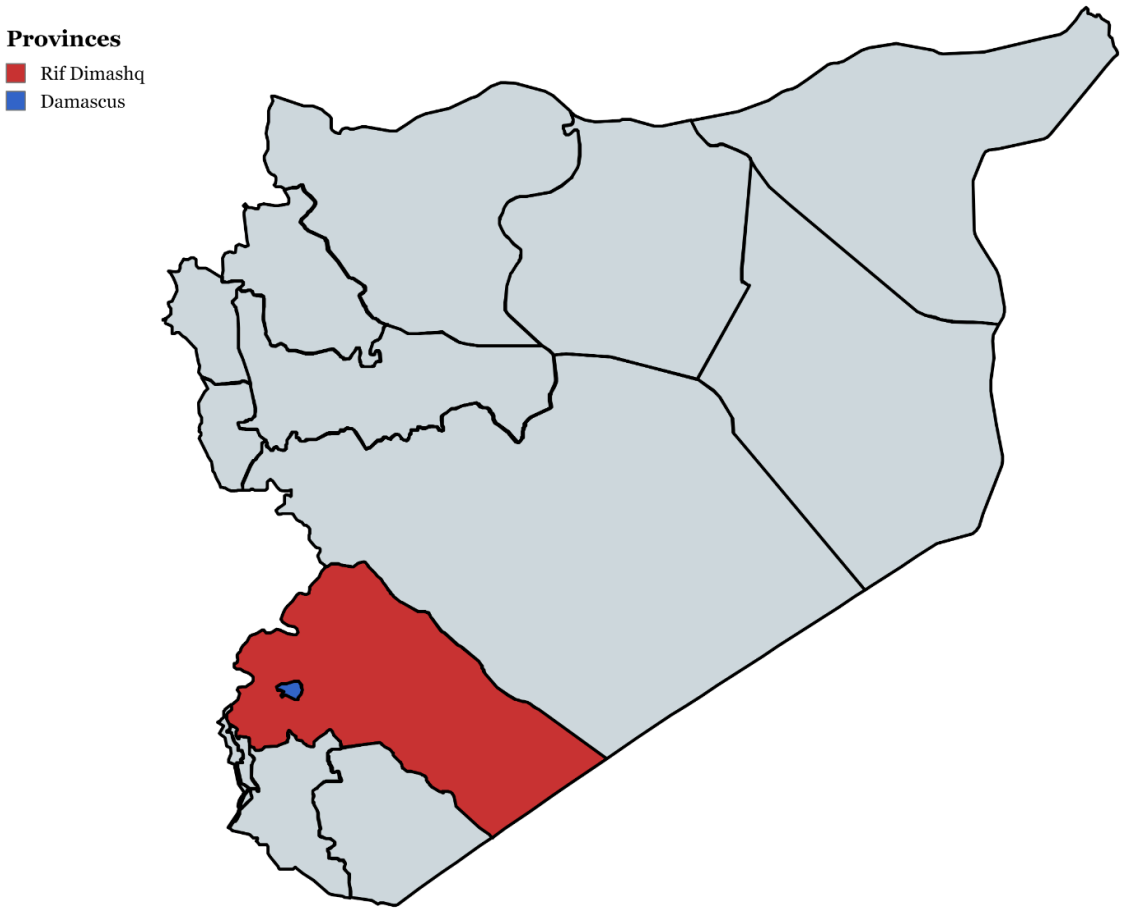
Within the framework of this project, we have developed an ASR system for the Modern Western Aramaic language (MWA), which is a low-resource language[1]. MWA is an endangered North-central Semitic language of the Levantine group. Native speakers of MWA

[1] Glottocode: west2763, ISO 639-3: amw. Also known as Western Neo-Aramaic (WNA) and Siryon

live in the suburbs of Damascus, in three villages: Bakh'a, Jubb'adin, and Maaloula, dividing the language into three corresponding dialects[2]. There are noticeable differences between them, so we have decided to choose only the latter to train and test our model since it is represented with a bigger amount of data than the other two. Moreover, there are ongoing fieldwork projects related to the dialect of Maaloula, so it is best to begin with this one. However, in the future, we are considering scaling the system to cover all three dialects.

There are no ASR models for any of the Aramaic languages, including MWA. Consequently, our project is a starting point of work towards developing ASR systems not only for Western Aramaic languages but for all the living Aramaic languages, such as Turoyo, Urmi, and others. This makes the process of analyzing and publishing field data much easier and faster and helps reach higher efficiency and quality of language documentation. Besides that, it allows for automatic corpus enrichment, so the scientists can use the data that has not been manually transcribed.

Figure 1. *Map of Syrian provinces*



[2] Maaloula dialect of MWA glottocode: malu1250

2. Literature review

2.1. Language of Maaloula

According to Glottolog[3], the idiom of Maaloula belongs to the Aramaic branch of North-west Central Semitic languages. Its closest relatives are other Modern Neo-Aramaic languages (dialects), that fall into the same subgroup. The Aramaic branch also includes different stages and varieties of Syriac, Mandaic, and Aramaic languages themselves.

Another branch of North-west Central Semitic languages is the Canaanite branch with Hebrew and Phoenician languages. By far the biggest group of Central Semitic languages is taken by different varieties of Arabic.

MWA has no writing system, and the scientists in their works usually use modified phonetic Semitic transcription (Kogan & Loesov, 2009). Neo-Aramaic of Maaloula has no official status and is highly influenced by local Levantine Arabic, particularly in terms of phonetics.

The MWA dialect of Maaloula has around 30 consonants and 10 vowels (Arnold 1990). The consonantal system has 4 distinctive features:

- manner (e.g. *b* vs. *m*);
- place (e.g. *z* vs. *h*);
- voice (e.g. *θ* vs. *ð*)
- emphaticity, i.e. pharyngealization (e.g. *t* vs. *tˤ*).

The vowels differ in:

- height (e.g. *o* vs. *u*);
- backness (e.g. *u* vs. *i*);
- length (e.g. *a* vs. *aː*).

Further phonological and phonetic information with the transcription principles for the Maaloula dialect is presented in (Duntsov et al., 2022). We mostly adopted the listed inventories and rules, although we also had to take into account the necessity to uniform the data, and recent novelties in the rules of transcription of the research group. The finalized standards we followed can be found in the preprocessing section.

As mentioned, Maaloula is located in the Governorate of Damascus (Rif Dimashq, see Figure 1), so Modern Western Aramaic has been in constant contact with Damascene Arabic (Syrian Levantine Arabic). That is why the latter is the closest high-resource language to the language of Maaloula in terms of phonetics. They have a similar vocalic inventory besides

[3] https://glottolog.org/

that schwa in MWA does not have a phoneme status (for the vocalic inventory of MWA, see Appendix B). However, Damascene Arabic shows the syllabic distribution of short vowels, which does not take place in the dialect of Maaloula.

Table 1. *Distribution of short vowels in Damascene Arabic (Klimiuk, 2013)*

Syllable Type	Short Vowels				
Open and final syllable	*a*	*e*	*i*	*o*	*u*
Closed and final syllable		*e*		*o*	
Remaining syllables		*ə*			

Modern Standard Arabic (MSA), for comparison, has much more differences with MWA. Firstly, MSA lacks the *e* and *o* phonemes, both long and short, and the schwa sound is not present in it[4]. On the other hand, the consonantal inventory of MWA is less similar to the Damascene Arabic; for example, it does not have interdental fricatives, which are highly frequent in MWA and MSA.

2.2. ASR for low-resource languages

We started the research with the analysis of ASR for Arabic dialects, as there are no released speech-to-text models for any of the Aramaic languages.

A literature review of ASR for different varieties of Arabic by Dhouib et al. (2022) shows that most works cover only Modern Standard Arabic. Only two of the listed works involve Levantine data and none has open weights to use the developed systems. Therefore, because of the lack of recent studies in Levantine Arabic computational phonetics, we concluded that it would be best to follow the current state-of-the-art approach, which is to start with multilingual models, which commonly outperform monolingual ones in terms of fine-tuning, as Bartelds and Wieling (2022) found out.

Yu et al. (2023) came up with a solution to a slightly different task: their goal was not only to develop methods for efficient low-resource language ASR but also to obtain a multilingual fully functioning system. Despite this, the paper presents a summary of relevant challenges and research, such as the first attempts to adapt multilingual models to new languages with limited data (Deligne et al., 2001; Miao et al., 2013). Two key concerns considered by the authors are overfitting (Hou et al., 2021) and forgetting (Kessler et al.,

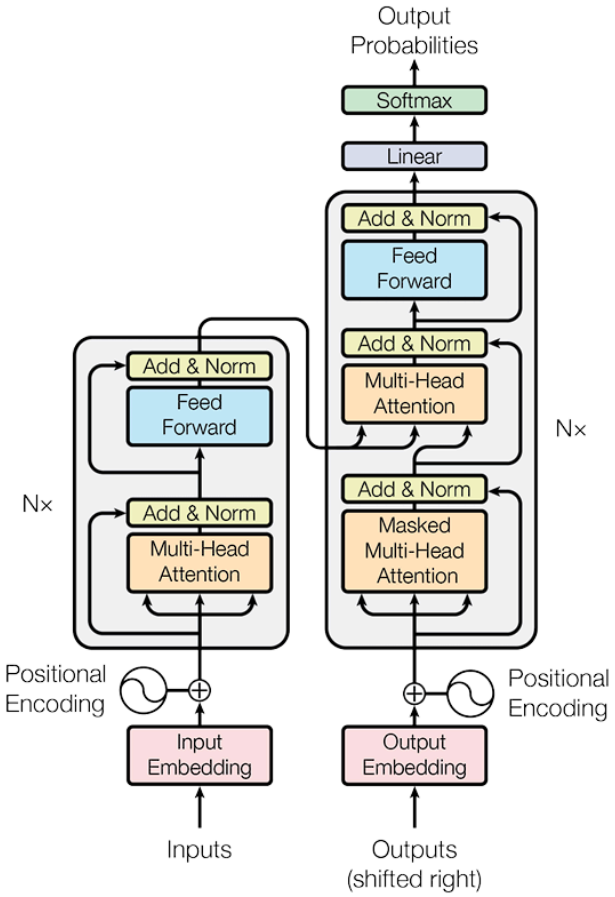
[4] Here, we present an analysis of Classical Arabic, because it is basically a dialect-independent shape of MSA (Birnstiel, 2019)

2021). The first one implies that a model learns to process the training dataset but not any other input and the second one encompasses cases when we observe a decrease in recognition quality on previous languages while multilingual models learn new ones. The latter does not affect our project, whilst the other may cause problems during the learning process.

2.2.1. Base models

When it comes to the choice of a base model to adapt to the task, recent studies normally take into account Whisper and XLS-R, both of which involve transformer models' architecture.

Figure 2. *The transformer model architecture (Vaswani et al., 2017)*

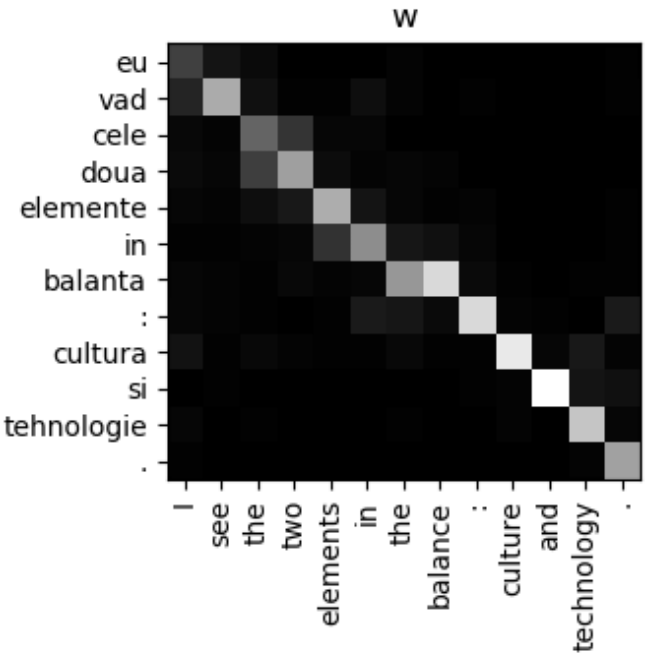


Note: The left part depicts the encoder architecture and the right one – the decoder. The information from the encoder is passed to the decoder alongside the previously generated elements.

Transformers are developed as encoder and decoder parts consisting of special blocks. Their purpose is quite self-explanatory: the encoder part turns an input sequence into a digital mathematical representation, while the decoder uses this representation to produce the right sequence back.

The main part of transformers is the attention block, which distributes relevant information across the entities of a given sequence to capture relations between them (e.g. syntactic, phonetic, semantic, etc.) and map them to the corresponding result.

Figure 3. *Example attention weights for attentional encoder-decoder (Collier & Beel, 2019)*



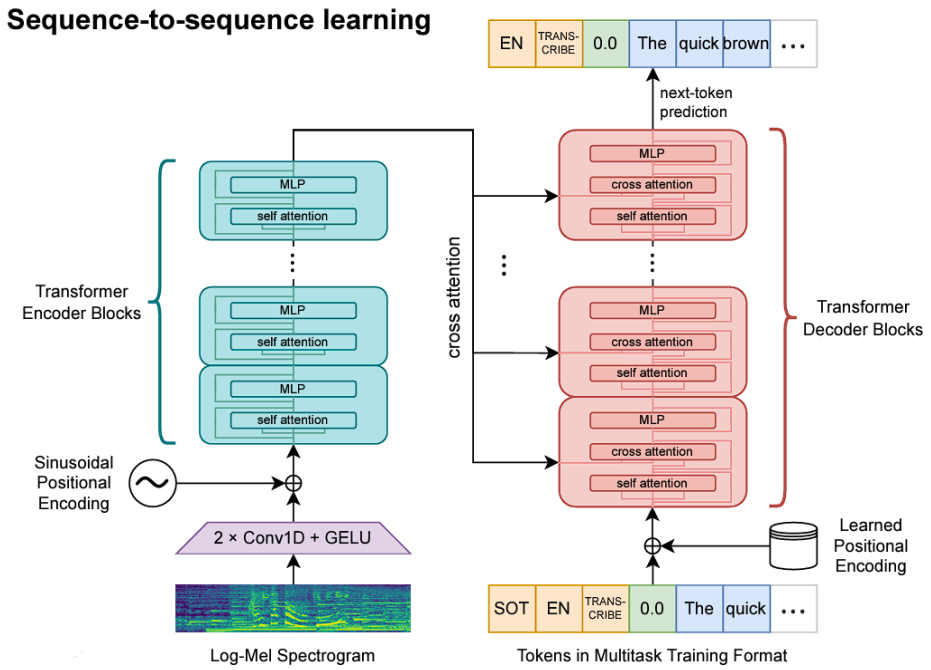
Note: This picture is an example of an attention mechanism in machine translation. Here, we can observe the way the algorithm finds correspondence between two sentences, one in Romanian and the other in English.

Just like in the example above, transformers with their attention blocks can map sound waves to the corresponding characters, taking into account phonetic surroundings and orthographic rules.

Whisper models represent an encoder-decoder transformer architecture, which is simultaneously taught to complete several tasks, such as ASR, language classification, translation, and more.

Whisper is trained using only labeled speech data in several languages, however, it involves noisy data as well, which makes the training process weakly-supervised.

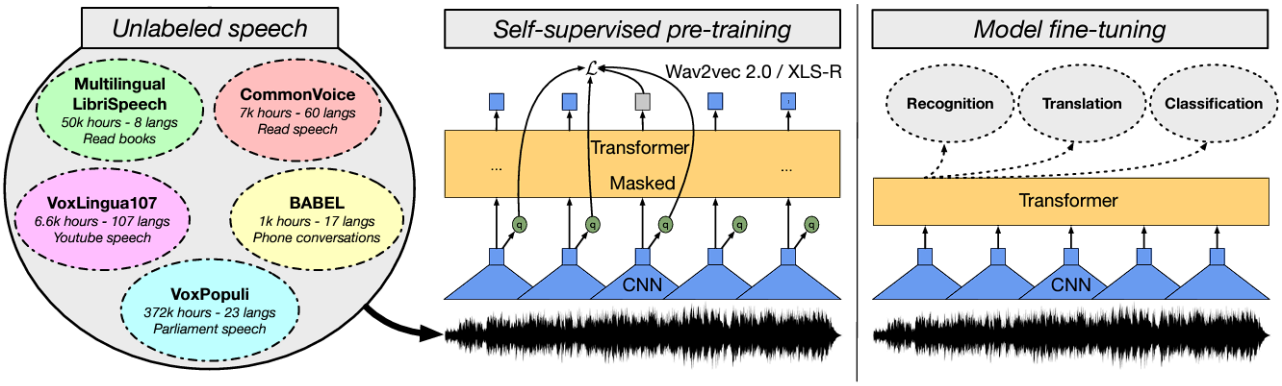
Figure 4. *Whisper architecture (Radford et al., 2022)*



XLS-R, in contrast, uses unlabelled data as well. It is based on Wav2Vec2.0 (Baevski et al., 2020) architecture, which learns a cross-lingual contextual representation of sounds before being trained for the target task. This process is called self-supervised pre-training. These representations help the transformer architecture receive more relevant information about sounds and their biggest advantage is that they do not require data labeling.

Rouditchenko et al. (2023) conclude that XLS-R mostly outperforms the other models on unseen languages, so we will focus on it.

Figure 5. *XLS-R training pipeline (Babu et al., 2021)*



2.2.2. Data augmentation

To avoid overfitting, data scientists apply data augmentation techniques. Those are usually rule-based algorithms that apply random changes to the training set on the go, such as adding noise or lowering the quality of input samples. Moreover, there are types of data

augmentation that involve the generation of additional synthetic data, like in Bartelds et al. (2023), where they use a pre-trained model to transcribe unlabeled audio and add it to a dataset for another model to base on (so-called ‘self-training approach’). Furthermore, for one of the languages in the article, the authors use a pre-existing text-to-speech (TTS) model to generate more audio data. Both approaches sufficiently increase the quality and mitigate the effect of data scarcity, although the second one does not apply to our work.

However, DeHaven and Billa (2022) find out, that for XLS-R, continuous pre-training shows results close to the aforementioned self-training approach while being more computationally efficient. They also conclude that the implementation of both guarantees the best performance, presenting the corresponding pipeline.

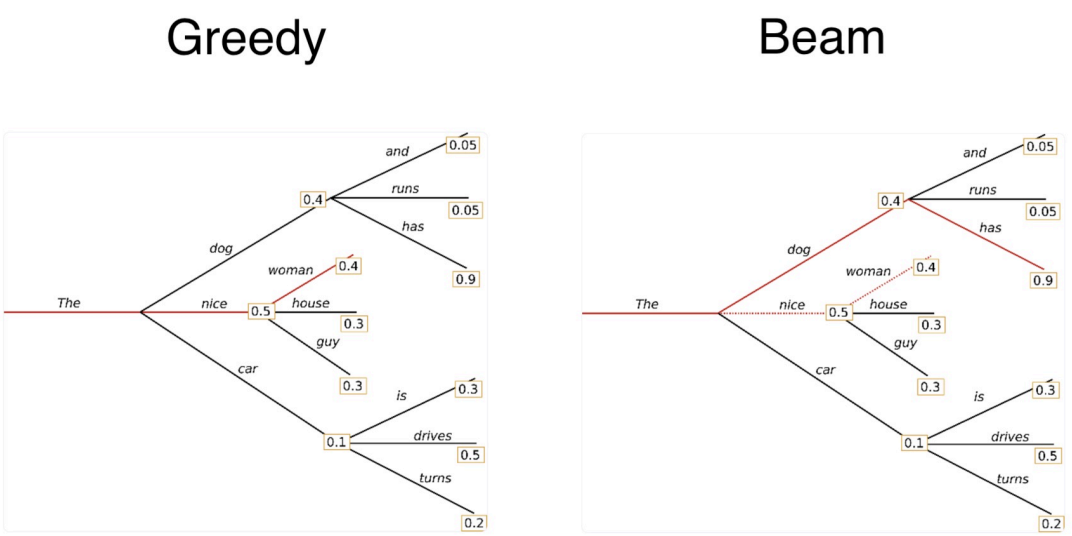
2.2.3. Decoding techniques

Before a model turns its inner representation into the final answer, it deals with probabilities computed for each character. There are three main options to decode the model’s probability distribution:

1. Greedy decoding – on each step, the most likely symbol is chosen no matter the probability of the eventual string;
2. Beam search – the probability of the overall output is measured and the most likely one is returned. It has a *number of beams* parameter, which corresponds to the number of options suggested in each decoding step;
3. Beam search with a language model – the probability of the output is calculated using not only the acoustic model but also text-based statistics.

In the article by Rouditchenko et al. (2023), the authors tried to enhance a model’s performance on the decoding step, when the output text sequence is generated: they compared a greedy and a beam search without LM for Whisper, however, a beam search improved the results insignificantly. Despite this, there is a rather promising increase in metrics with LM implementation in (San et al., 2023). The authors show that a corpus of texts can decrease error rates up to four times given enough data. However, they conclude that small corpora are not enough for a language model to make any change.

Figure 6. *Comparison of greedy decoding and beam search*[5]



Note: This example from text generation shows how greedy decoding and beam search differ. Here, we can observe that given the first word 'the', the greedy decoding chooses the option with the highest probability on each step, while the beam search generates 3 candidates for each step and chooses the sequence with the best score. That is why the greedy decoding ends up with 'the nice woman', the probability of which is $0.5 \times 0.4 = 0.2$, and the beam search chooses 'the dog has' with the probability at $0.4 \times 0.9 = 0.36$.

2.2.4. Metrics

Formula 1. *Word error rate and accuracy*

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

$$Accuracy = 1 - WER$$

Where:

- S is the number of substitutions;
- D is the number of deletions;
- I is the number of insertions;
- C is the number of correct words;
- N is the total number of words in the target sequence ($N = S + D + C$).

The main metric to estimate ASR systems is the word error rate (WER). There is also the character error rate (CER), which follows the same pattern, but concerning characters

[5] https://heidloff.net/article/greedy-beam-sampling

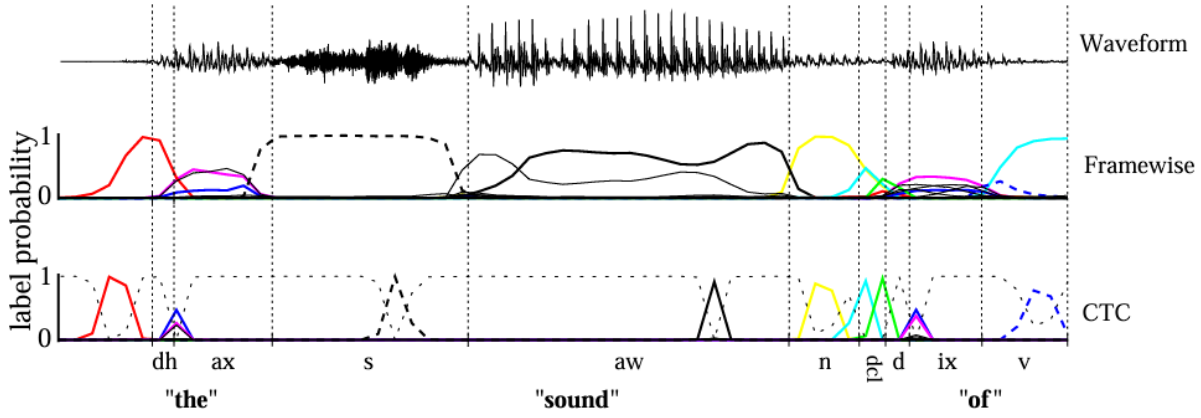
instead of words. CER is usually lower than WER, as one wrong character makes the whole word incorrect. Below, you can find an example of how these metrics are calculated.

Table 2. *Examples of WER and CER*

	WER	CER
Target	Modern **Western Aramaic**	Modern **Western** Aramaic
Prediction	**The** Modern **Aramayc**	**The** Modern Aramayc
Substitutions	1	1
Deletions	1	7
Insertions	1	3
Correct entities	1	12
Rate	1	0.55

The loss function minimized during the training process is the Connectionist Temporal Classification loss (CTC loss). It analyzes possible alignments of the soundwave to the output sequence and sound length, and proposes probabilities of each character in every frame.

Figure 7. *Visualization of the CTC target (Graves et al., 2006)*



3. Methods

3.1. Project Setting and Tools

This project follows the pipeline from the article by Bartelds et al. (2023) and includes five main steps:

1. Data collection and primary processing;
2. Continuous pre-training;
3. Teacher model tuning;

4. Transcription and student model tuning;
5. Evaluation.

This project focused on the XLS-R model, as it is more flexible in terms of using unlabelled data.

Firstly, we collected all available language data and estimated the possibilities to enhance recognition. At this point we had to do primary processing: the texts were standardized and the audio files were extracted from videos and cut to suit the purpose.

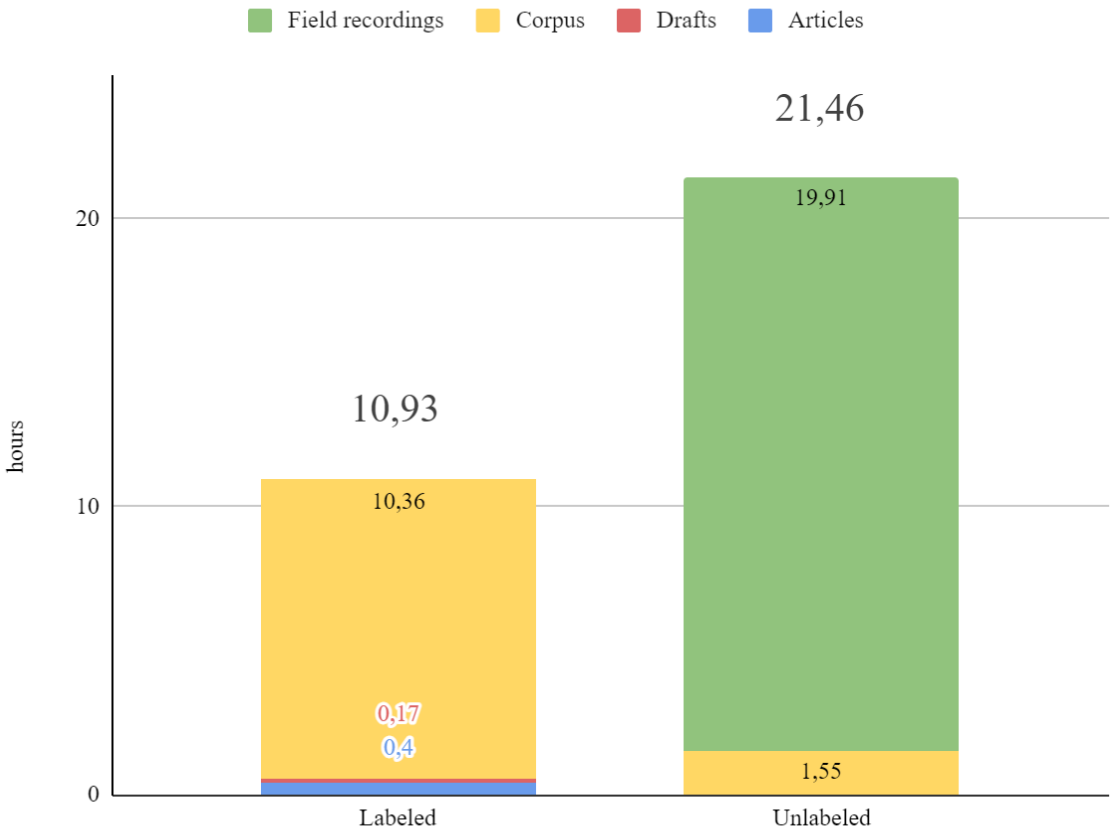
Secondly, we used available audio data to pre-train our model, i.e. to tune the wav2vec feature extractor. After that, we trained our first model on the labeled data. Then we transcribed unlabelled audios with it, which were eventually included in the training dataset for the second round of training.

Finally, we evaluated the quality of the model and the impact of applied techniques.

3.2. Language Data

This project involved gathering all the data available from different sources, including drafts and unpublished papers.

Figure 8. *Total amount of available audio data measured in hours*



The main source happened to be the corpus collected by Werner Arnold (1991a; 1991b; 2006) taking up to 95% of all transcribed material (see Figure 8). Texts were collected from the Moscow Aramaic circle web corpus[6]; audio files were scraped from Semitisches Tonarchiv[7] and matched with corresponding textual data afterward. For data collection and preprocessing algorithms, see our GitHub repository[8]. The corpus does not include any additional annotation: only texts and their German translations can be found there.

Another source is works published by members of a scientific group at HSE University, Moscow, that is dedicated to Modern Aramaic languages (Burlakov et al., 2022a; Duntsov et al., 2022; Bromirskaya et al., 2023a). Additionally, via close interaction with the group, we obtained drafts of future articles (Burlakov et al., 2022b; Burlakov et al., 2022c; Bromirskaya et al., 2023b). These works include the texts themselves, their glossing, and English translation.

Unannotated data consists of recent field audio and video recordings and those recordings from Semitisches Tonarchiv that do not have corresponding texts.

Texts without audio files come from earlier Prym and Socin's (Bergsträsser, 1915) and Spitaler's (1936) works that were digitalized and rewritten to the recent transcription format by members of the group.

The resulting text corpus contains only texts in the preprocessed format without translations, glosses and metadata.

3.3. Preprocessing

As we have mentioned before, we need to process the data so its format does not differ among sources. All audio files were simply converted to WAV extension, while texts had to be standardized. Based on (Duntsov et al., 2022), the works of the research group, and ASR requirements, we developed phonemic inventories (see Appendices A and B) and following transcription rules:

1. Prepositional clitics are separate graphic words;
2. No punctuation and diacritics allowed;
3. No double consonants before a consonant allowed;
4. No individual features of pronunciation reflected;
5. Lowercase.

[6] https://evb0110.github.io/aramaicsite
[7] https://semarch.ub.uni-heidelberg.de
[8] https://github.com/PhilBurub/Modern-Western-Aramaic-ASR

The point of these rules is to get rid of features that cannot be derived from the audio input and minimize the number of transcription options, which helps enhance the model's performance in the context of data scarcity.

4. Model development

4.1. Continuous pre-training

Initial field recordings and corpus audio represented long files containing lots of parts without speech. This would not allow for further evaluator tuning, as firstly, these files are too big to be processed, and secondly, non-speech parts would interfere with the main objective which is to train a model to process raw speech in MWA better.

To cut out the patches we needed, we used scripts from Soundsensing's Machine Hearing project to detect speech[9]. As a result, we processed around three thousand files suitable for pre-training. Both, labeled and unlabelled data were used, however, the test dataset was left out.

With a learning rate at 3e-5 after 9,000 steps we stopped the training process as the model reached the plateaux. In our opinion, the objective of tuning does not differ much from the objective of initial feature extractor training, as the total inventory of human sounds is not language-specific, which explains why we reached the best result that fast.

Our pre-trained model is uploaded to HuggingFace[10].

4.2. Fine-tuning

For fine-tuning we needed smaller audio files, so we followed the example of Eid et al. (2022) to split and align our data using WebMAUS tool[11] (Schiel, 1999, 2015) developed as a part of BAS Web Services (Kisler et al., 2017). We did not simply use The Maaloula Aramaic Speech Corpus because in that case we would need to develop new orthographic processing rules and we would still need to align the latest drafts and articles.

After converting the transcriptions to the SAMPA format, we used the service, having defined the target language as Arabic since the service does not support MWA and Arabic phonetics is the closest to the available languages.

The first model was trained 60,000 steps at learning rate 5e-6 reaching 0.33 word error rate and 0.1 character error rate on the validation dataset with greedy decoding. After that, it was used to transcribe approximately half of the unlabelled data which was included in

[9] https://github.com/jonnor/machinehearing
[10] https://huggingface.co/pburub/Aramaic_pretrained_XLSR
[11] https://clarin.phonetik.uni-muenchen.de/BASWebServices/interface

the training data for the second model (see Figure 9). The checkpoint of the model is available on HuggingFace[12].

Figure 9. *Amount of labeled data and its distribution measured in hours*

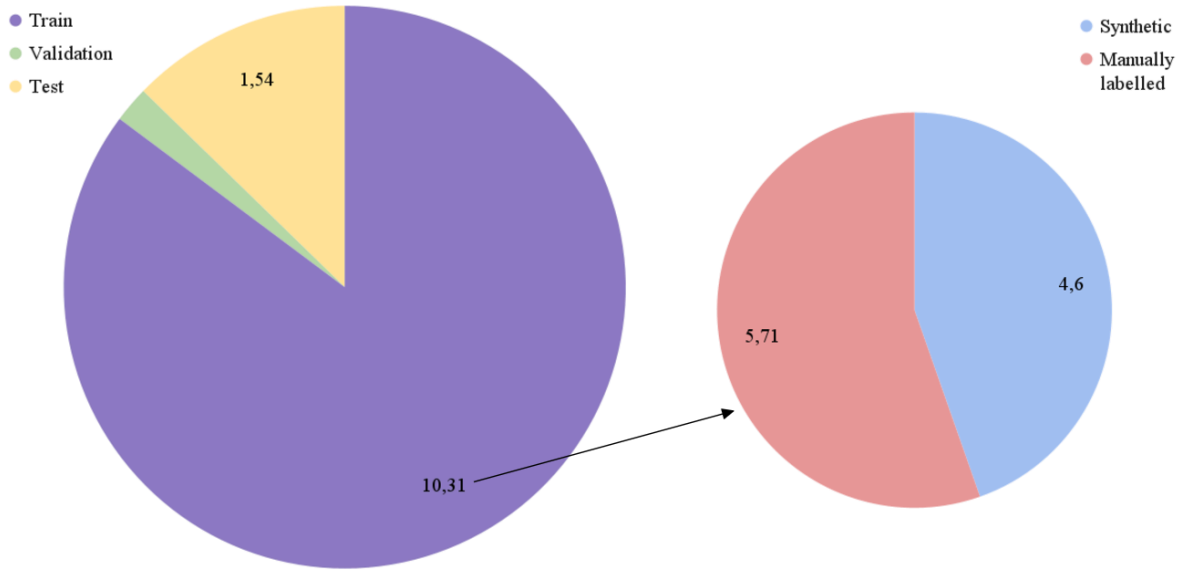
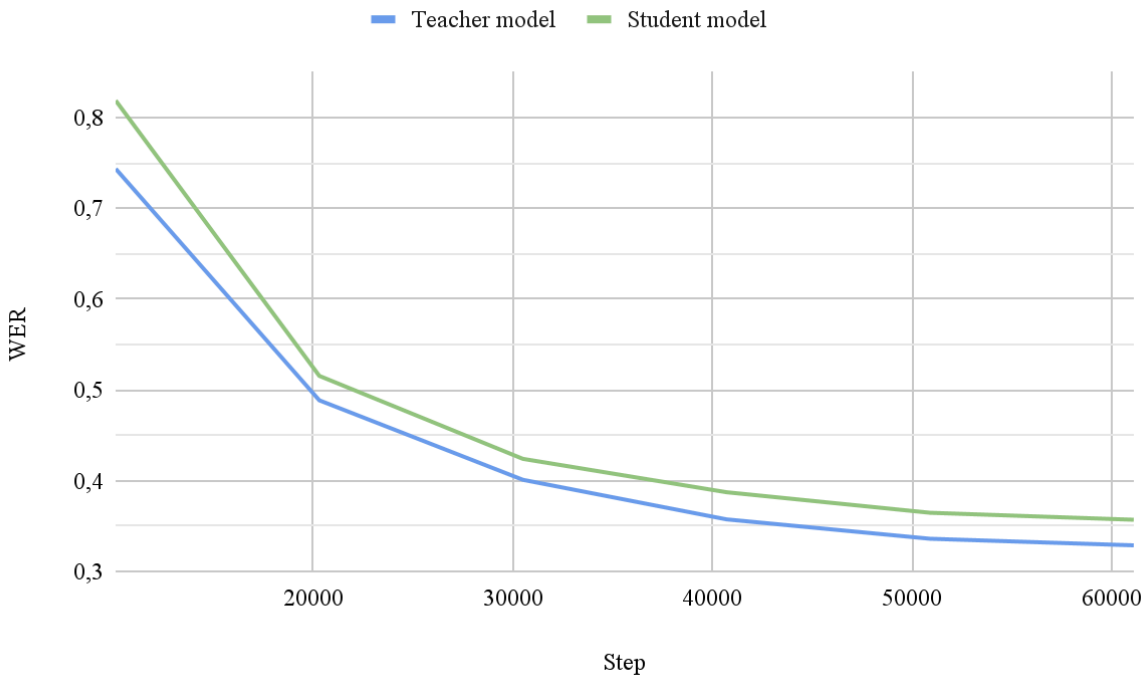


Figure 10. *WER values during the training process*



The second model was trained with the same parameters on the extended training dataset. Test and validation datasets were left unchanged. After 60,000 steps the model's WER and CER lowered to 0.36 and 0.11 respectively. You can find the change of WER on

[12] https://huggingface.co/pburub/Aramaic_finetuned_teacher

the validation dataset in Figure 10 and the metrics of both models on the training dataset in Table 3.

As we can see, with the same amount of training steps, the teacher model performed slightly better during evaluation despite the lack of data augmentation. There may be several reasons, such as:

- the student model had fewer iterations over manually labeled data because we compared models' performance at an equal number of steps, but not an equal number of epochs;
- the teacher model has not reached its highest quality during training, so the automatic labeling was not good enough to be trained on.

Table 3. *Word error rates comparison of the teacher and student models*

		Teacher model	Student model
hours of training data		5.7	10.3
number of epochs		71	49
greedy decoding	***train WER***	.23	.29
	test WER	.37	.40
beam search	***train WER***	.38	.43
	test WER	.50	.52

Either way, fixing that is resource-consuming, so we continued working on the teacher model only. After an additional 60,000 training steps at a 1e-5 learning rate, WER on the validation set dropped down to 0.24 with CER at 0.08. The final model is available on HuggingFace[13].

Table 4. *Word error rates comparison of models with different numbers of training steps*

		60k model	120k model
number of epochs		71	142
greedy decoding	***train WER***	.23	.06
	test WER	.37	.30
beam search	***train WER***	.38	.45
	test WER	.50	.13

[13] https://huggingface.co/pburub/wav2vec2-xls-r-300m-mwa-maaloula

4.3. Decoding options

Following the instructions presented in the article by San et al. (2023), we have built a simple 3-gram language model using texts that are not included in the test dataset. The corpus consisted of around 70,000 tokens and less than 13,000 unique word forms. The results are presented below.

Table 5. *Comparison of different decoding techniques*

		greedy decoding	beam search	beam search+LM
train	***WER***	.06	.25	.23
	CER	.02	.06	.05
validation	***WER***	.24	.41	.31
	CER	.08	.12	.10
test	***WER***	.30	.45	.48
	CER	.10	.13	.16

In the article (Rouditchenko et al., 2023), the authors find the quality improvement using beam search with Whisper. In our case, both beam search and beam search with the language model lowered the quality. We tried to use different numbers of beams from 5 to 50, but the result was always the same.

It is worth looking at the output to explain what may have gone wrong. In Table 6 we can notice 2 main tendencies, that determine the result of decoding:

1. Beam search with and without an LM does not get doubled phonemes right (examples 1, 3, and 5);
2. Beam search with an LM relies on its lexicon, which sometimes actually improves the decoding (examples 2 and 4), but also interferes with the processing of less frequent word forms (*mbašlilla* in example 1, *nimšamyille* and *nmakimille* in example 5) or in cases with assimilation (*makimin* from example 1).

Table 6. *Examples of different decoding techniques' outputs*

	Original text	Greedy decoding	Beam search	Beam search + LM
1	makimin naʕʕīmča mbašlilla	makimin naʕʕīmča mbašlilla	makimin naʕīmča mbašlila	makimil naʕīma baš ila
2	u ṭiflō minbaṣṭin	u ʕw ṭiflō min baṣṭin bē	u ʕw ṭiflō min baṣṭin bē	u ṭiflō minbaṣṭin b
3	u ḥetta yaxfel žorma	u ḥetta yaxfel žorma	u ḥeta yaxfel žorma	ḥeta yafel žorta
4	ṭarša wakčil	ʕṭarša wacṯi	ʕṭarša wacṯi	ṭarša wakčil
5	naʕʕīma nimšamyille nšīfəl cuppō nmakimille	naʕʕīman nimšamyille nšīfəl cuppō nmakimille	naʕīman nimšamyile nšīfəl cupō nmakimile	naʕīma nimšamyila šī əl cupō nmakimila

While the second tendency is pretty logical, the first one seems to be a consequence of an error within the CTC decoding. Further, we use greedy decoding only.

5. Estimation

Besides the metrics and examples presented in Tables 5 and 6, we find it rather informative to estimate the phoneme confusion matrix to estimate the adequacy of the resulting model (see Appendix C).

The first thing that we can conclude from the matrix is that the model does not distinguish between the phonemes /d̯/ and /ḍ/. However the latter phoneme is extremely rare: there are no affixes, that contain it, and only a few roots where it can be found. In the lexicon of the training set, there are only 4 word forms with this phoneme.

Other problems are predictable:

1. Distinction between long and short vowels (e.g. *a* vs. *ā*);
2. Distinction between emphatic (pharyngealized) and non-emphatic consonants (e.g. *z* vs. *ẓ*);
3. Distinction between palatal and velar voiceless consonants (*c* vs. *k*);
4. Schwa deletion;
5. Glottal stop deletion;
6. Word boundary mistakes.

The most unexpected mistake is confusing /g/ and /y/, which is hard to explain. This may be the result of a simplification we made while generating the confusion matrix, as we

used the Levenstein distance algorithm with an assumption that vowels are more likely to be confused with vowels, while consonants are more likely to be confused with consonants.

6. Further work

Right now there is only a command line interface for the model usage. To allow the scientific community convenient and stable access to the model, we plan to develop a minimal interface and possible web-hosting (probably in collaboration with the School of Linguistics or the Institute for Oriental and Classical Studies of HSE).

Additionally, having done the data scraping and preprocessing, we are now able to develop a database that would ease the process of keeping track of all available materials and their status and would grant uniformity of content. This would require us to restore the metadata and to further develop a new interface. However, it may be possible to include this database as the Moscow Aramaic circle web corpus extension, which seems to be a viable solution.

Besides that, as the research moves on to other dialects of MWA, we will use our system as a base model to be finetuned on Bakh'a and Jubb'adin data as well. Eventually, we may also include other living Aramaic languages.

7. Conclusions

Our project serves a rather specific and acute purpose, which is the documentation of the endangered language. The ASR system for the Maaloula dialect of MWA will make the recording transcription faster and less skilled work so that more data can be accumulated and processed by fewer people. This is especially important if we take into account the fact that there is only one scientific group that is involved in the ongoing research concerning the language, while the number of speakers decreases each year (Duntsov et al., 2022).

Moreover, as an interim project result, we have gathered and uniformed all currently available materials in the Maaloula dialect of MWA. In the future, this can be turned into an audio corpus with texts for further studies.

This is the first attempt to develop a speech-to-text solution for Aramaic languages in general. Therefore, our model can become a starting point for creating ASR systems of other living languages of the group, for instance, Bakh'a and Jubb'adin dialects of MWA, Turoyo, and Urmi.

At the same time, this work investigates ways to improve low-resource language speech recognition. The experiments we have carried out with data augmentation techniques, denoising, and decoding methods will be helpful not only for similar projects but also for

general research on ASR systems. In some cases, even for major languages, processing faces similar problems, for example, in low-scale projects or given a narrow task.

Firstly, we have proven that self-training data augmentation, which implies adding synthetically labeled data to the training dataset is indeed less computationally efficient, as DeHaven and Billa (2022) noticed, and therefore may not be reasonable for projects with limited access to the resources.

Secondly, our results turn out to be drastically different from what Rouditchenko et al. (2023) present in their article. For some reason, beam search does not improve decoding owing to the deletion of doubled consonants. Moreover, our corpus with 70k tokens could not change the output of the beam search much. That is why in the future, we will try to fix the problem with beam search and gather more texts to reach results close to those presented by San et al. (2023).

References

Arnold, W. (1990). *Das Neuwestaramäische V. Grammatik*. Harrassowitz.

Arnold, W. (1991a). *Das Neuwestaramäische III. Volkskundliche Texte aus Maʿlūla*. Harrassowitz.

Arnold, W. (1991b). *Das Neuwestaramäische IV. Orale Literatur aus Maʿlūla*. Harrassowitz. https://www.harrassowitz-verlag.de/titel_97.ahtml

Arnold, W. (2006). *Lehrbuch des Neuwestaramäischen*. Harrassowitz. https://www.harrassowitz-verlag.de/titel_1731.ahtml

Babu, A., Wang, C., Tjandra, C., Lakhotia, K., Xu, Q., Goyal, N., Singh, K., von Platen, P., Saraf, Y., Pino, J., Baevski, A., Conneau, A. & Auli, M. (2021). *XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale*. ArXiv:2111.09296. https://doi.org/10.48550/arXiv.2111.09296

Baevsk, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). *wav2vec2.0: A framework for self-supervised learning of speech representations*. ArXiv:2006.11477. https://doi.org/10.48550/arXiv.2006.11477

Bartelds, M., & Wieling, M. (2022). *Quantifying Language Variation Acoustically with Few Resources*. ArXiv:2205.02694. https://doi.org/10.48550/arXiv.2205.02694

Bartelds, M., San, N., McDonnell, B., Jurafsky, D. & Wieling, M. (2023). *Making More of Little Data: Improving Low-Resource Automatic Speech Recognition Using Data Augmentation*. ArXiv:2305.10951. https://doi.org/10.48550/arXiv.2305.10951

Bergsträsser, G. (1915). *Neuaramäische Märchen und andere Texte aus Malula in deutscher Übersetzung, hauptsächlich aus der Sammlung E. Prym's und A. Socin's*. Kommission bei F. A. Brockhaus.

Birnstiel, D. (2019). Classical Arabic. *The Semitic Languages*, 367-402. http://doi.org/10.4324/9780429025563-15

Bromirskaya, A., Häberl, Ch. & Loesov, S. (2023a). The Western Aramaic Context of a Famous Lullaby. *Aramaic Studies*, *21:2*, 205-232. https://doi.org/10.1163/17455227-bja10045

Bromirskaya, A., Häberl, Ch. & Loesov S. (2023b). *Of Rare and Obsolete Words*. Draft.

Burlakov, Ph., Golovina, A., Duntsov, A., Kostomarova, K. & Novokreshchennykh, E. (2022a). Description of the Syrian Civil War by a Resident of Maaloula. *Tirosh. Jewish, Slavic & Oriental Studies*, *21*, 234-247.

Burlakov, Ph., Häberl, Ch. & Loesov, S. (2022b). *The Church Militant: A Modern Western Aramaic Account*. Draft.

Burlakov, Ph., Häberl, Ch. & Loesov, S. (2022c). *Maaloula People in Palestine*. Draft.

Collier, M., & Beel, J. (2019). *Memory-Augmented Neural Networks for Machine Translation*. ArXiv:1909.08314. https://doi.org/10.48550/arXiv.1909.08314

DeHaven, M., & Billa, J. (2022). *Improving Low-Resource Speech Recognition with Pretrained Speech Models: Continued Pretraining vs. Semi-Supervised Training*. ArXiv:2207.00659. https://doi.org/10.48550/arXiv.2207.00659

Deligne, S., Eide, E., Gopinath, R. A., Kanevsky, D., Maison, B., Olsen, P., Printz, H. & Sedivy, J. (2001). Low-resource speech recognition of 500-word vocabularies. *Proc. 7th European Conference on Speech Communication and Technology (Eurospeech 2001)*, 1833–1836.

Dhouib, A., Othman, A., El Ghoul, O., Khribi, M. K. & Al Sinani, A. (2022). Arabic Automatic Speech Recognition: A Systematic Literature Review. *Applied Sciences*, *12*(*17*). http://doi.org/10.1109/ACCESS.2021.3112535

Duntsov, A., Häber, Ch. & Loesov, S. (2022). A Modern Western Aramaic Account of the Syrian Civil War. *WORD*, *68:4*, 359-394. https://doi.org/10.1080/00437956.2022.2084663

Eid, G., Seyffarth, E., & Plag, I. (2022). The Maaloula Aramaic Speech Corpus (MASC): From Printed Material to a Lemmatized and Time-Aligned Corpus. *Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022)*, 6513–6520. https://aclanthology.org/2022.lrec-1.699

Graves, A., Fernández, S., Gomez, F. & Schmidhuber, J. (2006). Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. *Proceedings of the Twenty-Third International Conference (ICML 2006)*, 369-376. https://doi.org/10.1145/1143844.1143891

Hou, W., Zhu, H., Wang, Y., Wang, J., Qin, T., Xu, R., & Shinozaki, T. (2021). *Exploiting adapters for cross-lingual low-resource speech recognition.* ArXiv:2105.11905. https://doi.org/10.48550/arXiv.2105.11905

Kessler, S., Thomas, B., & Karout, S. (2021). *An Adapter Based Pre-Training for Efficient and Scalable Self-Supervised Speech Representation Learning*. ArXiv:2107.13530. https://doi.org/10.48550/arXiv.2107.13530

Kisler, T., Reichel, U. D., & Schiel, F. (2017). Multilingual processing of speech via web services. *Computer Speech & Language, 45(September 2017*), 326–347. http://dx.doi.org/10.1016/j.csl.2017.01.005

Klimuk, M. (2013). *Phonetics and Phonology of Damascus Arabic*. Katedra Arabistyki i Islamistyki Uniwersytet Waszawski.

Kogan, L., Loesov, S. (2009). Neo-Aramaic of Maalula. *Languages of the World: Semitic Languages*, 705-750.

Miao, Y., Metze, F., & Rawat, S. (2013). Deep maxout networks for low-resource speech recognition. *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, 398–403. http://dx.doi.org/10.1109/ASRU.2013.6707763

Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C. & Sutskever, I. (2022). *Robust speech recognition via large-scale weak supervision*. ArXiv:2212.04356. https://doi.org/10.48550/arXiv.2212.04356

Rouditchenko, A., Khurana, S., Thomas, S., Feris, R., Karlinsky, L., Kuehne, H., Harwath, D., Kingsbury, B. & Glass, J. (2023). *Comparison of Multilingual Self-Supervised and Weakly-Supervised Speech Pre-Training for Adaptation to Unseen Languages*. ArXiv:2305.12606. https://doi.org/10.48550/arXiv.2305.12606

San, N., Bartelds, M., Billings, B., de Falco, E., Feriza, H., Safri, J., Sahrozi, W., Foley, B., McDonnell, B., & Jurafsky, D. (2023). *Leveraging supplementary text data to kick-start automatic speech recognition system development with limited transcriptions*. ArXiv:2302.04975. https://doi.org/10.48550/arXiv.2302.04975

Schiel, F. (1999). Automatic phonetic transcription of nonprompted speech. *Proceedings of the ICPhS 1999*, 607–610. https://doi.org/10.5282/UBM%2FEPUB.13682

Schiel, F. (2015). A statistical model for predicting pronunciation. *International Congress of Phonetic Sciences*.

Spitaler, A. (1936). *Grammatik des neuaramäischen Dialekts von Ma'lūla (Antilibanon)*. Brockhaus in Komm.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need*. ArXiv:1706.03762. https://doi.org/10.48550/arXiv.1706.03762

Yu, Zh., Zhang, Y., Qian, K., Wan, Ch., Fu, Y., Zhang, Y. & Lin, Y. (2023). *Master-ASR: Achieving Multilingual Scalability and Low-Resource Adaptation in ASR with Modular Learning*. ArXiv:2306.15686. https://doi.org/10.48550/arXiv.2306.15686

Appendix A

Consonantal inventory of the Maaloula dialect of MWA

	Bilabial	**Labiodental**	**Interdental**	**Alveolar**	**Post-Alveolar**	**Palatal**	**Velar**	**Uvular**	**Pharyngeal**	**Glottal**
Stops	b p - -			d t ḍ ṭ		- c - -	g k - -			ʔ - - -
Affricates					- č - -					
Fricatives		- f - -	ḏ̲ ṯ̲ ḏ̲ -	z s ẓ ṣ	ž š - -			ġ x - -	ʕ ḥ - -	- h - -
Nasal	m - - -			n - - -						
Lateral				l - - -						
Apical				r - - -						
Approximant	w - - -					y - - -				

Note. In each cell, 4 consonants are presented: voiceless, voiced, voiceless emphatic (pharyngealized), and voiced emphatic (pharyngealized) respectively. Consonants *ʔ*, *d,* and *g* do not have a phoneme status. The consonant *ḍ* has an unclear status, as it is only present in several Arabic loanwords.

Appendix B

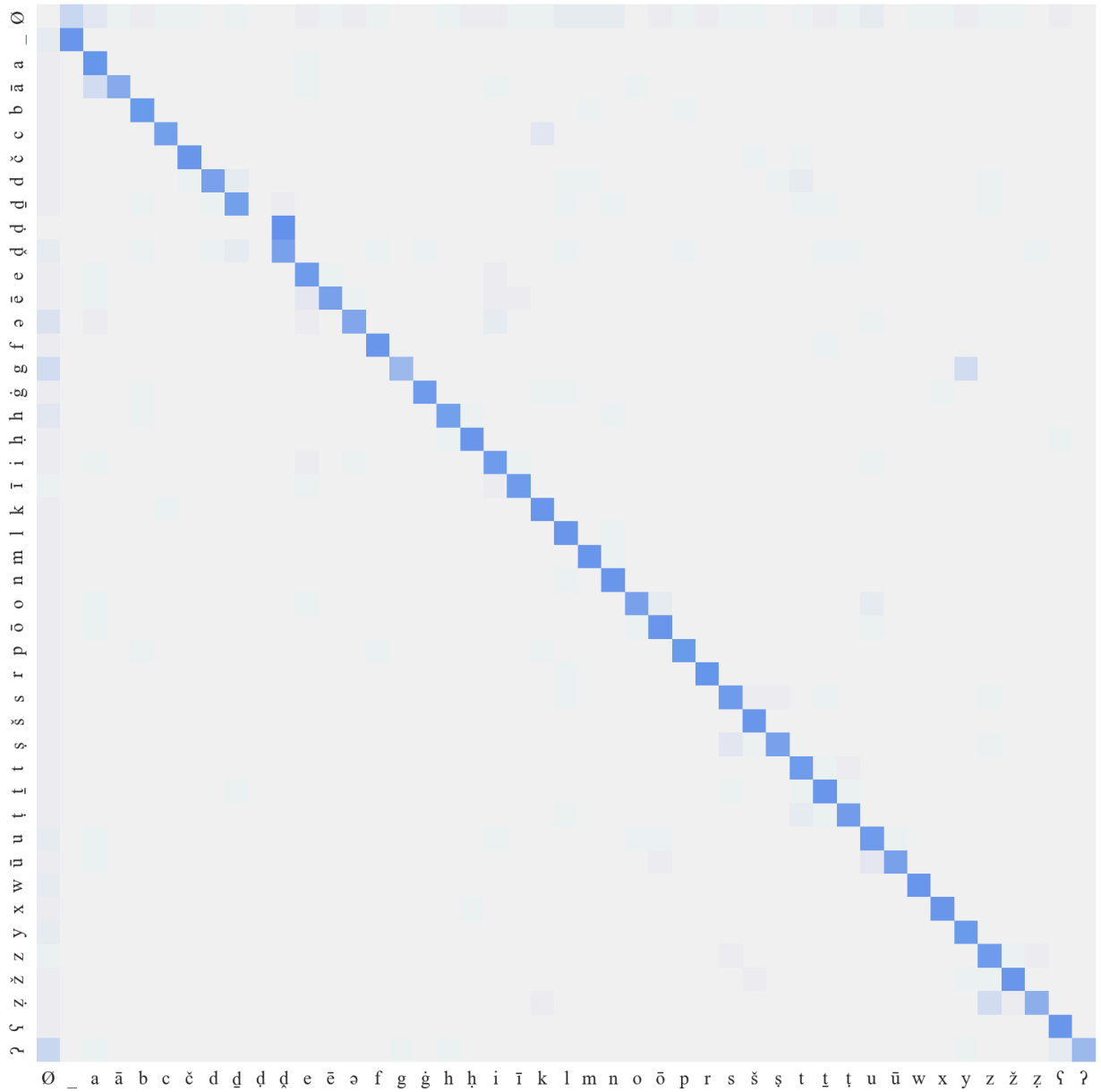
Vocalic inventory of the Maaloula dialect of MWA

	Front	**Central**	**Back**
Close	i ī		u ū
Mid	e ē	ə	o ō
Open		a ā	

Note. In each cell, 2 vowels are presented: short and long respectively. The schwa does not have a phoneme status, it appears as an epenthetic vowel.

Appendix C

Phonemic confusion matrix



Note. Rows represent target phonemes and columns are phonemes from the model's prediction. The symbol "_" stands for word break (i.e. space) and "Ø" is an absence of a corresponding phoneme. The cell color intensity indicates the relative frequency of using a symbol from the lower line in place of a symbol on the right.