The Effects of Fine-Tuning on the ASR Performance of Dialectal Arabic

Ömer Tarik Özyilmaz

university of groningen / campus fryslân

University of Groningen - Campus Fryslân

The Effects of Fine-Tuning on the ASR Performance of Dialectal Arabic

Master's Thesis

To fulfill the requirements for the degree of
Master of Science in Voice Technology
at University of Groningen under the supervision of
Dr. J.K. Schäuble (Voice Technology, University of Groningen)
with the second reader being
Dr. J. Doe (Voice Technology, University of Groningen)

Ömer Tarik Özyilmaz (s3951731)

June 10, 2024

Acknowledgements

First of all, I greatly appreciate the support I have received from family and friends. Of course, I am also glad I met so many great teachers and peers at Campus Fryslân, thank you for making the train rides more worthwhile.

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

I would like to thank everyone reading this thesis, contributing to this thesis by questioning my methodology, and the many people enlightening me on the intricacies of the Arabic dialects.

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

Abstract

Current commercial automatic speech recognition (ASR) applications support the formal Modern Standard Arabic (MSA). Yet, in conversational speech, the speakers' dialect often determines the meaning, diacritics, and accent. In the current study, dialectal Arabic is investigated as a low-resource ASR problem, due to the insufficiency of variation, volume, and balance in dialectal speech datasets. We focus on improving the performance of OpenAI's Whisper on five large Arabic dialects: Gulf, Levantine, Iraqi, Egyptian, and Maghrebi. The effect of MSA training size is evaluated to determine a proper cut-off point for the initial iteration of fine-tuning. Then, the outcome of this pre-training is evaluated to theorize whether the dialects share a commonality with each other and MSA. Finally, the difference in performance between dialect-specific and dialect-pooled models is presented and discussed. After fine-tuning a Whisper checkpoint on Mozilla Common Voice 16.1 for MSA and the large-scale MASC dataset for dialectal Arabic, we demonstrate the results using the word- and character error rate (WER and CER). We find that fine-tuning with a small amount of MSA training data can already show a large increase in performance and perform similarly to much larger models without fine-tuning. The effect of pre-training is minimal, leading us to believe that the differences between each dialect and MSA are too large to generalise. Further, a small drop in performance is found moving from dialect-specific to dialect-pooled models, and contrary to previous studies, we advocate that the benefits outweigh this cost. Dialect-pooled models present a exciting opportunity to reduce the data deficiency problem, especially paired with careful data curation. Overall, our experiments provide valuable insights for improving fine-tuning of dialectal Arabic ASR models and suggest potential implications for other low-resource languages.

Keywords: Automatic Speech Recognition (ASR), dialectal Arabic, fine-tuning, Modern Standard Arabic (MSA).

Contents

Chapter 1

Introduction

Dialectal Arabic automatic speech recognition (ASR) has been a topic of interest as a low-resource ASR problem [1]. Most commercial Arabic speech recognition systems are focused and therefore trained on formal or broadcast speech. This is a standardized form of Arabic that is called Modern Standard Arabic (MSA). However, as with many large languages that span different continents, Arabic knows a large variety of dialects in conversational or natural speech. These are mostly dependent on the region of origin of a speaker and can result in different meanings, diacritics, and pronunciation [2]. It is therefore critical to develop robust ASR systems that allow these dialectal speakers to use, for example, their smartphone's voice assistant in conversational speech and thus with their own dialect. Additionally, many commercial applications also require conversational speech recognition, such as call analytics software for customer support, transcriptions of patient visits in the hospital, and meeting transcriptions in companies. Even though Arabic dialects are not quite as low-resource as most typical low-resource languages because of the large number of speakers, most training data for ASR models has been available in formal MSA speech only, yielding it less accurate for dialectal speech [1]. Alsayadi et al. [1] lay out the following challenges in current Arabic ASR systems:

- There is an insufficiency of dialectal speech datasets.
- Variation in the type of data is lacking.
- Dialectal speech is often exclusively used for evaluation and not for training.
- ASR systems that support multiple dialects perform worse than their dialect-specific counterparts.
- There is an imbalance in the types of dialects, with some dialects being overrepresented while others are forgotten.

The challenges presented clearly indicate a need for a solution that is compatible with low-resource, low-variation datasets. Additionally, a more robust model working for multiple dialects, while taking into account the imbalance in dialects is essential. Recent advancements in multilingual speech recognition have significantly benefited from deep learning technologies, particularly through the use of web-scraped or unlabelled data to address the challenge of limited training resources [3, 4]. Despite these developments, the adaptability and performance of these models across various

languages exhibit a trade-off, often described metaphorically as being ”jack of all trades, master of none.” These models can recognize a broad array of languages but may not achieve high accuracy for each.

Let us demonstrate with an example from one of the current state-of-the-art models, Whisper. On the `small` checkpoint, Whisper achieves a 66.4% word error rate (WER) on MSA, while the same model shows a 14.2% WER on the Dutch test set ([3], Appendix D.2.2). The word error rate is a measure of the percentage of words that are correctly transcribed by an ASR model (see also Section 4.2). For a widely spoken language such as Arabic a WER of almost 70% is very detrimental for its adoption rate by the general public. If a user wants to try speech recognition software and for every ten words at least seven are (partially) incorrect, this leads to much frustration. Considering that the performance displayed is for the most formal form of Arabic as well, one can only speculate the disastrous performance on dialectal Arabic.

Additionally, the substantial computational requirements of these advanced, multilingual models constrain their deployment on devices with limited resources, as optimizing them for such environments could potentially degrade performance for less commonly spoken languages.

All in all, there does not seem to be a general solution to the problem of multilingual speech recognition yet. Nonetheless, this thesis will focus on leveraging the advantages of both large ASR models and using the large amount of MSA speech data to develop a novel dialectal Arabic ASR system and investigate its strengths and weaknesses.

The thesis is structured as follows. Chapter 2 lays out the current background and theoretical framework for the research. Then, in Chapter 3 the methodology is discussed and presented. The setup of the experiments is consequently demonstrated in Chapter 4. The results are presented in Chapter 5 and discussed in Chapter 6. Finally, in Chapter 7, a conclusion is proposed and the implications of this research are summarised.

1.1 Research Question and Hypothesis

The research gaps identified by the current thesis are plural. As laid out by Alsayadi et al. [1], the lack of variation and volume of dialectal Arabic speech data is problematic. Moreover, ASR systems claiming to support multiple dialects, never do so successfully. For example, Abdelali et al. [5] showed good results for one Egyptian dataset using the Universal Speech Model by Google [4], while the same model performed poorly on another Egyptian dataset and a Moroccan dataset. Finally, there is a clear imbalance in the available training data, while properly robust ASR systems should function well across multiple dialects without favouring any dialect due to an abundance of data.

Thus, motivated by the research gaps in fine-tuning for dialectal Arabic and the poor performance of Whisper on Arabic datasets, many questions can be raised. The following thesis will investigate the effect of fine-tuning Whisper first with MSA (pre-training) and subsequently with the Gulf-, Levantine-, Iraqi-, Egyptian-, and Maghrebi-Arabic dialects. The nature and extent of the impact of fine-tuning will then be investigated for the different dialects and across different pre-training configurations. This includes the linguistic diversity across dialects as well as different types and sizes of training data.

To summarize, the research questions are proposed as follows:

Q1. What is the effect of training data size of Modern Standard Arabic (MSA) on the performance of an ASR system?

Q2. Is pre-training on MSA beneficial for the performance of dialectal Arabic fine-tuning?

Q3. What is the difference in performance between dialect-specific and dialect-pooled models in terms of dialectal ASR?

We expect that pre-training on MSA will significantly enhance the performance of ASR systems in the recognition of Arabic dialects. This improvement will emerge from the ability of pre-trained networks to more effectively identify and use linguistic features that are common across these varieties, compared to current Whisper [6]. The extent of this improvement will vary on the variation of data used, the linguistic distance of the dialect to MSA, the amount of MSA pre-training data, and the fine-tuning strategy. We do expect amount of training data to have a positive effect on the performance to a certain extent where the improvement will not be significant anymore [6]. Finally, dialect-specific models should outperform dialect-pooled models on their matching dialect, as they generally do, according to Alsayadi et al. [1].

Chapter 2

Literature Review

Historically, Classical Arabic has been the most important and influential version of the Arabic language. As a language of religion (synonymous to "Quranic Arabic"), it has a rich history. In recent history, however, the rise in Arabic nationalism during the Ottoman rule as a result of Western influence caused many changes to the language [7]. During the revolution, the Arabic language became one of the uniting factors in this nationalism, with the protection of its integrity and the preservation from dialectal and foreign influence becoming crucial. Simultaneously, the need for an expanded lexicon was apparent, to include new (European) ideologies and technical notions. To this extent, the language was gradually reformed to both accommodate vocabulary expansion and to standardize the language over regional variations. This reformed version is commonly called Modern Standard Arabic (MSA). The main differences between Classical- and Modern Standard Arabic are related to the Europeanization of the sentence and verbal constructions, such as the order of words and the introduction of auxiliary words.

Despite the efforts of trying to minimize the regional variation with the "standard" MSA, dialectal Arabic still differs in lexicon use [7]. Two major reasons for the lexical variation between dialects, for example, are (1) the differences in the creation of new vocabulary and (2) the difference in colonial or regional influences to each dialect [8]. The dialect of the speakers in the Maghreb region in North-Africa, spanning roughly from Morocco to Algeria, is influenced by Berbers in the region. Algerian and Tunisian in particular are concurrently affected by the colonial time, borrowing many words from French. Egyptian Arabic similarly draws from English and French impacts. Furthermore, Levantine Arabic, spoken in a region primarily spanning Syria, Palestine, Jordan, and Lebanon, displays more Persian and Turkic influence. Iraqi or Mesopotamian Arabic on the other hand, has relations with Aramaic and Greek. Finally, Peninsular Arabic is mostly affected by Bedouin dialects.

Although there are many differences between dialects, the notion of uniting everyone under one "standard" MSA depicts dialects as plebeian, hindering the research and emphasization of them [7]. As mentioned previously, Alsayadi et al. [1] provided a review of the field of Arabic dialectal ASR in which they discuss the challenges lying ahead. Many of the problems with the availability of datasets, data scarcity and the performance of current techniques are addressed. It becomes clear from this review that there is a lack of studies that aim to fine-tune existing models rather than train a model from scratch or develop a system that can reliable support multiple dialects at the same time.

A study that comes close is the analysis performed Alsharhan and Ramsay [9], who showed that

training on specific dialects and genders results in better recognition performance on the corresponding dialect and gender than training on multiple dialects and genders at the same time. Additionally, they showed an apparent linguistic difference between the different dialects, especially in Maghrebi Arabic. They speculated that this could be due to the loanwords from other languages, such as French and Spanish, or differences in intelligibility. They also conclude that more data in the general model does not necessarily lead to better performance than the smaller models, leaving the reader with the notion that the best performance is only achieved through specialized models. Their approach focuses on deep neural networks combined with hidden Markov models (DNN-HMM), which leaves the question whether this is also the case for end-to-end DNN methods.

In contrast, recent end-to-end DNN methods show the immense potential of general, multilingual speech recognition models [3, 4]. These networks allow for general speech recognition representation learning as well better generalization through multi-task learning (often with speech translation) and wide data variety with the use of multi-lingual data. This can be motivated back to human phone perception being largely language-agnostic.

The fine-tuning of networks such as Whisper by Radford et al. [3] and USM by Zhang et al. [4] is compared against state-of-the-art (SOTA) systems by Abdelali et al. [5] in a novel Arabic benchmark. They demonstrate that the specialized SOTA system outperforms multilingual zero-shot networks such as USM and Whisper in most Arabic speech datasets, with USM showing the edge in some cases. They mention that even though USM outperforms Whisper in all cases, fine-tuning Whisper on just 2 hours of speech data drastically improves this and comes near to closing this gap. Unfortunately, Abdelali et al. [5] omit fine-tuning Whisper on larger amounts of data or dialect-specific data. The advantage of Whisper is the open-source nature compared to the closed-source, more powerful USM, which was trained on large amounts of unlabelled data. This allows Whisper to be fine-tuned and compared against domain-specific models.

Research on both dialectal diversity using DNNs and fine-tuning multilingual ASR systems on larger amounts of data in the target domain is therefore lacking. This begs the questions proposed in Section 1.1. The definition of the impact of this thesis can thus be described as follows. The effect and extent of shared feature learning from MSA is investigated in the context of dialectal Arabic ASR performance and the level of linguistic similarity between dialects according to a DNN architecture is presented.

Chapter 3

Methodology

3.1 Data

3.1.1 Mozilla Common Voice

Mozilla Common Voice is a widely favored option in the Automatic Speech Recognition (ASR) industry for accessing both high-quality and extensive datasets [10]. The dataset is collected through volunteers writing, recording, and/or validating samples. This ensures a wide variety in the data, while ensuring quality to a certain extent. We decide to use the Arabic partition of Common Voice 16.1[1]. This partition contains mostly MSA samples, as the contributed sentences are expected to be, which is why we can treat it as our MSA dataset. The Common Voice dataset can be conveniently loaded through the HuggingFace Transformers library, allowing for direct access and easy adaptation with the Whisper model [11].

Pre-processing of the dataset involved a sequence of simple steps, outlined in Figure 3.1. First, the audio is resampled from 48 kHz to 16 kHz to accommodate the feature extractor of the Whisper model architecture (see Section 3.2). Afterwards, the log-Mel features are extracted from the audio samples and these are stored. The labels are obtained by tokenizing the transcriptions with the Whisper tokenizer and zero-padding such that all labels have the same fixed dimension.

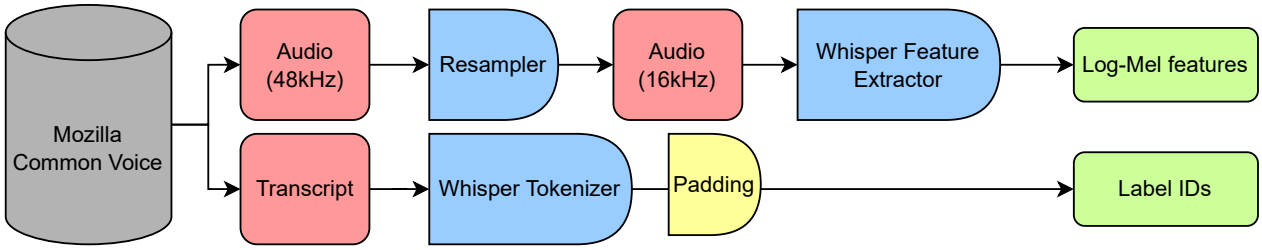

Figure 3.1: Pre-processing pipeline of the Mozilla Common Voice dataset. Red blocks represent the raw data, blue denotes mandatory transformations, while yellow denotes optional transformations. Final labels are depicted in green.

[1]Mozilla Common Voice 16.1 can be found at `https://commonvoice.mozilla.org/en/datasets`

3.1.2 Massive Arabic Speech Corpus (MASC)

The Massive Arabic Speech Corpus (MASC) dataset is another large-scale Arabic ASR dataset [12]. The data is crawled from YouTube videos and consists of over 1,000 hours of speech. The labels are inferred from the subtitles of these videos and subsequently cleaned and normalized. The rich metadata provided on top of this allows for more custom filtering than the Common Voice dataset. Moreover, dialect and region specification is defined in this metadata, which means that the dataset lends itself suitable for the current study. The dialects of interest are the five large dialects utilized by Alsharhan and Ramsay [9], namely Gulf, Levantine, Iraqi, Egyptian, and Maghrebi.

Pre-processing of the MASC data is largely similar to the Common Voice data, but some additional intricacies need to be defined. To start, the dataset has to be segmented from a full audio file and matched with the subtitles of these time-segments. The samples are then saved, with each sample containing the transcription, duration, and the dialect of the segment together with the audio itself. Since the audio data is already 16 kHz, it does not require resampling, but otherwise the pipeline of Figure 3.1 is equally valid for the MASC dataset. Further filtering and data splitting is discussed in Section 4.1.

3.2 Model

The model of choice, as explained previously, is the SOTA and multilingual ASR system Whisper by OpenAI [3]. The reason for using Whisper is its open-sourced nature compared to USM [4], and its impressive multilingual performance. The best model for this comparison is deemed to be the `whisper-small` configuration, as its size is located amidst two smaller and two larger versions. Furthermore, this configuration still has a competitive inference speed on lower-grade hardware, while containing enough parameters to exhibit model flexibility.

The Whisper architecture is displayed in Figure 3.2. It is a Transformer architecture, that takes in a log-Mel spectogram of 80 channels, obtained from 25ms windows with a stride of 10ms [13]. Normalization to $[-1, 1]$ is applied, after which two one-dimensional convolutional layers with a stride of 1 and 2, respectively, a kernel size of 3 and a Gaussian error linear unit (GELU) activation are used to process the encoder input [14]. The encoder input is then appended with sinusoidal positional embeddings and then fed through the encoder blocks of the network. The decoder blocks are provided with learned positional embeddings and have the same dimensions as the encoder blocks. For more details, the reader is directed to Radford et al. [3].

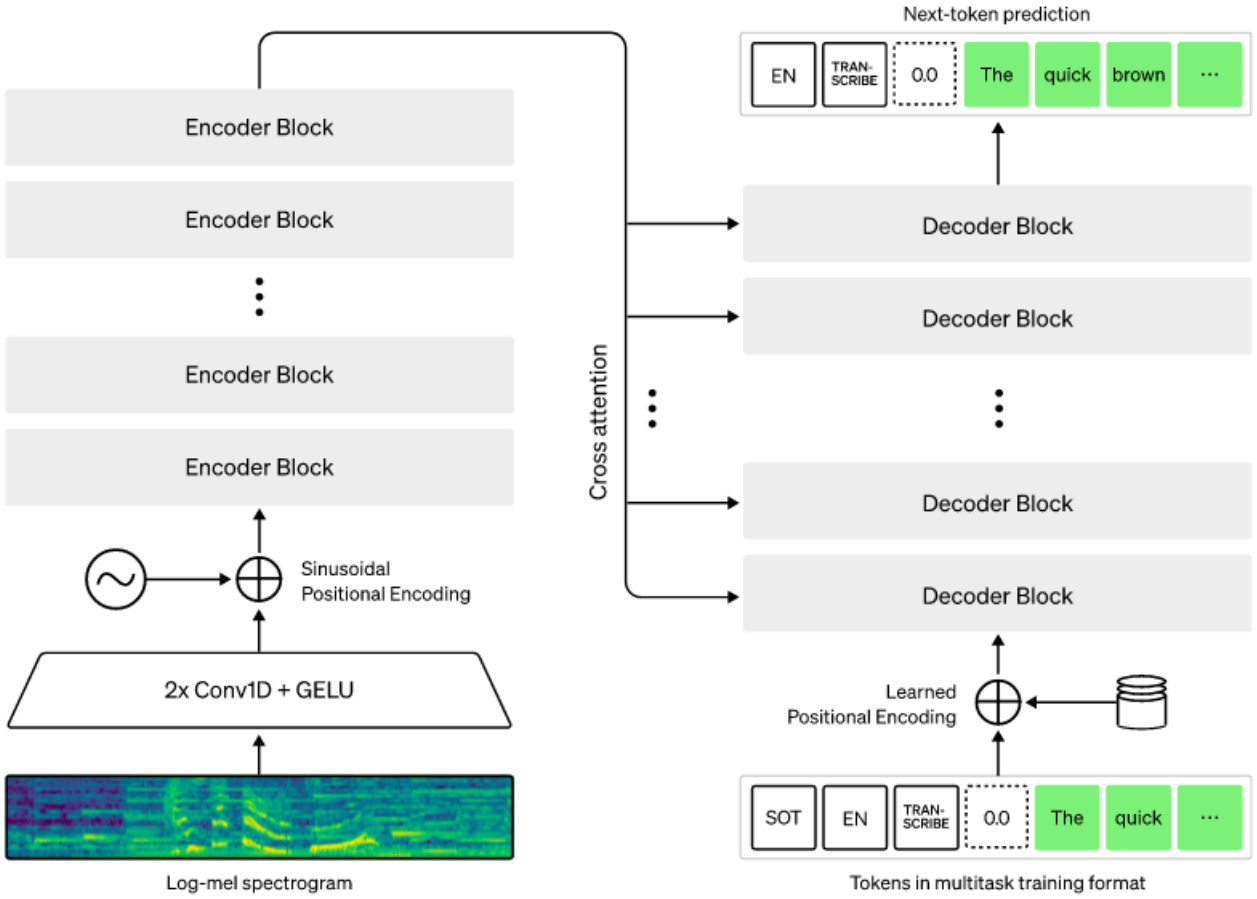

Figure 3.2: An overview of the Whisper architecture, reprinted from Hassan [15].

We utilize two different versions of Whisper in our experiments: the Small model and the Large-v3 model. The Small model consists of 12 layers, with a width of 768. Moreover, they use 12 heads in the multi-headed attention, resulting in a total of 244 million parameters being learnt. The Large-v3 model is an adaptation of the Large model, and contains 32 layers with a width of 1280. It uses 20 heads in the attention instead. The Large-v3 improved upon the Large model by using Mel spectograms with 128 channels instead, which means that the data has to be processed differently for this model. The Large-v3 model has 1550 million learnt parameters.

The weights are obtained through the HuggingFace Transformers library, using the `openai/whisper-small` and `whisper-large-v3` checkpoints. Other details about the model configuration and training parameters are provided in Section 4.

3.3 Tools and Technologies

The fine-tuning and evaluation of the models were performed by NVIDIA A100 GPU nodes on the Hábrók HPC GPU cluster of the University of Groningen. The implementation was done using Python version 3.10.4, PyTorch version 1.12.1, and HuggingFace Transformers version 4.39.3 [16]. The code is available on GitHub at `https://github.com/O-T-O-Z/finetune-ar-dialects`. The models are available on HuggingFace at `https://huggingface.co/collections/otozz/finetune-ar-dialects-664b26ffcf5fd472d7dd1c00`. All additional requirements and dependencies can be found there as well. The Whisper fine-tuning blog post by Sanchit Gandhi from HuggingFace was used as inspiration [17].

3.4 Ethical considerations

Both datasets that were used are available under a Creative Commons license. Participants consented to their data being used for research purposes, and can opt out at any time. Furthermore, the participants are anonymized. Finally, the results are presented as fairly and honestly as possible, with possible mistakes or omissions being out of the hands of the researcher.

Chapter 4

Experimental Setup

To perform an extensive analysis, we take inspiration from Alsharhan and Ramsay [9] and Abdelali et al. [5]. We aim to perform a variety of experiments, while keeping it feasible within the time-frame for this thesis. We will start by examining the different subsets of data used from both Common Voice and MASC in Section 4.1. Then, the metrics will be explained in Section 4.2. Finally, the experiments will be thoroughly described in Sections 4.3, 4.4, and 4.5.

4.1 Data Configuration

4.1.1 Common Voice dataset

The Common Voice dataset is prepared as follows. A split already exists in the original HuggingFace implementation of the dataset. The original "train" and "validation" partitions are merged, totalling 40 hours of speech, then reshuffled, and finally split again with a fixed random seed with a 80:20 ratio into train and validation sets, respectively. The test set contains 13 hours of speech. The data splitting is further visualized in Figure 4.1. The test partition was kept separate until final evaluation, while the validation set was used as overfitting measure.

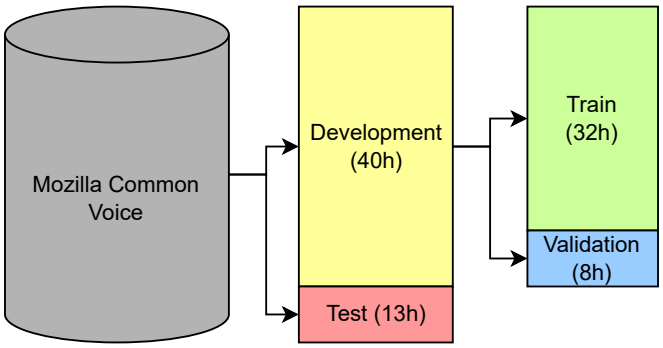

Figure 4.1: Mozilla Common Voice data split.

4.1.2 MASC dataset

The MASC dataset also already contains a train, validation, and test set. We decide to use the training set only since the size is already sufficient and our computational resources are limited. The size of the dialect subsets in this partition are then visualized in Figure 4.2. We observe a large imbalance in the dataset, as the Egyptian and Levantine datasets are clearly overrepresented, while Iraqi and Maghrebi are more limited. Due to this imbalance, we decide to use a maximum of 20 hours of speech per dialect. Dialects with more than 20 hours are randomly shuffled and sampled, while dialects with less than 20 hours are used in full. The result can be observed in Figure 4.3. Here we observe a much better distributed set with all dialects representing about 15-20% of the total set.
From these resampled datasets, we split the data with a 80:20 ratio into train and test with a fixed random seed. This test set is used for final evaluation. From the 80% train partition, we split further into a 80:20 partition of train and validation, respectively, again with a fixed random seed. Finally, all datasets are filtered on transcriptions that are longer than 448 tokens, because of the model configuration. This filtering only removes five to ten samples. The datasets are ported into the HuggingFace `Dataset` format, making the data-loading comparable to the Common Voice data.

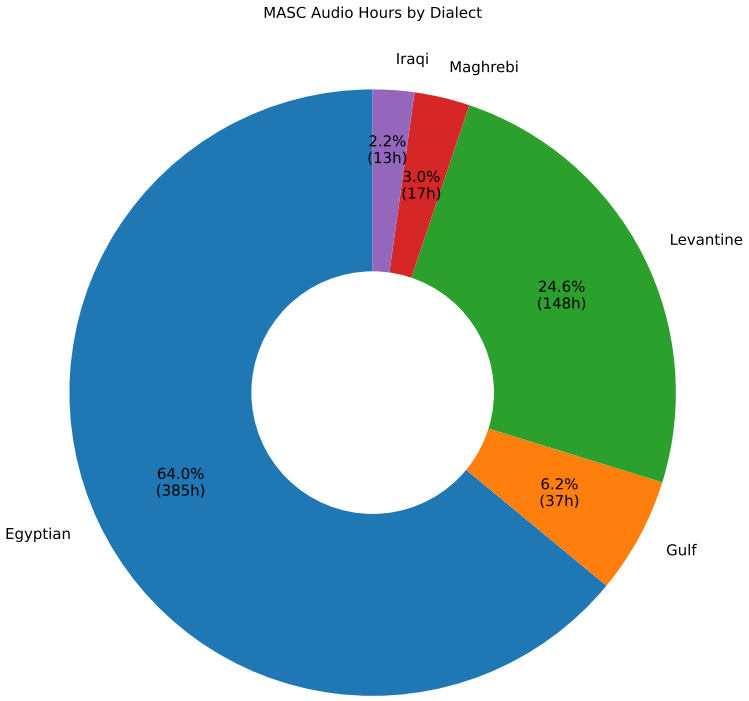

Figure 4.2: The MASC dataset before data balancing. Speech data in number of hours is shown in parentheses.

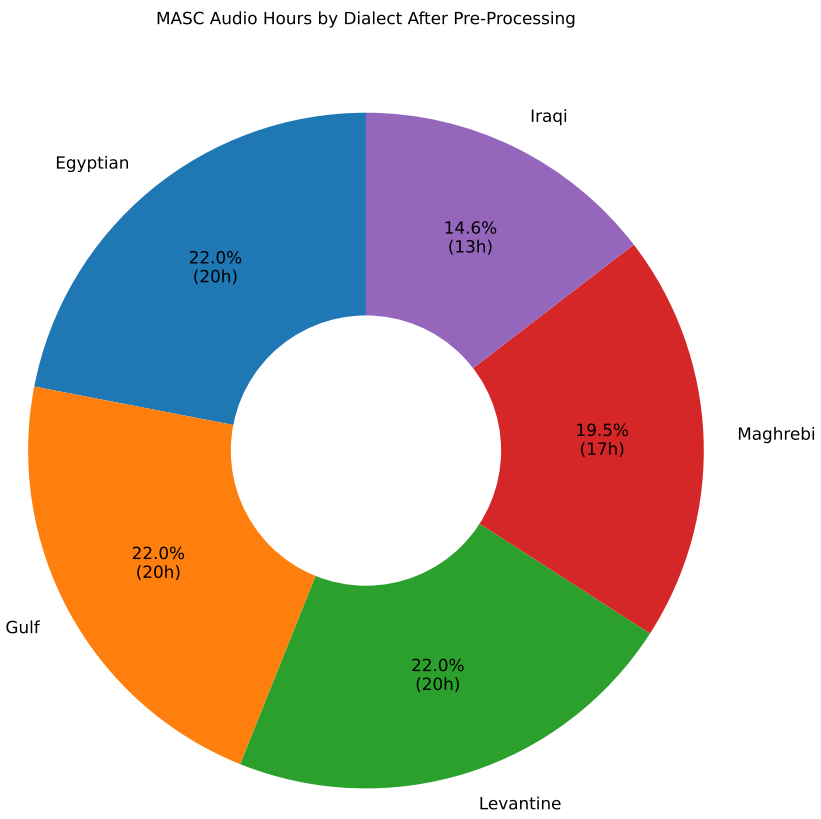

Figure 4.3: The MASC dataset after data balancing. Speech data in number of hours is shown in parentheses. As visible, the data is now almost perfectly balanced.

4.2 Metrics

The metrics of choice are the word error rate (WER) and character error rate (CER). WER is defined as the ratio between the number of modifications or edits and the total number of words, as defined in Equation 4.1.

$$WER = \frac{S_w + I_w + D_w}{N_w} \tag{4.1}$$

S_w denotes the number of substitutions, I_w the number of insertions, D_w the number of deletions, and N_w the number of words in the reference or ground truth. An example that shows how this applies to ASR systems can be found in Figure 4.4. The “an” has been removed and three other words have been substituted. One could also reason how a WER can exceed 100% when more words are inserted, deleted, and/or substituted than there were words in the reference. This is often an indication of “hallucination”, or the model generating nonsensical and exceedingly long output. The lowest WERs for English ASR are currently below 10% [18], but many other languages are higher. Alsharhan and Ramsay [9] and Abdelali et al. [5] also compare performance based on WER since it is an industry-wide benchmark.

Figure 4.4: An example calculation of the word error rate (WER).

A problem that can arise from exclusively reporting WER is that the word can be incorrect based on a single character. Therefore, the character error rate or CER is also evaluated to detect any inconsistencies between the two in analysis. We can define the CER similarly as the ratio between the number of edits and total number of characters in the ground truth. Equation 4.2 states this definition, with N_c now denoting the number of characters in the reference instead.

$$CER = \frac{S_c + I_c + D_c}{N_c} \tag{4.2}$$

4.3 Experiment 1: The effect of training size MSA

In the first experiment, we aim to answer the first research question:

Q1. What is the effect of training data size of Modern Standard Arabic (MSA) on the performance of an ASR system?

To answer this question, we fine-tuned the `whisper-small` checkpoint exclusively with the MSA dataset (Mozilla Common Voice). By fine-tuning the model with 20%, 40%, 60%, 80%, and 100% of the training data, we obtained a course of the effect, with gradual increases denoting the effect at each point. The data subsets are obtained by taking a random subset of the data with a fixed random seed. This results in five separate models being trained.

The models were trained for 5,000 steps with a batch size of 8. AdamW was used as an optimizer with a maximum learning rate set to 1.0×10^{-5} [19, 20]. The hyperparameters β_1 and β_2 of AdamW were kept at 0.9 and 0.99, respectively. Further, a linear learning rate scheduler was used, with 500 warm-up steps. This means that the learning rate increased from 5.0×10^{-7} until the maximum learning rate of 1.0×10^{-5} in the first 500 steps, after which it decreased again. Since this is a fine-tuning task, the learning rate was kept as low as possible to not disturb the original weights of the Whisper model. The training metrics were logged every 25 steps, while evaluation was performed every 1,000 steps, after which a model checkpoint was saved as well. The loss function used is cross-entropy loss, see Equation 4.3. The training process was followed using Tensorboard.

$$\mathcal{L}(p,q) = -\sum_{x \in C} p(x) \log q(x) \tag{4.3}$$

The training can be observed in Figure 4.5. The train and validation loss decrease together, after which the validation loss decreases less or converges. In the case of 20% training size, there seems to be a slight overfit, but it is insufficient to consider it detrimental. Early stopping could have been employed in hindsight as a regularization method to cut training time for most models.

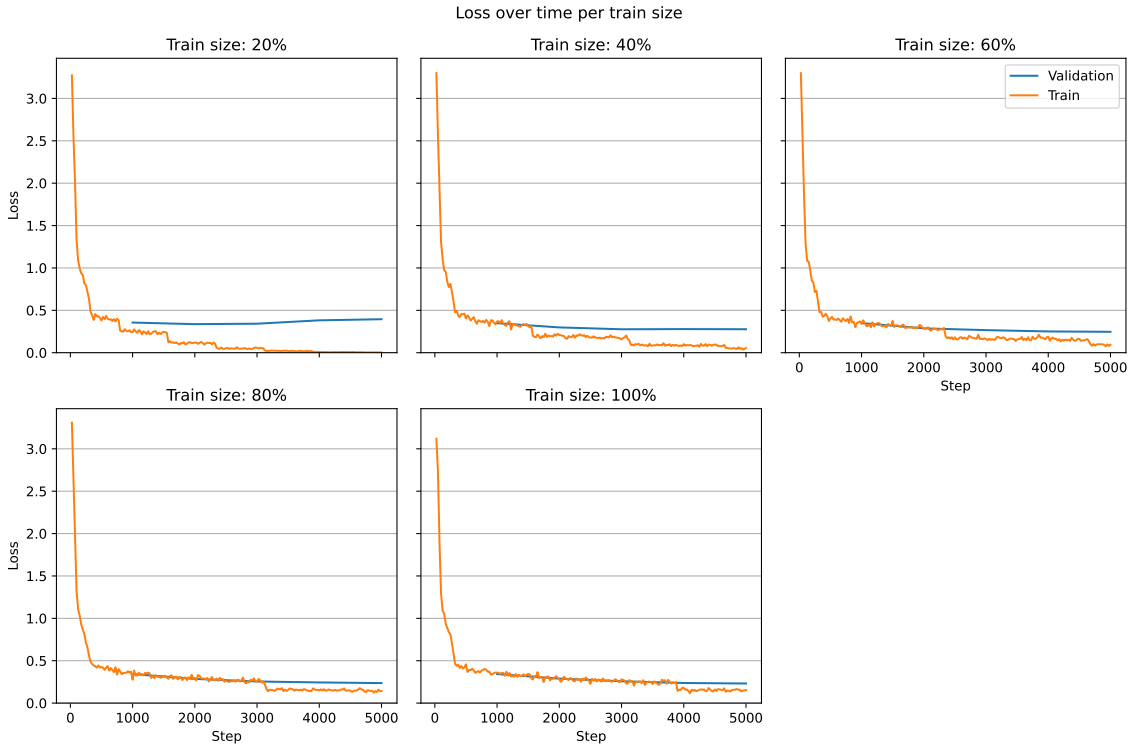

Figure 4.5: Training performance as a loss over time of the training data size experiments. The first 1,000 steps no evaluation was performed, which is why the validation line starts at 1,000.

4.3.1 Evaluation

The trained models were subsequently evaluated and compared with the non fine-tuned `whisper-small` and `whisper-large-v3` on performance. It is especially interesting to compare against both models, since we expect that the smaller model that is fine-tuned will perform equal or better than the larger model, which could have implications on industry-grade applications. If a smaller, fine-tuned model can be used in production, for instance, instead of a larger model, it would be beneficial both for hosting the models on cloud infrastructure and in terms of inference speed.

The performance metrics were once again defined as WER and CER. The test set of the MSA data was deemed sufficient for this comparison as no dialectal fine-tuning has been performed. Based on the results of this first experiment, an analysis will be performed to determine the right amount of MSA data such that no unnecessarily high computational load is required to train the model in later studies. The research question can simultaneously be answered on the basis of the highest performance or lowest WER and CER compared to the amount of data used.

4.4 Experiment 2: Comparison with and without pre-training

Then, we aim to answer the second research question:

Q2. Is pre-training on MSA beneficial for the performance of dialectal Arabic fine-tuning?

In order to test this, we fine-tune both the MSA fine-tuned (defined as pre-trained) from Experiment 1 (Section 4.3) as well as the `whisper-small` from scratch. By comparing training from scratch with a pre-trained configuration, we aim to investigate whether pre-training is even necessary. This results in two models per dialect and thus 10 models in total. The training parameters are kept identical from Experiment 1.

The training processes are displayed in Figure 4.6 and Figure 4.7. As expected, the pre-trained models display lower initial loss. Interestingly, however, the loss for both with and without pre-training seems to converge to similar points. Again, the training converges quite rapidly for all 10 models, with no serious signs of overfitting. Iraqi has the most trouble, which could be due to the smaller size of the training data (see Figure 4.3). Since the training shows minimal decline in loss, it leads us to believe that comparing performance with the `whisper-small` checkpoint will be beneficial.

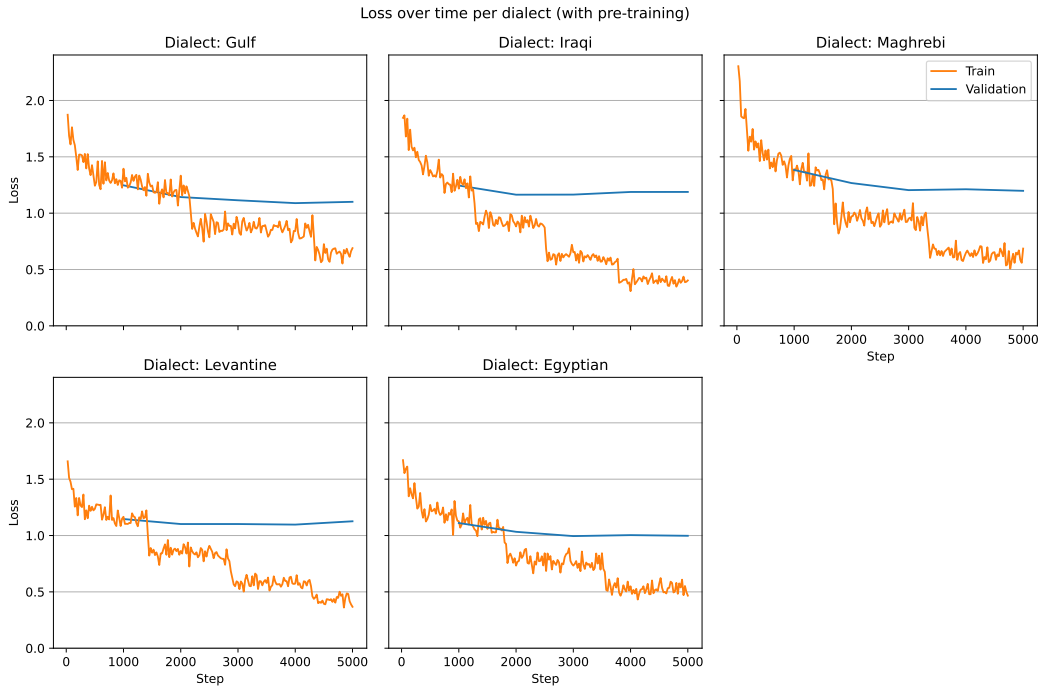

Figure 4.6: Training performance as a loss over time of the models trained on top of the MSA pre-trained model.

4.4.1 Evaluation

For this experiment, it is essential to compare performance both on MSA and dialectal Arabic test sets to investigate the effect over the entire spectrum. This ensures that the full picture is obtained

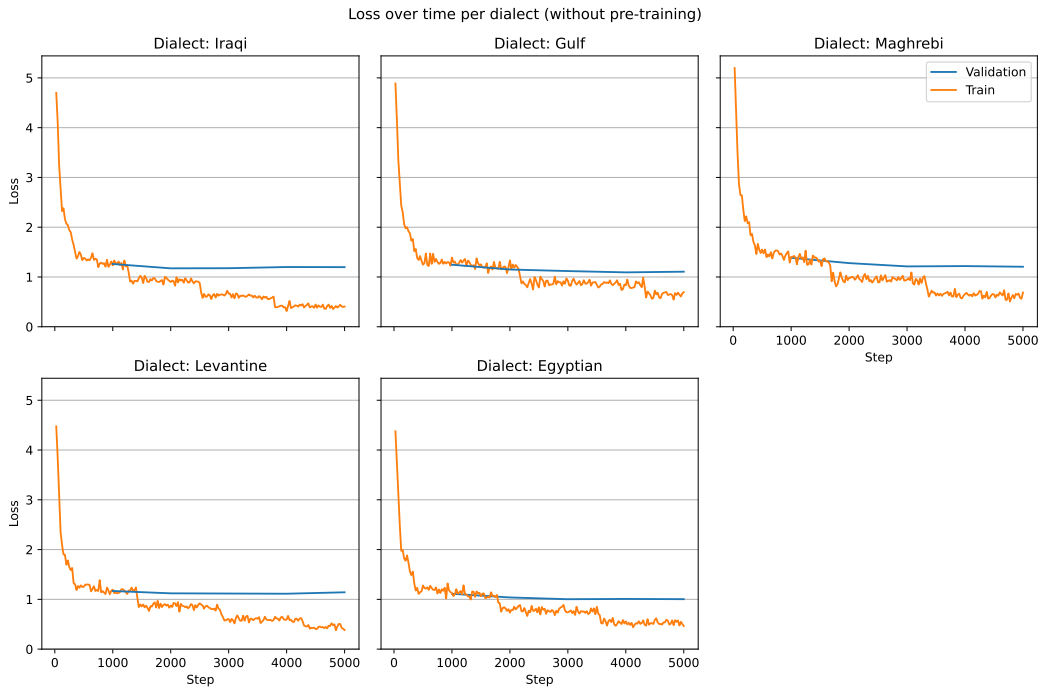

Figure 4.7: Training performance as a loss over time of the models that did not utilize the MSA pre-training and were trained on top of `whisper-small`.

and the research question can be answered as well as possible. The metrics for performance are defined as WER and CER and the models fine-tuned on the pre-trained model are compared against the models fine-tuned on the `whisper-small` checkpoint with a paired statistical test. Since we want to ensure validity, we do not assume parametric assumptions, and as the pairs are matched, we opt for a Wilcoxon signed-rank test.

4.5 Experiment 3: Dialectal Arabic fine-tuning

The last experiment is concerned with answering the last research question:

> Q3. What is the difference in performance between dialect-specific and dialect-pooled models in terms of dialectal ASR?

To this extent, the fine-tuned models both with and without pre-training from Experiment 2 (see Section 4.4) are compared with two models (pre-trained and from scratch) trained on all dialectal train sets.

These training procedures can be found in Figure 4.8. The pre-training does show a benefit when first starting out training as a large drop in the start of the training is not required in contrast to training without pre-training. Furthermore, the loss generally ends up converging at a slightly lower point for the pre-trained model. Overfitting is not visible in either, which could be due to the larger size of the pooled dialect data compared to the separate dialect datasets.

Figure 4.8: Training performance as a loss over time of the models trained with all dialects pooled into one dialectal dataset. The model trained on top of the MSA pre-trained model is displayed on the left, while the model that did not utilize the pre-training configuration is shown on the right.

An additional investigation that will be performed here is the possible similarity between dialects based on the cross-dialect performance of each model. The training parameters for the dialect-pooled models are again identical to the set defined in Experiment 1 (Section 4.3).

4.5.1 Evaluation

The dialect-specific and dialect-pooled models will be compared on the basis of WER and CER. The evaluation will be performed on all test sets again (MSA and dialectal) to visualize the full performance of both. For the additional linguistic difference analysis, we will visualize and discuss a confusion matrix.

Chapter 5

Results

In the current chapter, we will investigate the results for each experiment. This includes an analysis of the fine-tuning process of each model, model performance, particularities, and comparisons.

5.1 Experiment 1: The effect of training size MSA

The goal of the first experiment is to visualize the effect of the MSA training data size on the performance of Whisper. As we can observe in Figure 5.1, the expected gradual decrease in both WER and CER can be recognized. This decrease is most noticeable when moving from `whisper-small` to the model fine-tuned on just 20% of the MSA training dataset. There is an improvement of almost 10% visible in both WER and CER.

Furthermore, more data does not necessarily lead to much better performance, although the performance keeps improving slightly with every step of 20% extra data. Most notably, the difference between fine-tuning on the full MSA dataset and the non-fine-tuned `whisper-large-v3` checkpoint is almost negligible, while the latter is known to exhibit a considerably higher inference time than `whisper-small`.

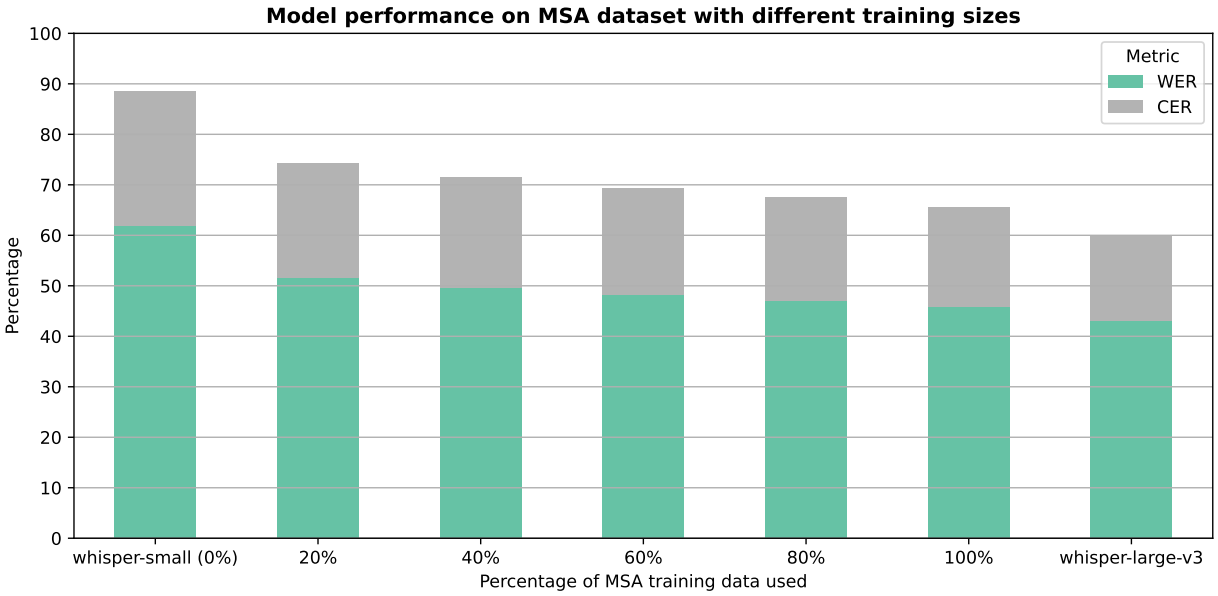

Figure 5.1: Barplot showing model performance over different training data sizes in both WER and CER. The Small and Large Whisper models are also shown for comparison

5.2 Experiment 2: Comparison with and without pre-training

The results of the remaining 12 models can be found in Figures 5.2 and 5.3. Additional results are found in Table 1 and Figure 1 of Appendix A. For this experiment, we are interested in the results of the models with- and without pre-training. This comparison can be done in multiple ways. We could compare the results on the test sets based on all models trained with- and without pre-training, which is displayed in Figure 5.2. However, it might also be fruitful to check differences per model on each test set, which is presented in Figure 5.3 and Figure 1 of Appendix A.

Starting with Figure 5.2, we do not observe large differences between pre-training and no pre-training, as most results are close to each other in terms of WER. The sole exception seems the MSA test set, where a large difference is visible. To validate our findings, we performed multiple Wilcoxon signed-rank tests, each on the differences between test set performances. The results can be observed in Table 5.1. As expected, the only statistically significant results ($Z = 0.00$, $p < 0.05$) is found for the performance of all trained models on the MSA test set. Thus, we find that the pre-trained models ($\mu = 58.66$, sd $= 3.59$) perform better than the models without pre-training ($\mu = 66.98$, sd $= 5.19$) on our MSA test set. This should come with no surprise, however, since the pre-trained model has been pre-trained on the train partition of the same MSA set. Still, it shows that the models do not exhibit catastrophic forgetting of weights. Some forgetting is definitely present, since Section 5.1 showed a 45.84% WER on the pre-trained model *before* fine-tuning.

Furthermore, we were interested if there was a difference between the two groups of models as a whole. To this extent, we performed another Wilcoxon signed-rank test comparing all results of the two groups. We found there to be no significant difference ($Z = 269.00$, $p = 0.32$) in WER performance between all pre-trained models ($\mu = 84.55$, sd $= 15.23$) and all non pre-trained models ($\mu = 85.50$, sd $= 11.86$).

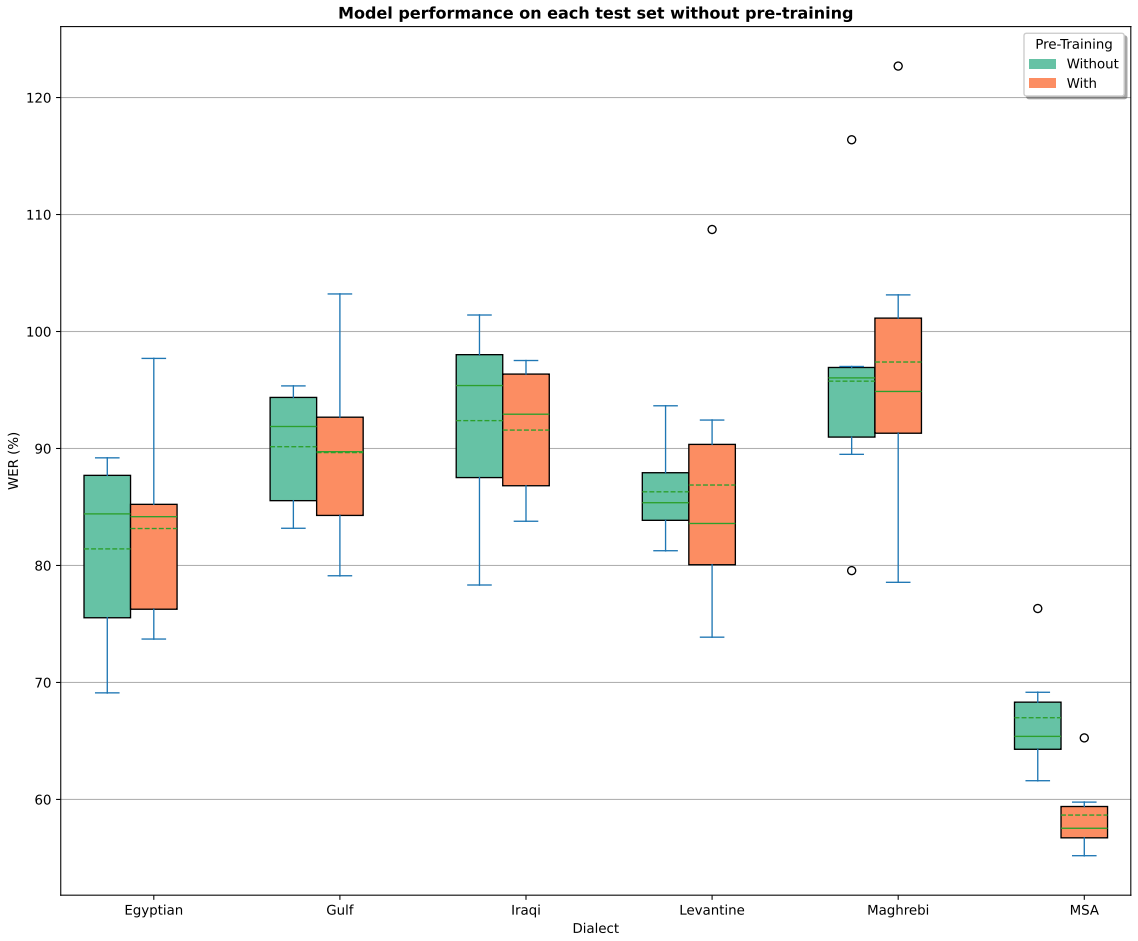

Figure 5.2: Boxplot showing the differences between pre-trained and non pre-trained configurations over all test sets.

Table 5.1: Wilcoxon signed-rank test results comparing the performance of models with and without pre-training. Statistically significant results ($p < 0.05$) are denoted by *.

	% WER without pre-training (↓)	% WER with pre-training (↓)	Z-statistic	p-value
Egyptian	81.42 ± 8.46	83.16 ± 8.88	7.00	0.56
Gulf	90.15 ± 5.52	89.65 ± 8.44	9.00	0.84
Iraqi	92.38 ± 8.8	91.57 ± 5.97	7.00	0.56
Levantine	86.3 ± 4.36	86.88 ± 12.33	9.00	0.84
Maghrebi	95.76 ± 12.08	97.39 ± 14.78	10.00	1.00
MSA	66.98 ± 5.19	58.66 ± 3.59	0.00	0.03*

Finally, some things could be noted from Figure 5.3. All dialectal models seem to display similar results both with and without pre-training. The only visual exceptions seem to be the models fine-tuned on Iraqi and Levantine. The former generally seems to perform better when fine-tuned on the pre-trained model, while the latter shows better performance overall when fine-tuned on the non pre-

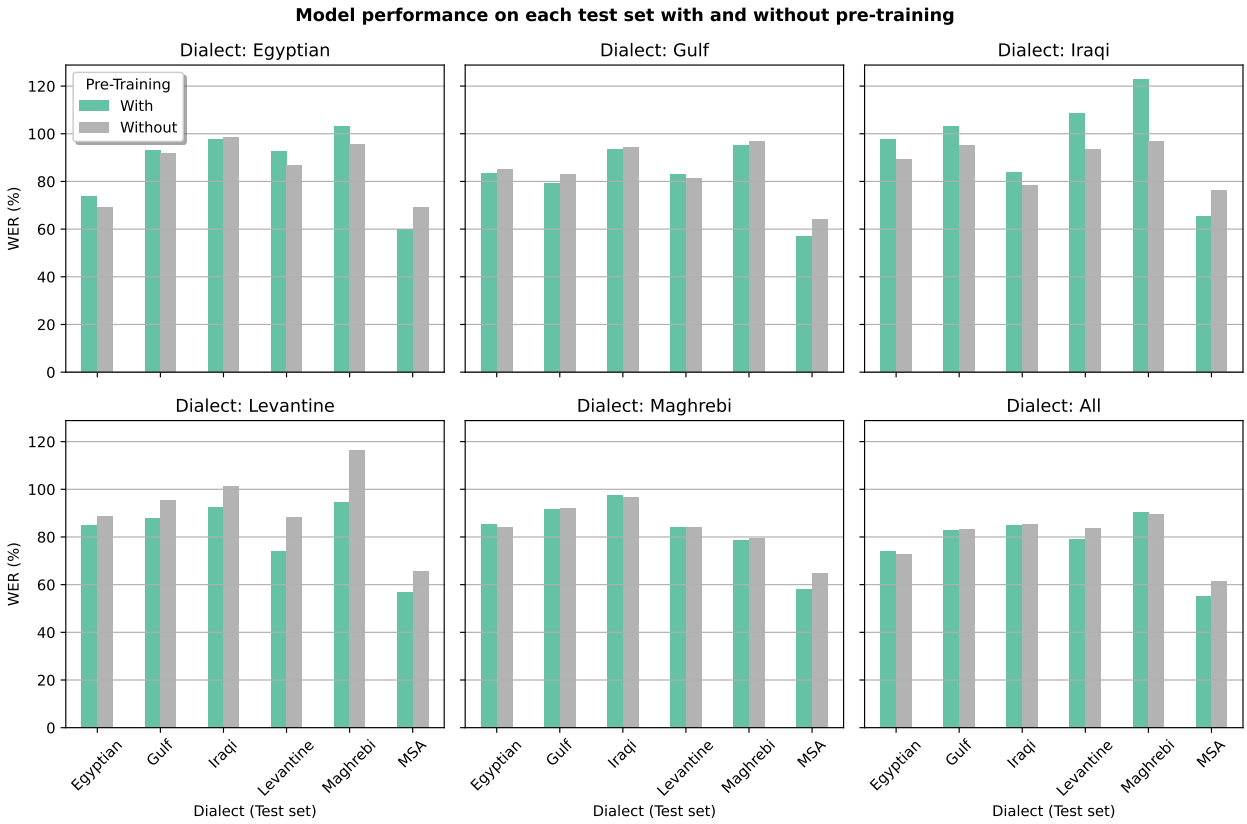

Figure 5.3: Pre-trained versus non pre-trained WER performance per trained model (plot title) and per test set (x-axis).

trained configuration. The CER results are available in Figure 1 of Appendix A and are generally consistent with the results in Figure 5.3.

5.3 Experiment 3: Dialectal Arabic fine-tuning

The final experiment is concerned about the dialect-specific and dialect-pooled results. Some of these results were already presented in Section 5.2, yet now we will focus on comparison of the models fine-tuned on the "All" subset and the models fine-tuned on the specific dialect training sets. The "All" subset is hereafter referred to as the "dialect-pooled" dataset.

For the difference in performance, we focus on the performance of the pre-trained models per dialectal training set, displayed in Figure 5.4. Firstly, the performance of `whisper-small` is the worst, which was expected from Experiment 1 (Section 5.1). Then, we can also find that the models trained on a specific dialectal dataset generally work second-best on their corresponding test set (indicated by the black hatches), with their best performance on the MSA test set.

The model fine-tuned on the dialect-pooled training set seems to perform close to the models fine-tuned on each dialect specifically. This can be observed in more detail in Figure 5.5, Table 1 of Appendix A, and Table 2 of Appendix B, where the dialect-pooled model's result per testing set is comparable to the corresponding dialect-specific model's result. In Figure 5.4, we can observe the same behaviour when comparing the performance on each test set for the dialect-pooled model with the hatched bars of the dialect-specific models. The mean absolute difference (MAD) for the WER between the dialect-pooled and dialect-specific models is 5.24%, while the MAD for the CER is 4.48%. Furthermore, the model trained on the dialect-pooled dataset outperforms all other models on MSA (except the non fine-tuned model from Experiment 1) and thus displays the best overall performance.

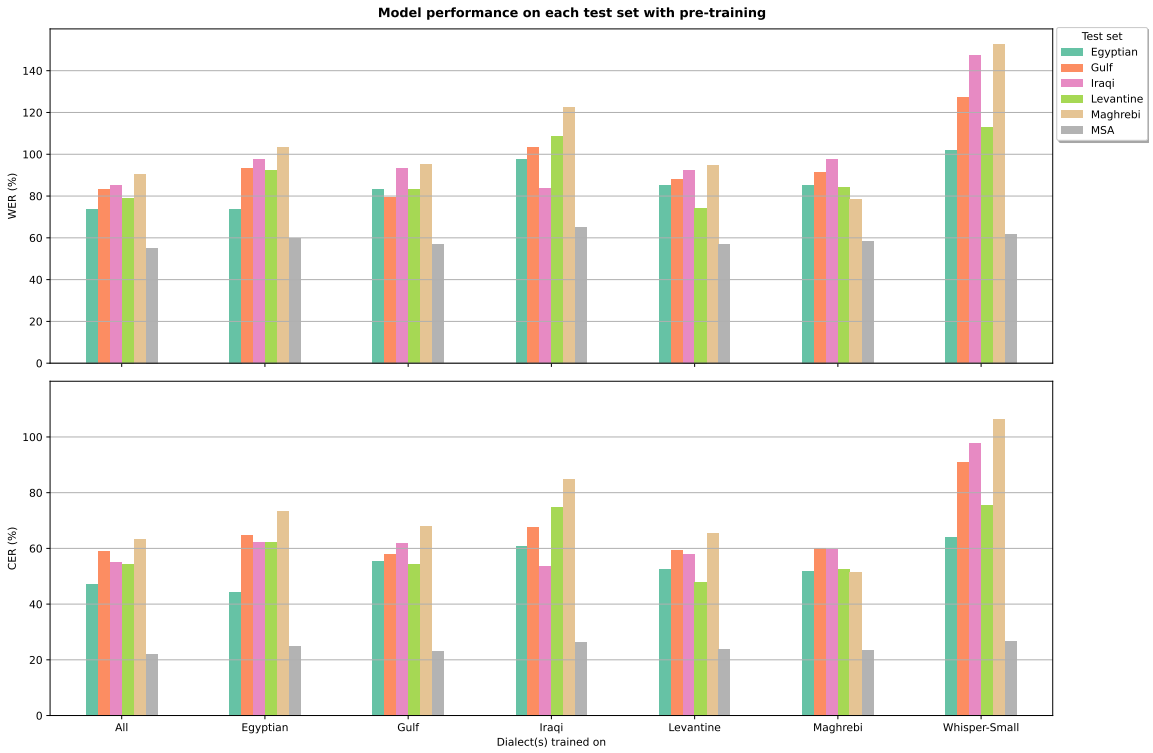

Figure 5.4: Model performance in WER and CER of the pre-trained configurations on each test set per model. Hatched bars denote the test set of the same dialect as the train set.

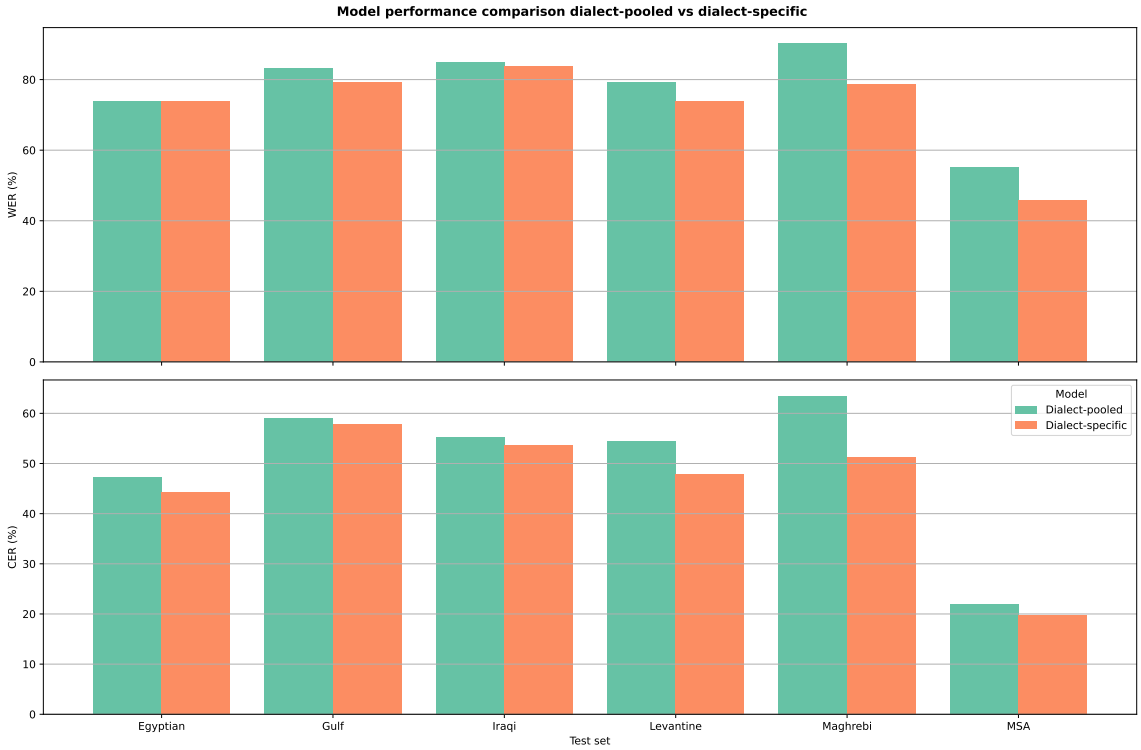

Figure 5.5: Comparison between dialect-pooled and dialect-specific models on all dialectal test sets. The dialect-specific model for MSA is the pre-trained model from Experiment 1.

Finally, we are interested in the possible linguistic distance between the different dialects, which is visualized in Figure 5.6. We are focusing on the WER performance of the pre-trained configurations, but the reader is invited to investigate Figures 3, 4, and 5 of Appendix B. As expected, the diagonal, i.e. the same dialect used for training as for testing, performs best for each dialect. Firstly, we find that fine-tuning on the Egyptian or Iraqi dialects worsens performance on other dialects the most (rows). In contrast, fine-tuning on other dialects seems to work best on the Egyptian and Levantine test sets (columns).

The two dialects that appear to be the most different from the others are Iraqi and Maghrebi (columns), since they show the worst performance when trained on other dialects than their own. However, interestingly, the Maghrebi fine-tuned model performs comparably on the other dialect test sets (row). The two dialects that show the worst performance on each other's test set are the Iraqi and Maghrebi dialects as well, while the Gulf and Levantine dialects show high and similar performance on each other's test set.

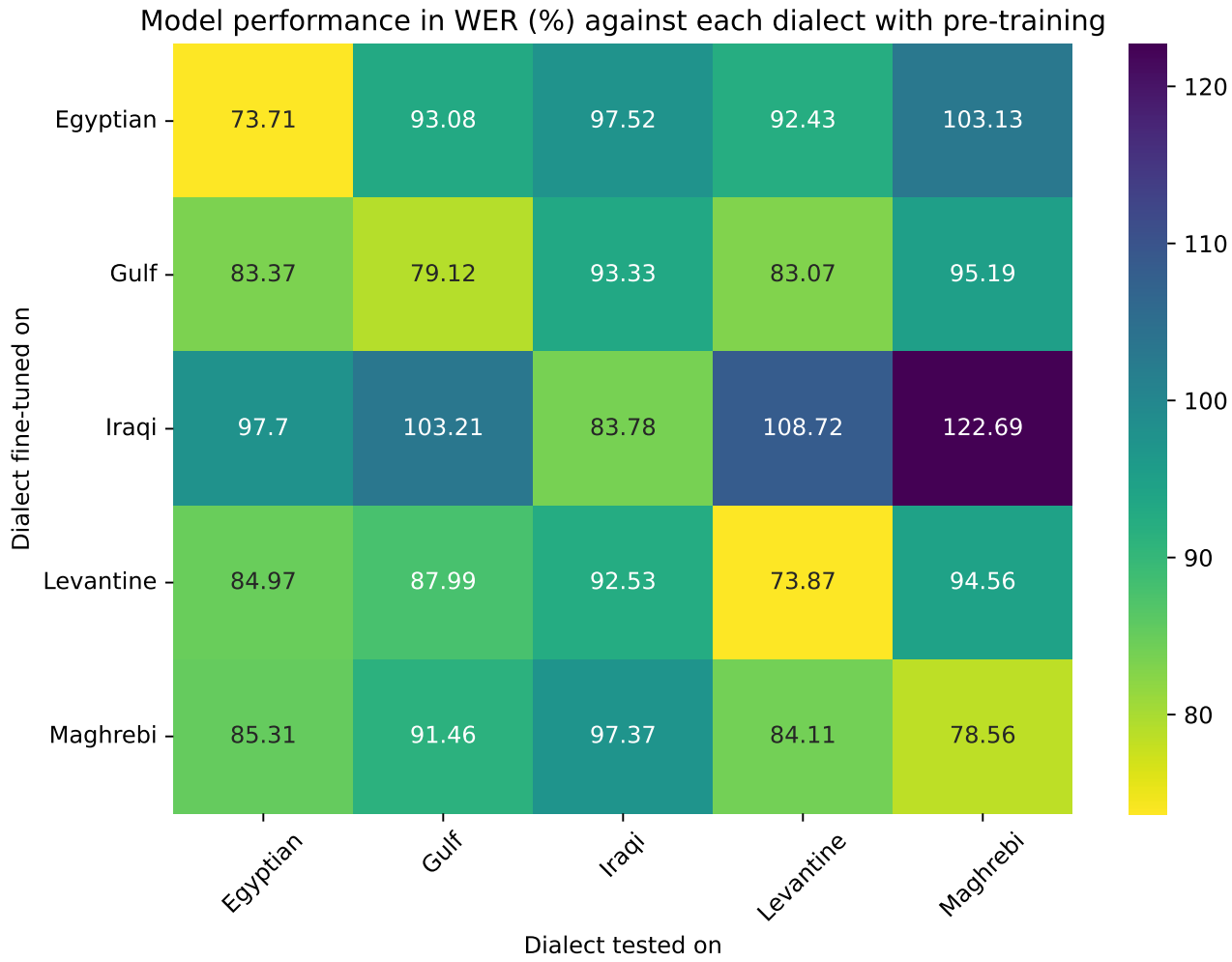

Figure 5.6: Confusion matrix of WER performance between the dialectal train and test sets. X-axis denotes the train dataset, while the test set is shown on the y-axis. Models are pre-trained on MSA.

Chapter 6

Discussion

In light of the results presented in the previous sections, the effects of training size and pre-training have been identified, and the difference between dialect-specific and dialect-pooled models has become clear. The hypotheses from Section 1.1 will now be discussed and validated.

6.1 Effect of training size MSA

We expected a positive effect of the amount of MSA pre-training data on the performance of the ASR system, to a limited extent. The results in Section 5.1 demonstrate that the improvements in performance are most pronounced in the first step from no pre-training to 20% training data. After that, another 80% is required to improve the model by around 6% (see Figure 5.1). This validates our hypothesis and leads us to believe that even small amounts of training data can be very beneficial for the performance of Whisper on MSA.

The most attractive outcome of this experiment is the comparable performance of the fine-tuned `whisper-small` model checkpoint with the much larger `whisper-large-v3` checkpoint in the automatic speech recognition performance in Modern Standard Arabic. Even though the latter has access to higher model complexity, the former is able to specialize in the specific task of MSA ASR with less weights and some fine-tuning (a maximum of 32 hours). Inference time and computational cost can therefore be reduced heavily by accepting a negligible drop in WER performance [3].

Furthermore, our findings are largely in line with Alsharhan and Ramsay [9], who found an increase in model performance with more data up to a certain extent. Even though they performed this similar experiment with multi-dialectal data, the conclusion is equivalent. After a while, their model began to decrease in performance again, which is something we also experience slightly in the subsequent experiments when training the dialectal models on top of the pre-training. Another interesting conclusion they note is that the training data should be drawn from the same population rather than simply increasing the size of the training data. This is also apparent in our subsequent experiments where the MSA performance starts to drop after performing additional fine-tuning on dialectal data, as experiment 2 (Section 5.2) points out. We suspect this is due to the model "forgetting" or re-calibrating its weights to try to accommodate both what it learnt previously and the new information.

6.2 Effect of pre-training

The effect of pre-training was slightly less pronounced than expected, however. We hypothesized a significant enhancement across the board for the pre-trained models compared to the non pre-trained models. In Section 5.2, the results exhibit some minor differences in WER performance between the results on each dialectal test set (Figure 5.2). Yet, the only test set demonstrating a statistically significant difference proved to be the MSA test set. Although this difference is obvious, since the models had been fine-tuned on MSA as well, it still indicates that the models manage to retain some of the knowledge that was previously obtained.

Some dialects showed mixed results, but due to the lack of statistical significance, these are disregarded. Figure 5.3 of Section 5.2 and Figure 1 of Appendix A show very comparable results when comparing the performances of the pre-trained and non pre-trained models on the dialectal test sets.

We hypothesized a more pronounced improvement because we expected the pre-trained network to generalise and identify linguistic features that are common across all variations of Arabic better than Whisper without pre-training. One possible reason could be that the variations between dialects and between each dialect and MSA are too vast to generalise. This could be due to the limited number of common words or sentence structures between the two. Alternatively, since Whisper has already been trained on multiple languages, this type of information could already be present in the network. Finally, we used the smaller checkpoint of Whisper, which has limited capacity to learn compared to the larger sizes. It would be interesting to investigate this effect for other model sizes as well.

6.3 Difference between dialect-specific and dialect-pooled models

To conclude, we expected that the dialect-specific models would outperform the dialect-pooled models on their trained dialect, since the review by Alsayadi et al. [1] mentioned this as a challenge. We did not find conclusive evidence, however, as the two dialect-pooled models performed very close to the dialect-specific models per dialectal test set. The mean absolute difference was only around 5% in both WER and CER, which indicates that only a bit of performance is sacrificed when opting for a dialect-pooled approach. Better yet, when considering the improved performance on the other dialects that could be considered out-of-distribution for the dialect-specific models, the dialect-pooled models show much potential in overall performance. These results are also in line with Alsharhan and Ramsay [9], who found a difference of roughly 2%. Thus, we have reason to believe that the challenges of insufficiency and lack of variation can be reduced through pooling the dialectal datasets. The fear of introducing of imbalance can be easily mitigated by ensuring data balancing before training, as we demonstrated in Section 4.1.

On the other hand, the initial aim of this thesis was to propose a solution to some of the challenges outlaid by Alsayadi et al. [1]. One of them included the difficulties imposed by the low-resource nature of the dialectal datasets. As Figure 5.4 showed, using a dialectal dataset of up to 20 hours can already reduce the WER per dialect by a large amount. Thus, by starting with a pre-trained ASR system such as Whisper, one reduces the data requirements significantly.

A final issue that Alsayadi et al. [1] requested to be addressed was the limited use of dialectal Arabic in training ASR systems. As our results also showed (see Figure 5.4 and Figure 2 of Appendix A), `whisper-small` as trained by OpenAI performs competitively on MSA but strongly lacks in performance on dialect-specific datasets [3]. By fine-tuning with dialects as an addition to MSA, we improved the performance of the same architecture on dialectal datasets.

Another advantage of using the dialectal speech datasets for training, is the ability it provided us to investigate the linguistic differences, similar to Alsharhan and Ramsay [9]. In Figure 5.6, we found that fine-tuning on Egyptian and Iraqi datasets decreases performance on other dialects. This is not in line with Alsharhan and Ramsay [9], who report that Egyptian is similar to other Arabic dialects, while Maghrebi performs the worst instead of Iraqi. Even so, we also found that training on other dialects showed poor performance on the Maghrebi test set, while training on the Maghrebi train set results in decent performance on other test sets. Furthermore, we argue that the largest difference between Iraqi and Maghrebi could make sense due to the large geographical distance between the two regions. Conversely, Levantine and Gulf are close both geographically and in terms of cross-dialect performance. Still, these result cannot be taken as a linguistic truth as a linguistic study would be more suited for this purpose. The current thesis simply investigated the similarities based on how the models performed.

6.4 Limitations

Even though we desired to present a reasonable comparison between the results we obtained and the results obtained by previous studies, we generally found much higher WERs. This would have been put into more context if we had access to the weights and models of the studies we compared to, so presenting the performance of fine-tuned Whisper relative to, for example, the DNN-HMMs of Alsharhan and Ramsay [9] would have been more interesting.

Further, the models trained in this study did not show much improvement over the training time, with the validation quickly converging and staying stable after convergence, as visible in Experiments 4.3, 4.4, and 4.5. Further hyper-parameter tuning would have been ideal, together with cross-validation. Yet, the training time required for fine-tuning seventeen different models and the subsequent evaluation of analysis proved it difficult to optimize each model individually.

Lastly, if we trained multiple models on different random splits of the data, we could have obtained a better statistical comparison of the effect of training size and pre-training on the performance of the models. Furthermore, we would have been able to investigate the extent to which dialect-pooled models performed better or worse than dialect-specific models in light of a statistical test as well.

Chapter 7

Conclusion

In the current and last section, the main contributions of the thesis will be summarized, together with the presentation of possible future directions of this research and its impact and relevance.

7.1 Summary of the main contributions

All in all, the current thesis first demonstrated the effect of dataset size on the word error rate performance of Modern Standard Arabic when fine-tuned on Whisper [3]. We showed that even a small amount of fine-tuning can lead to large increase in performance. Then, we found that the relation between training data size and WER performance is exponential and eventually stagnates with large amounts of data, which is in accordance with literature [9]. We also discovered that fine-tuning `whisper-small` can lead to similar performance as `whisper-large-v3` for MSA at a fraction of the inference time.

Next, we found that pre-training on MSA to subsequently train on Arabic dialects generally did not show a significant positive effect. This leads us to believe that the differences between the Arabic dialects are large and that MSA does not seem to share inter-dialectal information that is relevant for a Transformer architecture such as Whisper. However, pre-training on MSA did show a significant increase of performance when testing on a MSA test set. Thus, to retain competitive MSA performance, pre-training can still prove to be fruitful.

Then, we concluded that dialect-specific models only slightly outperform dialect-pooled models in terms of dialectal WER performance. With the generally better performance displayed by dialect-pooled models, it seems beneficial to combine dialectal datasets for a more robust model due to the larger size and increased variation of its training dataset. This is also in agreement with the results of Alsharhan and Ramsay [9], even though they concluded the opposite. Combined with pre-trained multi-lingual models such Whisper, we proposed a possible new direction of training low-resource automatic speech recognition systems.

Finally, we showed some large performance differences between the Arabic dialect-specific models. Dialects that are geographically further apart showed larger differences, such as the Maghrebi (North-African) and Iraqi dialects, while regions that are closer to each other showed more similar performances, for instance the Levantine (Jordan, Palestine) and Gulf (UAE, Saudi Arabia) dialects.

7.2 Future work

In terms of future directions, we would be interested in performing the same type of analysis on DNN-HMM systems and other deep learning approaches, such as the Universal Speech Model [4]. By doing so, we could provide a fairer comparison to Alsharhan and Ramsay [9] and visualize the effects to a more complete extent.

It also would be fascinating to investigate the actual linguistic differences that the fine-tuned models pick up on, leading them to perform better or worse on specific dialects. Together with a linguistic analysis, it would reveal a lot about a machine learning model's ability to pick up on these small dialectal differences.

As mentioned previously in the limitations, optimizations and cross-validation of each model would certainly be a good improvement on the current work. The analysis done here could be considered a preliminary proof-of-concept as the limited time available for this project led to the proposed research questions and subsequently to the scope of the current thesis. Although we were able to answer the research questions, many more are left.

7.3 Impact and relevance

This research has multiple implications and henceforth has an impact on the dialectal Arabic as well the low-resource ASR community as a whole.

To start, we demonstrated that we could achieve competitive performance with a much smaller, fine-tuned model compared to a large multi-lingual model. This could have a profound impact on industry-grade applications of (Arabic) ASR, as the deployment of smaller models is obviously preferred to deploying large, computationally expensive, and slow models. By reducing the computational burden, costs can be reduced on the side of the producer and the response time can be lowered on the client's side.

Additionally, we found that pre-training can prove to be beneficial to retain decent MSA performance, while it might be generally unnecessary. Training dialectal models can thus be done in a quicker fashion. On top of that, we also showed comparable performance of dialect-pooled models to dialect-specific models. This could be proposed as a solution to the lack of variation and data present in this sub-field of ASR as well as low-resource language ASR systems in general. By pooling similar dialects together, one could reduce computational burden, while generalization takes care in the case that there do prove to be more similarities than the Arabic dialects did in the current study. Examples could be cross-continental Spanish or Portuguese.

In short, we investigated the effect of multiple training configurations and found crucial implications that could lead to better decision-making in the fine-tuning of ASR models. Regardless, researchers could always perform a similar analysis, which means that this research can be taken as an example setup to replicate.

Bibliography

[1] H. A. Alsayadi, A. A. Abdelhamid, I. Hegazy, B. Alotaibi, and Z. T. Fayed, "Deep Investigation of the Recent Advances in Dialectal Arabic Speech Recognition," *IEEE Access*, vol. 10, pp. 57 063–57 079, 2022, conference Name: IEEE Access. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9780142

[2] D. Vergyri and K. Kirchhoff, "Automatic diacritization of arabic for acoustic modeling in speech recognition," in *Proceedings of the workshop on computational approaches to Arabic script-based languages*, 2004, pp. 66–73.

[3] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, and I. Sutskever, "Robust Speech Recognition via Large-Scale Weak Supervision," in *International Conference on Machine Learning*. PMLR, Jul. 2023, pp. 28 492–28 518, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v202/radford23a.html

[4] Y. Zhang, W. Han, J. Qin, Y. Wang, A. Bapna, Z. Chen, N. Chen, B. Li, V. Axelrod, G. Wang, Z. Meng, K. Hu, A. Rosenberg, R. Prabhavalkar, D. S. Park, P. Haghani, J. Riesa, G. Perng, H. Soltau, T. Strohman, B. Ramabhadran, T. Sainath, P. Moreno, C.-C. Chiu, J. Schalkwyk, F. Beaufays, and Y. Wu, "Google USM: Scaling Automatic Speech Recognition Beyond 100 Languages," Mar. 2023. [Online]. Available: https://arxiv.org/abs/2303.01037v3

[5] A. Abdelali, H. Mubarak, S. Chowdhury, M. Hasanain, B. Mousi, S. Boughorbel, S. Abdaljalil, Y. E. Kheir, D. Izham, F. Dalvi, M. Hawasly, N. Nazar, Y. Elshahawy, A. Ali, N. Durrani, N. Milić-Frayling, and F. Alam, "LAraBench: Benchmarking Arabic AI with Large Language Models," Mar. 2024, pp. 487–520. [Online]. Available: https://aclanthology.org/2024.eacl-long.30

[6] D. Yu, L. Deng, and G. Dahl, "Roles of pre-training and fine-tuning in context-dependent dbn-hmms for real-world speech recognition," in *Proc. NIPS Workshop on Deep Learning and Unsupervised Feature Learning*. sn, 2010.

[7] K. Versteegh, *Arabic language*. Edinburgh University Press, 2014.

[8] K. Ibn and F. Rosenthal, *The Muqaddimah : an introduction to history*, 2nd ed., ser. Bollingen series; 43. Princeton, N.J.: Princeton University Press, 1967.

[9] E. Alsharhan and A. Ramsay, "Investigating the effects of gender, dialect, and training size on the performance of Arabic speech recognition," *Language Resources and Evaluation*, vol. 54, no. 4, pp. 975–998, Dec. 2020, company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 4 Publisher: Springer Netherlands. [Online]. Available: https://link.springer.com/article/10.1007/s10579-020-09505-5

[10] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, Eds. Marseille, France: European Language Resources Association, May 2020, pp. 4218–4222. [Online]. Available: https://aclanthology.org/2020.lrec-1.520

[11] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-demos.6

[12] M. Al-Fetyani, M. Al-Barham, G. Abandah, A. Alsharkawi, and M. Dawas, "Masc: Massive arabic speech corpus," 2021. [Online]. Available: https://dx.doi.org/10.21227/e1qb-jv46

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[14] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *arXiv preprint arXiv:1606.08415*, 2016.

[15] H. Hassan, "Open ai introducing whisper," *Medium*, 2022. [Online]. Available: https://medium.com/@TheHaseebHassan/open-ai-introducing-whisper-ab7517a91108

[16] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[17] S. Gandhi, "Fine-tune whisper for multilingual asr with transformers," 2022. [Online]. Available: https://huggingface.co/blog/fine-tune-whisper

[18] V. Srivastav, S. Majumdar, N. Koluguri, A. Moumen, S. Gandhi *et al.*, "Open automatic speech recognition leaderboard," https://huggingface.co/spaces/hf-audio/open_asr_leaderboard, 2023.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[20] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

Appendices

A Additional results experiment 2

Table 1: Model performance on each test set with and without pre-training on MSA. Best results per test set (columns) and metric are indicated in **bold**.

Pre-Training	Metric	Train set	Egyptian	Gulf	Iraqi	Levantine	Maghrebi	MSA
With	**WER (↓)**	**All**	73.89	83.04	84.91	79.06	90.22	**55.19**
		Egyptian	73.71	93.08	97.52	92.43	103.13	59.77
		Gulf	83.37	**79.12**	93.33	83.07	95.19	56.80
		Iraqi	97.70	103.21	83.78	108.72	122.69	65.26
		Levantine	84.97	87.99	92.53	**73.87**	94.56	56.69
		Maghrebi	85.31	91.46	97.37	84.11	**78.56**	58.27
		Whisper-Small	101.79	127.14	147.23	113.13	152.69	61.83
	CER (↓)	**All**	47.21	59.03	55.24	54.54	63.48	**22.04**
		Egyptian	44.34	64.73	62.36	62.13	73.43	24.80
		Gulf	55.37	**57.77**	61.85	54.28	68.08	23.14
		Iraqi	61.01	67.53	53.66	74.76	84.90	26.48
		Levantine	52.66	59.45	58.08	**47.87**	65.61	23.76
		Maghrebi	51.78	60.13	59.93	52.48	**51.21**	23.52
		Whisper-Small	64.12	91.11	97.91	75.47	106.29	26.63
Without	**WER (↓)**	**All**	72.73	83.43	85.31	83.77	89.50	61.60
		Egyptian	**69.11**	91.89	98.48	86.60	95.40	69.16
		Gulf	84.89	83.18	94.12	81.26	96.67	64.05
		Iraqi	89.20	95.19	**78.33**	93.65	97.01	76.32
		Levantine	88.64	95.35	101.41	88.37	116.39	65.78
		Maghrebi	83.93	91.87	96.64	84.13	79.56	65.00
		Whisper-Small	101.79	127.14	147.23	113.13	152.69	61.83
	CER (↓)	**All**	45.48	60.32	54.89	58.83	63.01	25.07
		Egyptian	**42.22**	63.89	62.64	55.45	64.83	29.78
		Gulf	53.67	61.65	60.81	54.16	67.74	26.86
		Iraqi	58.42	67.37	**52.93**	70.12	69.35	36.31
		Levantine	57.27	66.94	65.50	60.55	88.48	28.58
		Maghrebi	52.13	62.47	60.65	53.08	53.69	27.29
		Whisper-Small	64.12	91.11	97.91	75.47	106.29	26.63

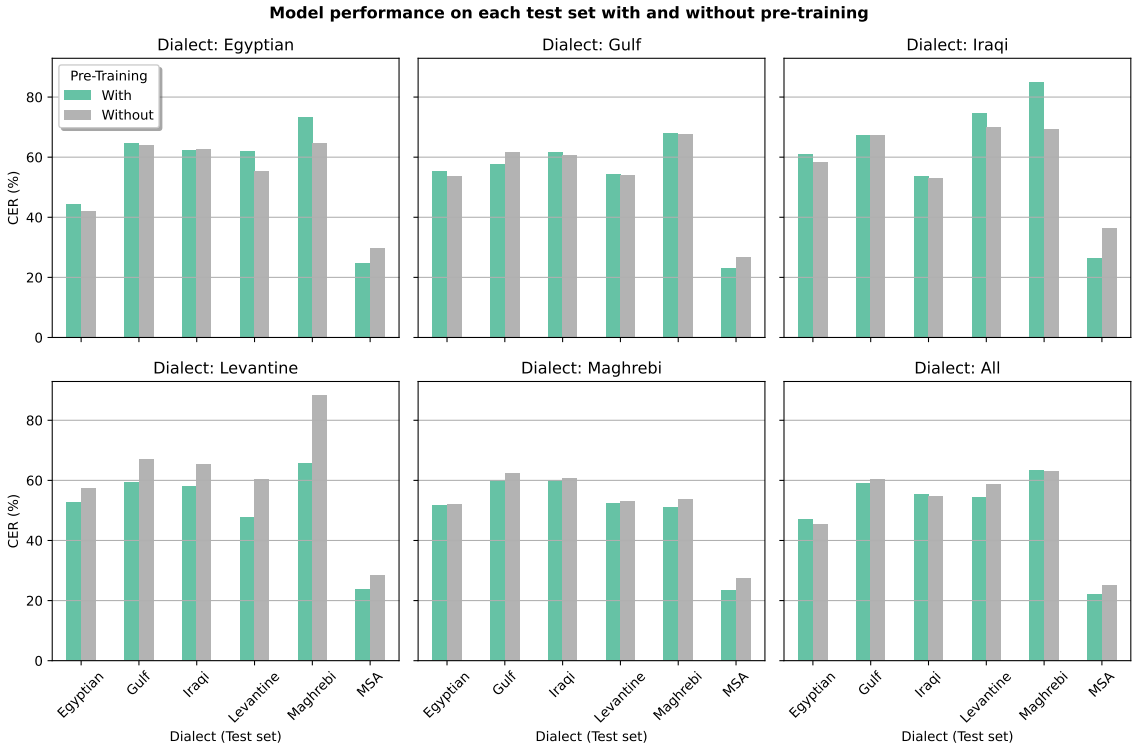

Figure 1: Pre-trained versus non pre-trained CER performance per trained model (plot title) and per test set (x-axis).

B Additional results experiment 3

Table 2: Model performance comparison dialect-pooled vs dialect-specific. The dialect-specific model for MSA is the pre-trained model from Experiment 1.

Metric		Dialect-pooled	Dialect-specific
WER (↓)	**Egyptian**	73.89	73.71
	Gulf	83.04	79.12
	Iraqi	84.91	83.78
	Levantine	79.06	73.87
	Maghrebi	90.22	78.56
	MSA	55.19	45.84
CER (↓)	**Egyptian**	47.21	44.34
	Gulf	59.03	57.77
	Iraqi	55.24	53.66
	Levantine	54.54	47.87
	Maghrebi	63.48	51.21
	MSA	22.04	19.78

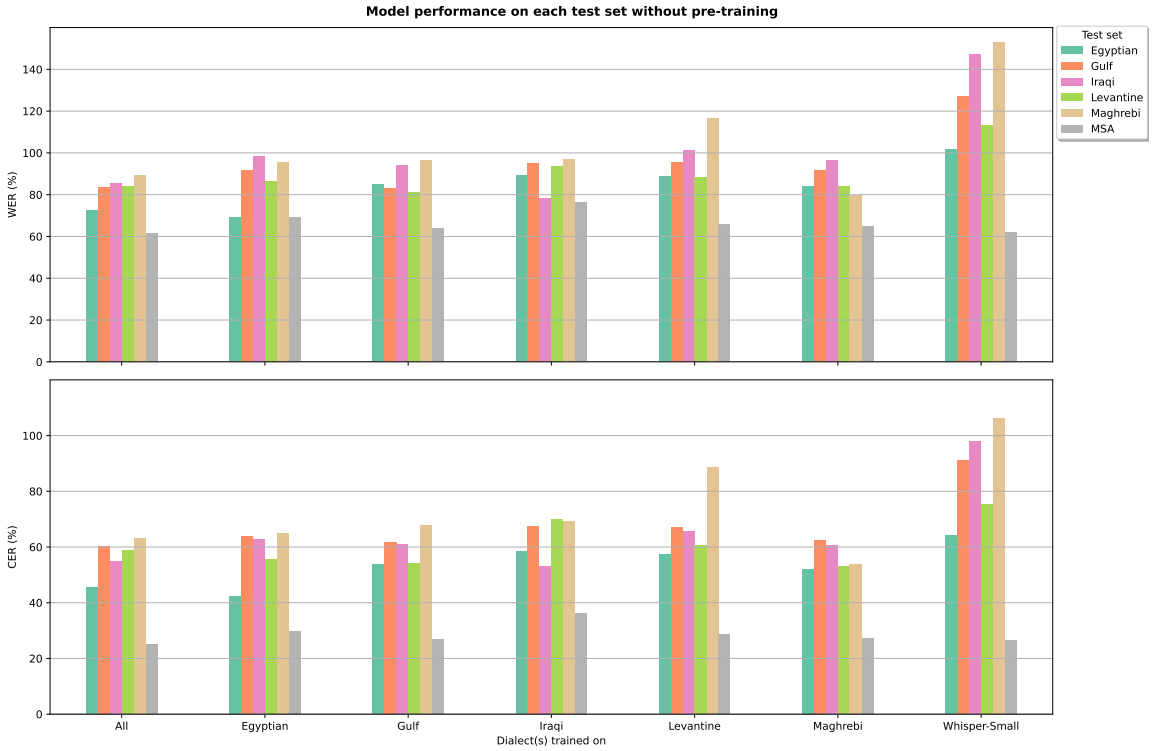

Figure 2: Model performance in WER and CER of the non pre-trained configurations on each test set per model. Hatched bars denote the test set of the same dialect as the train set.

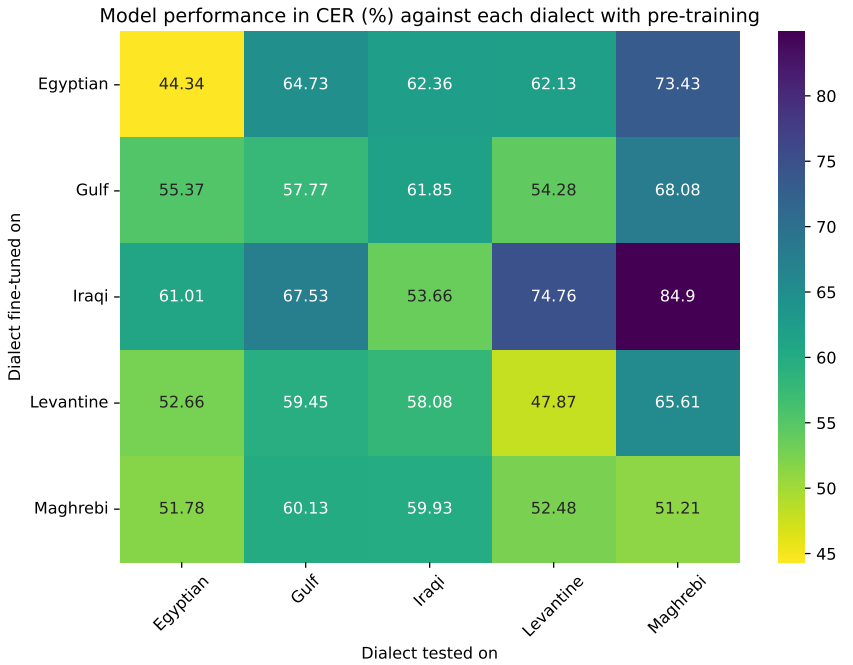

Figure 3: Confusion matrix of CER performance between the dialectal train and test sets. X-axis denotes the train dataset, while the test set is shown on the y-axis. Models are pre-trained on MSA.

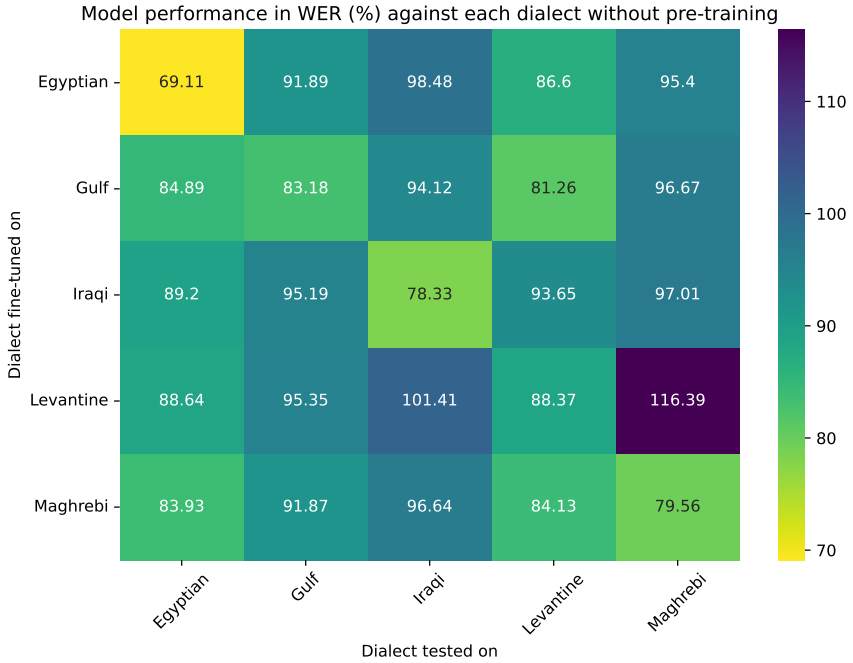

Figure 4: Confusion matrix of WER performance between the dialectal train and test sets. X-axis denotes the train dataset, while the test set is shown on the y-axis. Models are not pre-trained on MSA.

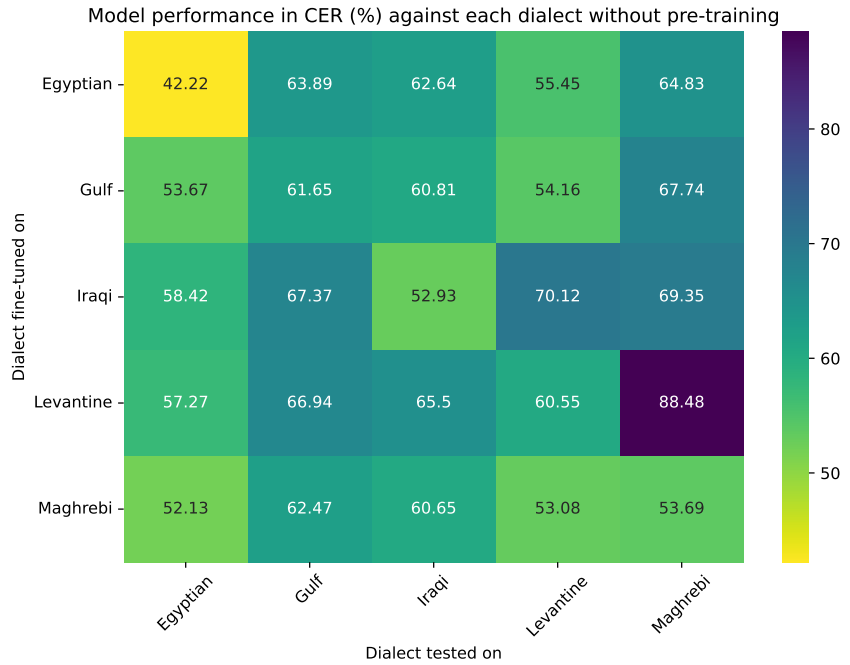

Figure 5: Confusion matrix of CER performance between the dialectal train and test sets. X-axis denotes the train dataset, while the test set is shown on the y-axis. Models are not pre-trained on MSA.