Learning Cross-Dialectal Morphophonology with Syllable Structure Constraints

Salam Khalifa, Abdelrahim Qaddoumi, Jordan Kodner, and Owen Rambow
Department of Linguistics, and
Institute for Advanced Computational Science (IACS)
Stony Brook University
{first.last}@stonybrook.edu

Abstract

We investigate learning surface forms from underlying morphological forms for low-resource language varieties. We concentrate on learning explicit rules with the aid of learned syllable structure constraints, which outperforms neural methods on this small data task and provides interpretable output. Evaluating across one relatively high-resource and two related low-resource Arabic dialects, we find that a model trained only on the high-resource dialect achieves decent performance on the low-resource dialects, useful when no low-resource training data is available. The best results are obtained when our system is trained only on the low-resource dialect data without augmentation from the related higher-resource dialect. We discuss the impact of syllable structure constraints and the strengths and weaknesses of data augmentation and transfer learning from a related dialect.

1 Introduction

Many of the world's under-resourced language varieties are closely related to higher-resourced varieties. This suggests two possibilities for progress on the under-resourced varieties: the development of systems that perform better with smaller training data, and the development of systems that leverage information from the higher-resource variety to augment learning for the lower-resource one. In this paper, we combine these two approaches: we employ a learning technique that works well with small amounts of data (namely, rule learning) and we evaluate the impact of providing the the model training data combined from both a low-resourced variety and a similar but higher-resourced variety.

Arabic is particularly well-suited for studying such techniques because the Arabic dialects represent a continuum of related but distinct and thriving spoken varieties, yet most have limited computational resources available for them. On the other hand, the dialects maintain varying degrees of mutual intelligibility. A system developed for one dialect will may not capture everything in another dialect, but they are generally close enough that some transfer learning should be feasible. Furthermore, Arabic is morphologically rich. Even affixation in Arabic triggers a range of morphophonological processes which may yield surface forms that are noticeably different from their underlying morphological analyses. Uncovering these processes is crucial for understanding Arabic morphology. Moreover, difference in these processes account for much of the difference between the spoken forms across dialects. Underlyingly identical forms across dialects may surface very differently, as examples show in Table 1. As a consequence, a morphological analyzer or generator developed specifically for one dialect will not work reliably on other dialects.

	kitaab+hum	**kaatib+iin+ha**
Egyptian	kit**a**bhum	k**a**tbinha
Sudanese	kitaabum	kaatbinna
Jordanian	*kitaabhum*	kaatbiinha
Hijazi	kitaabahum	kaatbiinaha
	their book	*they/we are writing it*
	كتابهم	كاتبينها

Table 1: Realizations of two words across four dialects. The dialects share the same underlying representations. Changes in the realized forms are highlighted as follows: shortened vowels are **bolded**, epenthetic phones are underlined, deleted phones are not shown, and finally, realizations faithful to the underlying representations (i.e., no change) are *italicized*.

In this paper, we study how we can use resources for this relatively resource-rich dialect and apply them to resource-poor dialects. We take on the task of matching annotated underlying forms to attested surface forms (Khalifa et al., 2022, 2023). Khalifa et al. (2023) study this task for Cairene Egyptian

January 19, 2025. ©2025 Association for Computational Linguistics

Arabic (EGY), which, while not high-resourced in absolute terms, has quite a few scholarly and corpus resources available, much more than other dialects. They show that when only small amounts of data are available, rule-learning approaches outperform neural sequence-to-sequence models. They also perform a somewhat perfunctory study on Sudanese Arabic (SUD). We adopt their general problem formulation, but we specifically investigate how we can apply EGY resources to other dialects, choosing for our study SUD and Jordanian (JOR), two low-resource dialects. We investigate a variety of techniques on these two dialects which are relatively close to Cairene, but differ in their details.

We investigate four training conditions, which combine transcribed spoken Arabic dialect data in different ways. In our experience, while not naturally occurring, some transcribed speech is available for many Arabic dialects, and it is more easily obtained than the underlying representations. The conditions are a follows. For clarification, "full data" refers to a corpus of pairs of underlying morphological representations and surface forms, while "surface forms" only refers to a corpus which contains attested forms (transcribed spoken language), but no linguistic analysis has been performed to create the underlying representations:

1. Only EGY Full data, with target dialectal data only used for testing. This is the only option Khalifa et al. (2023) explore.
2. Only EGY Full data, and in addition we have surface forms for the target dialects.
3. Full data for EGY and the target dialects.
4. Full data for the target dialects.

Our paper makes two primary contributions:

- We present a novel approach that uses syllable structure constraints in words to derive surface forms from underlying representations. We compare two ways of deriving such constraints. We show that using such constraints nearly always helps over not using them.
- We compare and contrast the above four ways of using combinations of higher- and lower-resource dialectal data. For SUD and JOR, just training on even a very small amount of dialectal data only outperforms including EGY data. When we only have surface forms for the lower-resource dialects, then using syllable constraints in conjunction with EGY outperforms using EGY alone.

The structure of this paper is as follows. We discuss related work in Section 2. We present the linguistic background in Section 3 and the data in Section 4. We present our method with details on all steps in Section 5, and report on experimental results in Section 6. We conclude with an analysis and a report on ongoing and future work.

2 Related Work

2.1 Arabic Cross-Dialectal Learning

Cross-dialectal learning is a popular area of study in Arabic NLP due to the nature of the language as a dialect continuum (Zalmout, 2020; Khalifa et al., 2020; Inoue et al., 2022; Micallef et al., 2024). However, most efforts explore the task of knowledge transfer through different neural network architectures. These approaches suffer from a lack of linguistic interpretability, which often hampers their applicability in a scientific setting. One exception is Salloum and Habash (2014), who presented their morphological analyzer, ADAM, for multiple dialects in Arabic. ADAM extends an existing Modern Standard Arabic (MSA) morphological analyzer to three dialects through the mapping of MSA affixes and clitics while assuming similar stems: Levantine, Egyptian, and Iraqi. These mappings were explicit and interpretable, however, they relied on hand-crafted rules and only addressed morphotactics (distributions of morphemes) and orthotactics, not morphophonology.

2.2 Learning Morphophonological Mappings

We take an explicit rule-based approach to Arabic dialectal morphophonology, the interaction between morphology and phonological processes. Rule-based learning provides interpretable outputs, unlike off-the-shelf neural approaches, and this facilitates comparison across dialects, is valuable for text-to-speech tasks, and supports the linguistic analysis of less-studied language varieties. Morphophonological rule learning in particular has usually been studied within computational phonology (Antworth, 1991; Albright and Hayes, 2002; Ellis et al., 2022). However, there has been recent work dedicated to morphophonology learning (Khalifa et al., 2023; Wang, 2024). Both works study different morphophonological phenomena through learning constraints in different representations. In this work, we base our core learning algorithm on Khalifa et al. (2023), which was primarily tested on Arabic and focused on learning morphophonological mappings between underlying morphological representations such as those generable by a generic Arabic morphological analyzer (URS) and surface forms which actually appear in dialect corpora (SFS). We make novel contributions in incorporating models of syllable structure constraint learning from the grammatical inference literature as well as the evaluation of transfer learning strategies between multiple dialects.

3 Linguistic Background

3.1 Morphophonology

Morphophonology is the interaction between phonology and morphology, where certain phonological processes are triggered when the word structure is modified. Studying morphophonology across different dialects of Arabic allows understanding different phonological processes through morphologically related words. Such morphophonological processes are governed by phonological constraints on syllable structure which interact with the morphology, especially concatenative morphology. These constraints can differ drastically between different dialects resulting in noticeably different surface form realizations for the same underlying morphological representation as shown in Table 1.

3.2 Syllable Structure

Most phonological processes in dialectal Arabic are triggered by strict dialect-specific requirements on how segments are organized into syllables. Affixation triggers resyllabification, which in turn forces morphophonological repairs which maintain these restrictions.

We lay out some examples of dialectal morphophonological patterns here. One requirement shared across Arabic is that each syllable must begin with exactly one consonant. When an underlying representation begins with a vowel, that onset consonant is supplied by insertion of a glottal stop (hamza) when the word is in isolation. Some dialects, such as Jordanian JOR, additionally permits word initial syllables starting with a complex consonant cluster of two consonants. A second requirement is that syllables may end in no more than one consonant (one so-called coda consonant). The dialects differ in the strictness of this constraint: while SUD bans them across the board, EGY and JOR only ban them word-internally. They permit multiple coda consonants in word-final positions. Furthermore, dialects repair clusters of coda consonants differently. As such, when concatenation of morphemes creates a sequence of three consonants, such as in the underlying representation of the word 'you wrote us' `/katab-t-na/`, the three dialects yield different surface forms. EGY and SUD both insert a vowel after the second consonant (which happens to differ between them) yielding `[katabtina]` and `[katabtana]` respectively, while JOR inserts it after the first consonant as in `[katabitna]`.

The phonological form of the affix can trigger different repairs as well. For example, if a suffix starts with `/h/`, then SUD deletes the `/h/` rather than inserting a vowel to break up the sequence of three consonants. EGY, unlike SUD or JOR, only permits high and low vowels to be long and only permits them in stressed syllables. Similarly, long vowels are restricted to open syllables except in word-final position. Thus, underlyingly long vowels are shortened when unstressed or in word-medial closed syllables, and they are raised if underlyingly mid. There is a myriad number of literature discussing more requirements and in-depth analysis cross-dialectally (Hamid, 1984; Broselow, 1976, 1992; Broselow et al., 1995, 1997; Broselow, 2017; Farwaneh, 1995).

4 Data

In order to learn morphophonology mappings, the data is represented in pairs of underlying representations (UR) which is a sequence of morphs in a hypothetical but consistent form that could be motivated theoretically or derived from the output of a morphological analyzer, and a surface (spoken) form (SF), which is phonically transcribed. The LDC transcription scheme was used for both UR and SF. A mapping between LDC, IPA, and Arabic script can be found in Appendix Table 7.

We augmented the character set with a symbol for word boundaries #, a symbol for prefix boundaries -, and a symbol for suffix boundaries =. We opted for only open class words, i.e., nouns, adjectives, and verbs, as other categories such as proper nouns are more likely to manifest exceptional processes which violate the otherwise norms in their respective dialects. In addition, we restrict learning to concatenative morphology. We leave templatic morphology for future studies. The major consequence of this design decision is that different templatic realizations within a given morphological

EGY		JOR		SUD	
UR	SF	UR	SF	UR	SF
ti-kallif	tikallif	ti-kallif	'itkallif	ta-kallif	takallif
#CV CVC CVC#		#CVC CVC CVC#		#CV CVC CVC#	
bi-ti-kallif	bitkallif	bi-ti-kallif	bitkallif	bi-ta-kallif	bitkallif
#CVC CVC CVC#		#CVC CVC CVC#		#CVC CVC CVC#	
samaH=t	samaHt	samaH=t	samaHit	samaH=t	samaHta
#CV CVCC#		#CV CV CVC#		#CV CVC CV#	
$Af=U=kI	$afUki	$Af=U=kI	$AfUki	$Af=U=kI	$AfOki
#CV CVV CV#		#CVV CVV CV#		#CVV CVV CV#	

Table 2: Examples showcasing the pairs of UR-SF of the same words in the three different dialects along with the syllabification of each SF. In some cases dialect share the same UR but have different realizations as can be seen in the last two rows. In other cases they can shared the same SF but with different UR as seen in the second row. The '-' represent prefixes boundary and '=' represent suffixes boundary. Underlining across rows indicate identical URs and SFs across the dialects. The words are تكلف 'it [f.sg] costs', بتكلف 'it [f.sg] is costing', سمحت 'you [m.sg] permitted', شافوكي 'they saw you [f.sg]', respectively.

paradigm are treated as distinct unrelated stems.

4.1 EGY

We treat EGY as our high-resource dialect in our cross-dialectal learning setup. Following (Khalifa et al., 2023) for purposes of comparison, we use the same dataset that was built on (**ECAL**; Kilany et al., 2002), a pronouncing dictionary based on CALLHOME Egypt (Gadalla et al., 1997). This provided surface forms (SF). To match these SFs with appropriate (UR)s, we used CALIMA$_{EGY}$ (Habash et al., 2012), a morphological analyzer for Egyptian Arabic, to generate (UR)s through the morphological tokenization produced by CALIMA$_{EGY}$. See (Khalifa et al., 2024) for details about (UR) generation. We use the same data splits as ECAL provides a split into TRAIN (12,658 types), DEV (5,181 types), and EVAL (6,976 types) sets, which we adopted. However, since these splits were based on running text, individual words overlap between the sets. To account for this, we create two additional sets, OOV-DEV (2,190 types), and OOV-EVAL, based on DEV and OOV-EVAL (2,271 types) based on EVAL, but without their intersections with TRAIN.

4.2 Annotation for SUD and JOR

We chose to study SUD and JOR due to their status as under-resourced dialects compared to EGY. EGY lies between the two both geographically and in the dialect continuum and so shares some properties with both. For both low-resource dialects, the datasets were created by picking the 700 most frequent open class words from the Multi-Arabic Dialect Applications and Resources dataset (**MADAR**; Bouamor et al., 2018), which is a 25-way parallel corpus representing the dialects of 25 cities. SUD and JOR were taken from portions of the Khartoum and Amman city dialects, respectively. MADAR was created by translating sentences from English and French from the Basic Traveling Expression Corpus (**BTEC**; Takezawa et al., 2007). The corpus is orthographic, so we created both the underlying representation (UR) and the dialect-specific surface forms (SF) during our annotation.

Unlike EGY, there are no available morphological analyzers that would have otherwise expedited the annotation by generating potential URs. While other phonemically transcribed corpora of Arabic exist (Appen, 2006a,b, 2007; Maamouri et al., 2007), we opted for MADAR because it is open source and will allow us to publish the data publicly. However, one caveat with using MADAR is the potential limited diversity of the data due to the specific domain of MADAR, which is the travel domain. This is unlike EGY, since ECAL was compiled from more diverse and naturalistic spoken conversations.

For both dialects, native speakers with adequate training in linguistics were asked to transcribe the spoken form for each word to the best of their ability. The speakers were then asked to provide URs. When there were multiple plausible URs for a given SF, we limited the analysis to one UR chosen to be consistent with the rest of the annotation. This is followed by a series of revisions and well-formedness checks to insure consistency between the URs within each dialects as much as possible.

Each dialect was annotated by a single native speaker due to logistical constraints that limited access to additional annotators. Consequently, it was not possible to measure inter-annotator agreement. This effort resulted in a total of 710 and 771 pairs for SUD and JOR, respectively. We used 300 for TRAIN, 200 for DEV, and the rest for EVAL for each dialect. Since these dataset were annotated based on a frequency list, there are no overlap between them.

We show examples of pairs of UR-SF of common shared words and contrast the difference between the three dialects in Table 2.

5 Methodology and Experimental Setup

Our approach extends the Pruned Abundance Rule Learning Algorithm (PARLA; Khalifa et al., 2023) as the primary rule learning technique; our contributions lie mainly in exploring several aspects of cross-dialectal learning. Our research focus is on new data augmentation techniques for dialect transfer, improved rule learning scope, and the inclusion of syllables structure as a linguistically motivated signal for rule learning.

System	R	R%	TRAIN	DEV	OOV-DEV
Kh '23	2,922	23.1	97.2	89.4	80.4
Ours	1,721	13.6	95.9	88.9	81.6

Table 3: A comparison between our implementation and prior work (Kh'23 Khalifa et al., 2023) in terms of the number of rules (R) and their ratio with respect to the size of the TRAIN (R%), and accuracy on each split of the data.

5.1 Rule Learning Algorithm

We reimplemented PARLA as a base and made several additions. Our implementation outperforms the system of Khalifa et al. (2023) on EGY, as presented in Table 3.

First, we enhanced the rule extraction step by enforcing morpheme boundaries on the SF before rule extraction, this was inspired by a similar technique in (Antworth, 1991). This was implemented through character alignment between the UR and SF to approximate morpheme boundaries using (Khalifa et al., 2021). It greatly reduced the number of rules by eliminating any superficial rules that resulted from encoding morpheme deletion as an actual change. We increased the left and right context windows from PARLA's 1 to 2 in order to accommodate the extra boundary characters that are retained in the SF at this step.

Second, we include syllable structure information to assess the well-formedness of prediction SFs when selecting rules at inference time. Dialect-specific syllable structure constraints are learned from the set of SFs in the training data. We evaluate two different approaches based on learning positive or negative constraints as expounded in Section 5.5.

5.2 Data Utilization

We explore three methods of training augmentation with data from the (relatively) high-resource dialect TRAIN$_{EGY}$ using three methods.

High Resource + Surface-Only Low Resource In this transfer learning setup, we simulate a situation where no training data exist for the target dialect, but only surface form. Therefore, PARLA is trained using only TRAIN$_{EGY}$, and the surface-only low-resource is used for syllable structure constraint as we will explain shortly.

Low Resource Only Here, we assume we only have a small training dataset for each of the target dialect, i.e., TRAIN$_{SUD}$ and TRAIN$_{JOR}$. These datasets will be used to train PARLA and to extract syllable structure constraints.

High and Low Resource We look at two methods for combining training data from a high- and low-resource dialects: **a)** naive concatenation of the datasets, i.e., TRAIN$_{EGY+SUD}$ and TRAIN$_{EGY+JOR}$, **b)** concatenation of only the compatible entries of TRAIN$_{EGY}$ with respect to the target dialect. Compatibility is based on both UR and SF: Entries from TRAIN$_{EGY}$ are removed if they share a UR with an entry in the target dialect training set or if their SF has a syllable structure that is invalid in the target dialect. We call these training sets TRAIN$_{EGY'+SUD}$ and TRAIN$_{EGY'+JOR}$.

5.3 Training Scope

The core mechanism of our rule-learning approach is the *rule evaluation* step, where each extracted rules is evaluated against the the entire training set. However, it is not immediately obvious how this should be performed when the training set mixes dialects, since evaluating a rule for one dialect against data from another could mislead the system. We consider three alternative approaches:

Default Every extracted rule is evaluated against the entire training set regardless of the source dialect of the data point corresponding to the rule.

Partitioned Each rule is evaluated only against the portion of the training data that matches the dialect of the data point that it was extracted from. This is practically equivalent to training on each dialect separately and combine the resulting rules.

Target Only Each rule is evaluated only against the portion of the training set matching the target dialect regardless of which dialect original data point is from.

5.4 Rule selection

At inference time, the rules learned during training are sorted by their specificity and are traversed sequentially until some rule's left-hand side matches the context of the input UR (Khalifa et al., 2023). Given the mixed training, we experiment with the additional sorting criterion in which rules that have been extracted from the target dialect's training data are ranked ahead of those extracted from EGY.

5.5 Syllable Structure

Most phonological changes associated with morphological processes in Arabic are in fact resyllabification as discussed in Section 5.5, therefore, we posit that leveraging syllables structure should boost performance. We use learned syllable structure constraints at inference time to probe the well-formedness of generated SFs in or to filter out invalid predictions. When an invalid SF is produced, the system moves onto the next applicable rule. If all applicable rules yield ill-formed structures, then it is assumed no change happens.

We evaluate two types of syllable structure constraints, positive constraints that license structures that are attested among the surface forms of a dialect's training data, and negative constraints which ban structures absent in the training data. Low-resource languages provide a particular challenge here, since learned constraints are highly sensitive to the size and syllabic diversity of the training data. A small training set may result in an excessively restrictive grammar due to accidental gaps. Nevertheless, we find syllable structure constraints to be helpful in practice. For both approaches, we first automatically syllabify a dialect's surface forms using (Kodner, 2016). Syllabification itself is fairly trivial, especially for Arabic since syllables without onsets are prohibited. Example 1 shows surface forms along with their syllabification, and the abstracted syllabic structure. Consonants and vowels are abstracted to C and V, and a long vowel is represented with VV. Word boundaries are represented with a '#'.

(1)	`kitaab`	`ki.taab`	`#CV CVVC#`
	`qalam`	`qa.lam`	`#CV CVC#`
	`kutub`	`ku.tub`	`#CV CVC#`
	`kibiir`	`ki.biir`	`#CV CVVC#`

Positive Syllable Grammar ($\mathbf{G}_{+}$) We extract a positive grammar by syllabifying the SF from the training data and then extracting all attested syllable structures. For example, the surface forms in Example 1 will generate the following grammar of two permissible syllable sequences:

```
(2)  {[#CV CVVC#],[#CV CVC#]}
```

The $\mathbf{G}_{+}$ in (2) is used as follows: if the syllable structure of a predicted SF at inference time does not match in any instance in the set, it is rejected as invalid.

Negative Syllable Grammar ($\mathbf{G}_{-}$) We apply the Bottom-Up Factor Inference Algorithm (BUFIA; Chandlee et al., 2019)[1] to extract negative constraints in the form of *forbidden factors*. We present BUFIA with the same syllabified representation for SFs as above. Using the same example, BUFIA generates the following negative grammar:

```
(3)  {[CV CV], [CV #], [CVC CV],
     [CVC CVC], [CVC CVVC],
     [CVVC CV], [CVVC CVC],
     [CVVC CVVC], [# CVC],
     [# CVVC], [# #]}
```

The $\mathbf{G}_{-}$ in (3) is used as follows: if the syllable structure of the predicted SF includes any sequence in the is rejected as invalid. Using very little data, such as in the toy example above, we generate extremely conservative grammars and are likely accept similar output. However, as the data increases, we expect $\mathbf{G}_{-}$ to be more general.

6 Evaluation

We organize the evaluation discussion according to our data setups introduced in Section 5.

[1] https://github.com/heinz-jeffrey/bufia

6.1 Baselines

We consider two baselines. These have different goals and elucidate different aspects of the task.

DoNothing: Not all underlying forms undergo morphophonological alternations, since not all affixation requires repair. This baseline is the proportion of test SFs which undergo no change beyond removing morpheme boundaries from their corresponding URs, or in other words, the performance achieved when nothing is done. Thus, **DoNothing** establishes a hard lower bound on performance. A model should not perform worse than doing nothing.

Neural: The task of mapping URs to SFs is conceptually similar to a grapheme-to-phoneme task in how it maps one similar string to another. We train and evaluate a state-of-the-art a neural character-based transformer for this task (Wu et al., 2021). Ideally, a rule based model should perform competitively with the neural model, especially in low-resource data settings.

6.2 High Resource + Surface-Only Low Resource

The first set of experiments rely on Train$_{EGY}$ alone for annotated data, while syllable structure constraints are learned from unannotated dialect SFs. The Egy columns in Table 4 showcases the results for this scenario. Using any syllable information helps and improves upon base Parla trained only on Egy with no syllable structure information. As expected, the improvements are greater when the constraints are learned from the target dialect's SFs than from Egy. Both **G**$_+$ and **G**$_-$ yield improvements over basic Parla, though **G**$_-$ underperforms **G**$_+$. The weak performance of **G**$_-$ constraints for Jor is likely due to the sparsity in the syllable structures in its training. While the training data shows that Sud has 74 unique syllable structures and Jor has 61, Jor has 11 syllable shapes while Sud has only 8. This affects the restrictive behavior or **G**$_-$, BUFIA extracted 80 negative factors for Sud and a 100 for Jor.

From this experiment, we can conclude that transfer from the high-resource dialect to the low-resource target dialect is effective. It sometimes even surpasses the **Neural** baseline, even with no additional information. Adding syllable structure information from even a small amount of data in the target dialect further improves performance. Such data is available for many Arabic dialects (Appen, 2006a,b, 2007; Maamouri et al., 2007).

6.3 Low Resource Only

In this scenario, we train Parla only on the limited annotated training data available for the target dialect. The second column in Table 4 showcases the results. Training on limited target data directly greatly outperforms all settings including Egy as well as the **Neural** and **DoNothing** baselines. Using **G**$_-$ yields a further small improvement for both dialects, while **G**$_+$ does not.

6.4 High and Low Resource

In this setup, we leverage all available training data by concatenating Train$_{EGY}$ with each dialect using two settings. The first, is naive concatenation while the second is concatenating a filtered Train$_{EGY}$ as described in Section 5.2. The last column for each dialect in Table 4 showcases the results for the the naive concatenation setup. The general trend seems to be that concatenating the data does not help when compared with training using the dialect alone. Results with syllable structure information follow similar trends as the previous experiment. We trained **Neural** on the naive concatenated set and it outperformed Parla+**G**$_-$ for both dialects in the same setup, however, it still lags behind the best performing setup for both dialects which is inline with previous findings on the value of rule learning approaches for extremely low-resource setups. Additionally, the performance of **Neural** appears correlated with that of Parla on a by-dialect basis.

Following the discussion in Section 5.3, we perform additional training experiments using Train$_{EGY'+DIA}$ for each dialect. Even though the performance using the concatenation techniques was similar, we opted for Train$_{EGY'+DIA}$ since it learns fewer rules from a smaller set of data as we will show in the discussion section. Table 5 shows the results for all three setups in Section 5.3. In all setups except for Train$_{EGY'+DIA}$, we ordered the rules at inference time as described in Section 5.4. The effect of the sorting alone is indicated in the difference in performance between the first two columns of each dialect in Table 5. Partitioned training, as shown in columns 'PART' in Table 5, boosts the performance for Sud but not as high as training on Train$_{SUD}$ alone, unlike the case with Jor where in fact it hurts the performance quite noticeably. For both dialect, using

	SUDanese			JORdanian		
Sys/Train	**EGY**	**SUD**	**EGY+SUD**	**EGY**	**JOR**	**EGY+JOR**
PARLA	67.5	85.0	73.0	68.0	76.0	70.0
+EGY_G+	69.5	-	-	69.5	-	-
+EGY_G-	68.5	-	-	68.0	-	-
+DIA_G+	71.5	79.0	69.5	70.0	75.5	71.0
+DIA_G-	72.0	85.5	73.0	68.5	77.0	71.5
NEURAL	73.5	50.5	79.0	65.0	37.5	74.5
DoNoth	60.0			58.5		

Table 4: Accuracy (%) results when training Parla using different training sets in addition to using positive and negative syllable structure grammars at inference time and testing on the Dev of the respective target dialects Sud and Jor. +Egy indicates syllable structure constraints trained on Egyptian, +Dia indicates syllable structure constraints trained on the target dialect. $\mathbf{G}_+$ indicates positive constrains and $\mathbf{G}_-$ indicates negative constraints. Our baselines are reported as **DoNothing** and **Neural**. Note that **DoNothing** is independent of any training data.

$\mathbf{G}_-$ boosts the performance. In the last setup, rules are extracted from both datasets but only evaluated against the target dialect. In this setup, as shown in last columns for each dialect, Sud reaches peak performance with the boost from $\mathbf{G}_-$. The performance of Jor while relatively high, it is still a tad behind Train$_{JOR}$+$\mathbf{G}_-$ on its own.

7 Analysis and Discussion

7.1 Acquired Knowledge

In this section we take a closer look into the system's "knowledge" in terms of *rules* that are learned and their relationship with the training data. This is summarized in Table 6. For both Train$_{SUD}$ and Train$_{JOR}$ the trend in the number of rules is clearly related to data paucity. This also manifests in the poor **DoNothing** baselines and the size of $\mathbf{G}_-$ as discussed in Section 6.2. Additionally, it seems that Train$_{EGY'+SUD}$ with the partition (+PART) configuration acquires the same set of rule as Train$_{SUD}$ with evidence in the similar performance. However, Train$_{EGY'+JOR}$ with the same configuration learns more rules and the performance stays relatively the same.

Additionally, training using both Train$_{EGY+SUD}$ and Train$_{EGY+JOR}$ yielded more rules than Train$_{EGY}$ alone, suggesting that the system learned rules from the target dialect as well as Egy. While it improved the performance over Train$_{EGY}$, it was still substantially lower than training on the dialect alone. This could be due to the relative attestation of each dialect in the combined training set. With Egy being much larger, its contribution to the rule set "washed out" the contribution of the target dialect.

7.2 Effect of Augmenting with Egy

Syllable structure proved beneficial for cross-dialectal learning, on the other hand, data augmentation did not meet our expectations. We analyzed the errors that differentiated training on Train$_{DIA}$ and Train$_{EGY'+DIA}$ for both dialects. For Jor, most of the errors that were unique to Train$_{EGY'+JOR}$ were on entries that should have been copied from from Ur (**DoNothing** predicts the correct Sf), because rules extracted from Egy applied unnecessarily. Most of these rules were long vowel shortening and high vowel deletion rules which are prevalent in Egy phonology but not Jor. On the other hand, Train$_{EGY'+JOR}$ did pick up a few cases with the help of rules from Egy that were not recovered on Train$_{JOR}$. While these rules covers similar linguistic phenomena, the Jor rules had more specific context compared to those from Egy, which could lead to overapplication. The difference between Train$_{SUD}$ and Train$_{EGY'+SUD}$ is more substantial. In addition to types of errors similar to those found in Jor, rules enforcing resyllabification of final complex codas were not extracted because the evidence from the Sud component of the combined training set was insufficient in the face of counterexamples in the Egy component.

	SUDanese				**JORdanian**			
Sys/Train	**EGY'+SUD**	**DEF**	**PART**	**SUD-only**	**EGY'+JOR**	**DEF**	**PART**	**JOR-only**
PARLA	73.0	76.0	80.0	85.0	69.5	69.5	67.0	75.0
+DIA_G+	69.0	70.0	75.5	79.0	70.5	71.0	71.0	75.5
+DIA_G-	73.0	76.0	81.5	85.5	71.0	72.5	71.5	76.5
DoNoth	60.0				58.5			

Table 5: Accuracy (%) results when training Parla using Train$_{EGY'+DIA}$ with different training methodologies. Evaluation is on the Dev of the target dialects Sud and Jor. We also report accuracies when using both positive and negative grammars for each setup. We also report **DoNothing** which is independent of any training data.

Train	**ACC@**	**R**	**R%**
Train$_{EGY}$	40%	1,721	13.6
Train$_{SUD}$	60%	49	16.7
Train$_{EGY+SUD}$	60%	1,759	13.6
Train$_{EGY'+SUD}$	60%	1,639	13.5
+DEF	60%	1,640	13.5
+PART	60%	49	0.4
Train$_{JOR}$	40%	80	26.7
Train$_{EGY+JOR}$	40%	1,772	13.7
Train$_{EGY'+JOR}$	40%	1,337	12.9
+DEF	40%	1,351	13.0
+PART	40%	95	0.9

Table 6: The number of Rules (**R**) for each training setup using Parla in addition to their ratio, (**R%**), with respect to the training size.

8 Conclusion and Future Work

In this work we investigated cross-dialectal learning of morphophonology of three Arabic dialects – Egyptian, Sudanese, and Jordanian – through rule learning, where we generate a spoken form from an underlying morphological representation. We explored different scenarios of data availability where Egyptian is taken to be the rich-resource dialect while Sudanese and Jordanian are under-resourced. We found that training on the under-resourced dialect alone outperformed transfer from the higher-resourced dialect, alone or in combination with the under-resourced dialect. Furthermore, we introduced learned syllable structure properties as an additional linguistic well-formedness measure, which nearly always boosted performance, particularly when used in the absence of training data from the under-resource dialect.

Some of the analyses suggest that cross-dialectal learning using high resource data that is potentially contradictory with the target dialect is needed. Potential techniques we plan to explore involve reinforcement learning and active learning. We additionally plan on carrying more careful analysis of the rules and how they compare across the dialects. We will also explore incorporating more linguistic signals such as stress assignment since it is closely tied with some phonological processes. Additionally, we are working on investigating more dialects across the continuum as more data become available. Finally, we plan to investigate ways to unify underlying representations in reasonable ways to allow a clearer classification of the rule types across dialects.

Acknowledgments

We thank Robert Hoberman and Ellen Broselow for their helpful discussion and efforts. Khalifa, Qaddoumi, and Rambow gratefully acknowledges support from the Institute for Advanced Computational Science at Stony Brook University.

References

Adam Albright and Bruce Hayes. 2002. Modeling English past tense intuitions with minimal generalization. In *Proceedings of the ACL-02 workshop on Morphological and phonological learning*, pages 58–69.

Evan L Antworth. 1991. Introduction to two-level phonology. *Notes on Linguistics*, 53:4–18.

Pty Ltd, Sydney, and Australia Appen. 2006a. Gulf Arabic conversational telephone speech, transcripts LDC2006T15. Web Download. Philadelphia: Linguistic Data Consortium.

Pty Ltd, Sydney, and Australia Appen. 2006b. Iraqi Arabic conversational telephone speech, transcripts LDC2006T16. Web Download. Philadelphia: Linguistic Data Consortium.

Pty Ltd, Sydney, and Australia Appen. 2007. Levantine Arabic conversational telephone speech, transcripts LDC2007T0. Web Download. Philadelphia: Linguistic Data Consortium.

Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, et al. 2018. The madar arabic dialect corpus and lexicon. In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*.

Ellen Broselow. 1976. *The Phonology of Egyptian Arabic*. Ph.D. thesis, University of Massachusetts Amherst.

Ellen Broselow. 1992. Parametric variation in arabic dialect phonology. *Perspectives on Arabic linguistics IV*, pages 7–45.

Ellen Broselow. 2017. Syllable Structure in the Dialects of Arabic. *The Routledge handbook of Arabic linguistics*, pages 32–47.

Ellen Broselow, Su-I Chen, and Marie Huffman. 1997. Syllable weight: convergence of phonology and phonetics. *Phonology*, 14(1):47–82.

Ellen Broselow, Marie Huffman, Sui-I Chen, and Ruohmei Hsieh. 1995. The timing structure of cvvc syllables. *Amsterdam Studies in the Theory and History of Linguistic Science Series 4*, pages 119–119.

Jane Chandlee, Remi Eyraud, Jeffrey Heinz, Adam Jardine, and Jonathan Rawski. 2019. Learning with partially ordered representations. In *Proceedings of the 16th Meeting on the Mathematics of Language*, pages 91–101, Toronto, Canada. Association for Computational Linguistics.

Kevin Ellis, Adam Albright, Armando Solar-Lezama, Joshua B Tenenbaum, and Timothy J O'Donnell. 2022. Synthesizing theories of human language with bayesian program induction. *Nature communications*, 13(1):1–13.

Samira Farwaneh. 1995. *Directionality effects in Arabic dialect syllable structure*. Ph.D. thesis, The University of Utah.

Hassan Gadalla, Hanaa Kilany, Howaida Arram, Ashraf Yacoub, Alaa El-Habashi, Amr Shalaby, Krisjanis Karins, Everett Rowson, Robert MacIntyre, Paul Kingsbury, David Graff, and Cynthia McLemore. 1997. CALLHOME Egyptian Arabic transcripts LDC97T19. Web Download. Philadelphia: Linguistic Data Consortium.

Nizar Habash, Ramy Eskander, and Abdelati Hawwari. 2012. A Morphological Analyzer for Egyptian Arabic. In *Proceedings of the Workshop of the Special Interest Group on Computational Morphology and Phonology (SIGMORPHON)*, pages 1–9, Montréal, Canada.

Abdel Halim Hamid. 1984. *A Descriptive Analysis of Sudanese Colloquial Arabic Phonology*. Ph.D. thesis, University of Illinois at Urbana-Champaign.

Go Inoue, Salam Khalifa, and Nizar Habash. 2022. Morphosyntactic Tagging with Pre-trained Language Models for Arabic and its Dialects. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1708–1719, Dublin, Ireland. Association for Computational Linguistics.

Salam Khalifa, Jordan Kodner, and Owen Rambow. 2022. Towards learning Arabic morphophonology. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Salam Khalifa, Ossama Obeid, and Nizar Habash. 2021. Character Edit Distance Based Word Alignment.

Salam Khalifa, Sarah Payne, Jordan Kodner, Ellen Broselow, and Owen Rambow. 2023. A cautious generalization goes a long way: Learning morphophonological rules. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1793–1805, Toronto, Canada. Association for Computational Linguistics.

Salam Khalifa, Abdelrahim Qaddoumi, Ellen Broselow, and Owen Rambow. 2024. Picking up where the linguist left off: Mapping morphology to phonology through learning the residuals. In *Proceedings of The Second Arabic Natural Language Processing Conference*, pages 258–264, Bangkok, Thailand. Association for Computational Linguistics.

Salam Khalifa, Nasser Zalmout, and Nizar Habash. 2020. Morphological Analysis and Disambiguation for Gulf Arabic: The Interplay between Resources and Methods. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3895–3904, Marseille, France. European Language Resources Association.

Hanaa Kilany, Hassan Gadalla, Howaida Arram, Ashraf Yacoub, Alaa El-Habashi, and Cynthia McLemore. 2002. Egyptian Colloquial Arabic Lexicon. LDC catalog number LDC99L22.

Jordan Kodner. 2016. Simple Syllabify.

Mohamed Maamouri, Tim Buckwalter, David Graff, and Hubert Jin. 2007. Fisher levantine arabic conversational telephone speech, transcripts ldc2007t04. Web Download. Philadelphia: Linguistic Data Consortium.

Kurt Micallef, Nizar Habash, Claudia Borg, Fadhl Eryani, and Houda Bouamor. 2024. Cross-lingual transfer from related languages: Treating low-resource Maltese as multilingual code-switching. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1014–1025, St. Julian's, Malta. Association for Computational Linguistics.

Wael Salloum and Nizar Habash. 2014. ADAM: Analyzer for Dialectal Arabic Morphology. *Journal of King Saud University - Computer and Information Sciences*, 26(4):372–378.

Toshiyuki Takezawa, Genichiro Kikui, Masahide Mizushima, and Eiichiro Sumita. 2007. Multilingual Spoken Language Corpus Development for Communication Research. *Computational Linguistics and Chinese Language Processing*, 12(3):303–324.

Yang Wang. 2024. *Studies in Morphophonological Copying: Analysis, Experimentation and Modeling*. Ph.D. thesis, University of California, Los Angeles.

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2021. Applying the transformer to character-level transduction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1901–1907, Online. Association for Computational Linguistics.

Nasser Zalmout. 2020. *Morphological Tagging and Disambiguation in Dialectal Arabic Using Deep Learning Architectures*. Ph.D. thesis, New York University.

A Appendix

Arabic	IPA	LDC
إ أ ؤ ئ ء	/ʔa/	’
ب	/b/	b
ي	/j/	j
د	/d/	d
ه	/h/	h
و	/w/	w
ز	/z/	z
ح	/ħ/	H
ط	/tˤ/	T
ي	/y/	y
ك	/k/	k
ل	/l/	l
م	/m/	m
ن	/n/	n
س	/s/	s
ع	/ʕ/	c
ف	/f/	f
ص	/sˤ/	S
ق	/q/	q
ر	/r/	r
ش	/ʃ/	$
ت	/t/	t
ة	/-a(t)/	a,at
ث	/θ/	v
خ	/x/	x
ذ	/ð/	*
ض	/dˤ/	D
غ	/ɣ/	g
ظ	/ðˤ/	Z
*َ	/a/	a
*ُ	/u/	u
*ِ	/i/	i
ا ى	/aː/	A
و	/u:/	U
ي	/i:/	I

Table 7: Transcription Map