Orthographic Transliteration for Kabyle Speech Recognition

Chris Haberland
Mosaix AI
Palo Alto, California
crh2ke@virginia.edu

Ni Lao
Mosaix AI
Palo Alto, California
nlao@cs.cmu.edu

Abstract

Training on graphemes alone without phonemes simplifies the speech-to-text pipeline. However, models respond differently to training on graphemes of different writing systems. We investigate the impact of differences between Latin and Tifinagh orthographies on automatic speech recognition quality on a Kabyle Berber speech corpus. We train on a corpus represented in a Latin orthography marked for vowels and gemination and subsequently transliterate model output to a consonantal Tifinagh orthography not marked for these features, which results in 10% absolute improvement in word error rate over a model trained on the unmarked orthography. We find that this performance gain is primarily due to a reduced error rate for graphemes marked for vocalic and voiced consonantal phonemes. Our results suggest that speech-to-text corpora for languages with alternative defective orthographies may lead to better model quality by being fully marked for vowels and gemination.[1]

1 Introduction

Graphemic modeling units and their correspondence with the spoken word can vary between different language communities (Turki et al., 2016), and even a single language community may have multiple orthographic conventions for application in different contexts (Zitouni, 2014) (diglossia). Minority languages in particular have often undergone less standardization (Jaffe, 2000), contributing to a greater tendency to be written in multiple orthographies. Improving speech technologies to support minority and 'low-resource' languages and orthographies is crucial to ensuring their vitality and their users' access to information in the digital era (Cooper, 2019). Poor quality of low-resource language systems can compel users to interact with ASR systems in languages of which they are non-native, diminishing use of their native language. Furthermore, high error rates for a low-resource language ASR systems disadvantage monolingual speakers of the low-resource language that have a limited ability to switch to systems in more prevalent languages with better recognition quality.

Modern speech-to-text (S2T) models are trained on audio data paired with sequences of modeling units (Davel et al., 2015), which may be graphemes, phonemes, or other representations (Belinkov et al., 2019) that represent the linguistic constituents. Training models on phonemes constitutes a general paradigm in the creation of S2T systems (Yu et al., 2020a) especially in the context of low-resource languages (Besacier et al., 2014). Training on phonemes can be advantageous for decoding out-of-vocabulary words or words from an external language (Hu et al., 2019), but manual annotation of speech data can be prohibitively expensive for low-resource languages (Cooper, 2019).

ASR pipelines often include a component to automatically generate phoneme-based training data through grapheme to phoneme (G2P) conversions (Kubo and Bacchiani, 2020; Chen et al., 2019) by training supervised models (Rao et al., 2015; Jyothi and Hasegawa-Johnson, 2017; Arora et al., 2020) or constructing rule-based systems (Abbas and Asif, 2020a). Recently, there is a trend towards G2P conversion with minimal intervention and preparation to streamline the end-to-end learning process. Several systems have sought to streamline the G2P process using self-training (Hasegawa-Johnson et al., 2019), en-

[1] A repository of our work can be found at https://github.com/berbertranslit/berbertranslit

sembles of varying degrees of supervision (Yu et al., 2020b), and leveraging open dictionaries of higher-resource languages (Deri and Knight, 2016). For low-resource languages, training S2T systems with graphemes alone obviates the G2P step in the S2T pipeline and the need for language-specific expert annotations (Le et al., 2019). However, S2T models respond differently to training on graphemes of different writing systems.

In this paper, we the study the impact of graphemic vowel inclusion, gemination marking, and elision on S2T performance for Kabyle, a Berber language of northern Algeria. We chose to experiment with this language to augment the discussion surrounding orthographic choice on S2T quality that has been conducted primarily on Semitic languages that are comparatively more resourced, such as Arabic. While several previous studies (Alhanai, 2014; Alshayeji et al., 2019; Al-Anzi and AbuZeina, 2017) have demonstrated the effect of training and decoding using defective and non-defective orthographies separately, our study is the first to compare a neural speech model's performance between a) training and decoding in a defective orthography, and b) training on a non-defective orthography and decoding into its defective representation. Our study is also the first to analyze the nature of phonemic errors made by neural ASR models trained on a corpus in a defective and non-defective orthography to understand any systematic difference of types of errors made by models trained on these orthographies. The results demonstrate the importance of including vocalic graphemic inputs for improved S2T recognition of vowels and voiced consonants. To our knowledge, this result represents the first S2T system trained on a Tifinagh-encoded corpus of a Berber language.

2 Related Work

The investigation of orthographic choices on S2T system performance parallels the research on human language comprehension of written text. A significant body of research has sought to uncover how different G2P mappings across writing systems may predict reading level achievement and interactions with dyslexia (Daniels and Share, 2018; Rafat et al., 2019). For example, Law et al. (2018) assess the reading abilities of children diagnosed with dyslexia when taught a novel orthography consisting of new G2P mappings. Maroun et al. (2020) study the effect of diacritization and non-diacritization of dyslexic and non-dyslexic readers' processing of the Arabic script and found spelling knowledge of study participants to be the most significant predictor of processing speed.

S2T learning solely with graphemes has a long history (Eyben et al., 2009). More recently, Wang et al. (2018) report that the phonemic-graphemic performance gap closes when model architecture and hyperparameters are attuned to the specific data input. Rao and Sak (2017) found improved performance of graphemically trained models in multi-accented corpora and in trials of increased input data scale. Other work has tested derivatives of graphemes, such as bytes (Li et al., 2019), wordpieces (Rao and Sak, 2017), and context-dependent graphemes (i.e. chenones) (Le et al., 2019; Wang et al., 2020). Wang et al. (2020) achieved state-of-the-art results on English data with graphemically-derived modeling units for English.

Imputation of diacritics to augment defective model inputs has been, and continues to be, an active area of research (Schone, 2006; Ananthakrishnan et al., 2005; Alqahtani and Diab, 2019; Alqahtani et al., 2019; Darwish et al., 2020). Diacritic imputation systems are designed to help computational models resolve heterophonic homographs, or congruent graphemic sequences that have multiple phonemic interpretations, in orthographies that do not mark certain features. Sequences of this type are prevalent in consonantal writing systems, such as that used for Arabic, in which roughly one-third of tokens may be pronounced differently when not diacritized (Maroun and Hanley, 2017).

There has been work investigating diacritization's effect on speech modeling in languages that are written in defective orthographies, or those not marked for certain phonemes. Afify et al. (2005) used HMMs to demonstrate that training on voweled graphemes could increase performance over training on unvow-

elled graphemes on Arabic broadcast transcripts, even when decoding into unvowelled text. However, to the authors' knowledge, this has not been demonstrated in modern neural speech models. However, more recently, Alhanai (2014) showed that training neural acoustic models *and decoding* into voweled graphemes generally improved WER over unvowelled graphemes. Alsharhan and Ramsay (2019) pre-annotate training transcripts with phonetic information deduced from graphemic context with rules to improve system performance. Alshayeji et al. (2019) and Al-Anzi and AbuZeina (2017) compare diacritized and non-diacritized input with various S2T model architectures and hyperparameters and observe higher WER for diacritized trials, though they do not train on diacritized data and decode on non-diacritized data.

Augmenting inputs via transliteration has been shown to improve S2T systems or machine translation performance. Emond et al. (2018) transliterate model output as a post-process to improve the recognition of code-switched speech. Le and Sadat (2018) and Cho et al. (2020) model the G2P task as a neural sequence-to-sequence model and record improvements in named entity recognition and code-switched speech for Vietnamese and mixed Korean-Chinese scripts, respectively. While these studies use neural G2P models, rule-based systems are commonly developed for under-resourced languages (Ahmadi, 2019; Abbas and Asif, 2020b).

To date, there are limited efforts that apply neural speech models to Berber languages. OCR techniques have been applied to Tifinagh recently (Sadouk et al., 2017; Benaddy et al., 2019), and Lyes et al. (2019) produced a pronunciation dictionary for speech modeling of phonemes. However, to the best of our knowledge, the ASR research community has not documented the training of Berber S2T models aside from those produced from the CommonVoice initiative (Ardila et al., 2019) trained with a Latin-script corpus, although Zealouk et al. (2020) do describe a speech recognition system for Amazigh of Morocco.

3 The Kabyle Language and Berber Writing Systems

Kabyle is a Berber language spoken in northern Algeria that has historically been written in Latin, Arabic, and Tifinagh scripts. Contemporary Kabyle is most widely written in a Latin orthography popularized by the linguist Mouloud Mammeri in a 1976 grammar of the language, though the Arabic and Tifinagh scripts are still promoted among certain groups within Algeria society (Souag, 2019). Souag (2019) contends that the Latin script predominates over the others in modern usage.

The alphabetic Neo-Tifinagh orthograpies came into use after language planning initiatives for the Berber languages in the mid-twentieth century spearheaded by organizations such as Morocco's IRCAM (Amazigh), the Nigerien APT (Tuareg) (Blanco, 2014), and the Académie berbère (Kabyle) (Souag, 2019). The traditional, consonantal Tifinagh orthographies are not commonly used to write Kabyle. However, we transliterate Kabyle into a consonantal orthography to expand the incomplete literature on decoding into defective orthographies, which has primarily focused on Semitic languages. To our knowledge, no prior study has trained or decoded a speech model for a Berber language using a Tifinagh orthography.

We outline the fundamental differences between the Latin Kabyle orthography and the consonantal Tifinagh orthography: the first is that the Latin marks for gemination via digraphs, unlike the traditional Tifinagh. Some consonantal digraphs are spirantized with respect to their singleton counterparts (e.g. 'tt' from 't'). In the Latin orthography, these digraphs are phonemically "tense" and correlate with increased pronunciation length and register a fortis-lenis contrast, including devoicing. They are phonemically distinct from their singleton counterparts and can form minimal pairs (Elias, 2020).

The second fundamental difference is of vowel denotation. Although vowels are written in all contexts in Neo-Tifinagh orthographies, they are not marked save for word-final positions in the traditional Tifinagh orthographies (Elghamis, 2011; Savage, 2008). From the set of Tifinagh characters that may repre-

sent vowels, only '◌ⴰ' exclusively represents non-glide vowels (for 'a', 'ə'[2]), while '◌ⵓ' ('u') and '◌ⵉ' ('i') also represent semi-vowels ('w' and 'j', respectively). These latter two graphemes are analogous to the *matres lectionis* of Semitic language scripts (Posegay, 2020).

A final difference is that certain Tifinagh orthographies make use of ligatures that elide certain combinations of adjacent graphemes. The number of attested ligatures across the many varieties of traditional Tifinagh is vast (Savage, 2008) and most are not supported by Unicode[3]. We test the effect of ligatures by encoding those used in the Ahaggar orthography Elghamis (2011) as distinct characters in trial (1c) described in Section 5.

4 Approach

4.1 Mozilla CommonVoice

We use the original CommonVoice Kabyle corpus for all experiments[4]. The audio-transcript pairs from Mozilla's CommonVoice crowdsourced initiative (Ardila et al., 2019), which has collected data for over 54 languages at the time of writing. All corpora are released with train/dev/test subsets, and a unique speaker may appear in only a single set among each split. Most utterances are derived from Wikipedia, but some have been added by annotators through the language community's Pontoon page[5]. We removed special symbols and normalized Unicode characters of similar graphical appearance to ensure that characters intended to represent a single grapheme were treated as such[6].

4.2 Mozilla DeepSpeech

For S2T model training, we use Mozilla's DeepSpeech pipeline, which is based on the DeepSpeech framework (Hannun et al., 2014) and is maintained by a large community. After parameter tuning we found that the default hyperparameters worked well. For all experiments, we used models of 1024 hidden units and trained for 50 epochs, with a learning rate of .0001 and dropout of 0.3. We used batch sizes of 32, 16, and 16 for train, dev, and test sets, respectively. We used the default trigram settings for training the LM with KenLM (Heafield et al., 2013) in our experiments.

4.3 Transliterator

To convert the Latin-script CommonVoice corpus to the Tifinagh orthographies in our experiments, we use the Graph Transliterator Python package(Pue, 2019). This constructs a directed tree of ranked transition rules (e.g, **mm** -> ⵎ (not ⵎⵎ) because **mm** -> ⵎ ranks before **m** -> ⵎ) to convert between between Latin and Berber orthographies. We write rules for two distinct defective orthographies modelled after Elghamis (2011)'s description of the Ahaggar variant of Tiginagh - one with ligatures, and one without. In cases where multiple Unicode graphemes represent the same phonemes across Berber languages and orthographies (e.g. ⴽ, ⵅ), we opted to use the symbol closest to that described in Elghamis (2011). Heterophonic homographs in the Latin corpus remain as such in the transliterated Tifinagh (e.g. 'd' represents both 'd' and 'ð', and is transliterated as 'ⴷ' and not the IRCAM 'ⴸ'. All Kabyle phonemes that do not have distinct graphemes in the orthography described in Elghamis (2011) are represented with a corresponding Neo-Tifinagh symbol (e.g. č -> ⵛ, ṛ -> ⵕ).

Table 1: Kabyle CommonVoice Data Statistics

Split	Downloaded	Processed	Length
Train	37,056	35,715	35 hrs, 24 min
Dev	11,482	11,100	10 hrs, 52 min
Test	11,483	11,125	11 hrs, 42 min

4.4 Sequence Alignment

We sought to investigate which, and to what degree, phonemic classes are affected by different training orthographies. To facilitate this analysis, we required a tool to align the graphemic output sequences from the ASR systems, such that the aligned character pairs represented the audio data at the same time periods in the input data. Therefore, we conduct a phonemic confusion analysis from the

[2] We do not find attestations of '◌ⵯ' in the traditional Tifinagh orthographies described in Elghamis (2011), so we transliterate word-final 'e' (primarily in loanwords) as '◌ⴰ'

[3] https://www.unicode.org/charts/PDF/U2D30.pdf

[4] Accessed April 2020, 4th ed.

[5] https://pontoon.mozilla.org/projects/common-voice/

[6] E.g., ɛ, and ∈ were converted to ɛ (U+025B)

Table 2: Normalization and transliteration examples

Original	Normalized	Tifinagh Transliteration
D tasnareft taserdasit i yettreṣṣin deg Lezzayer.	**d tasnareft taserdasit i yettreṣṣin deg lezzayer**	ⴰ ⵜⵙⵏⴰⵔⴻⴼⵜ ⵜⵙⴻⵔⴷⴰⵙⵉⵜ ⵉ ⵢⵜⵜⵔⴻⵚⵚⵉⵏ ⴷⴻⴳ ⵍⴻⵣⵣⴰⵢⴻⵔ
Teĉĉiḍ iles-ik waqila?	**teĉĉiḍ iles ik waqila**	ⵜⵜⵛⵛⵉⴹ ⵉⵍⴻⵙ ⵉⴽ ⵡⴰⵇⵉⵍⴰ
Σerḍeɣ-t-id ad yekkes lxiq, yeẓẓel iḍarren.	**ɛerḍeɣ t id ad yekkes lxiq yeẓẓel iḍarren**	ⵄⴻⵔⴹⴻⵖ ⵜ ⵉⴷ ⴰⴷ ⵢⴻⴽⴽⴻⵙ ⵍⵅⵉⵇ ⵢⴻⵥⵥⴻⵍ ⵉⴹⴰⵔⵔⴻⵏ
Tawaɣit d lmeḥna d-yeɣḍel ṭrad ɣef tmurt.	**taɣaɣit d lmeḥna d yeɣḍel ṭrad ɣef tmurt**	ⵜⴰⵖⴰⵖⵉⵜ ⴷ ⵍⵎⴻⵃⵏⴰ ⴷ ⵢⴻⵖⴹⴻⵍ ⵟⵔⴰⴷ ⵖⴻⴼ ⵜⵎⵓⵔⵜ

Table 3: Modelling unit experiment (1c) input example. Note: ⵠ and ⵣ are stand-in single-character substitutions for ligatures that are not represented in Unicode and are not graphically representative of the traditional graphemes for these ligatures

Non-ligatured	ⵑⵠⵒⵔ	ⵚⵣⵠⵑ	ⵣⵜⵒ	ⵔⴰⵜ	ⵄⵑⵠ	ⵒⵜ	ⵜⵏⵑ	ⴰⵚ	ⵜⵎⵑⵜⵡⵜ
Ligatured	ⵒⵒⵔ	ⵚⵣⵠⵑ	ⵣⵜⵒ	ⵔⴰⵜ	ⵄⵒ	ⵒⵜ	ⵜⵏⵑ	ⴰⵚ	ⵜⵎⵠⵣ

graphemes with Sound-Class-Based Phonetic Alignment (SCA) List (2014). This was possible due to the high transparency, or unambiguous correspondence between graphemes to phonemes (Marjou, 2021) of the Kabyle Latin script. We use the *prog_align* function contained in the Lingpy package (List et al., 2019), which constructs a similarity matrix and applies a Neighbor-Joining algorithm (see Saitou and Nei (1987)) to construct a guide tree to successively align phonemic sequences. A dynamic programming routine finds a least-cost path through the matrix to align the two sequences according to similar sound classes. We find that this approach gives reliable alignment for phonemic sequences. We found no errors after manually inspecting a thousand aligned phoneme pairs[7].

5 Experimentation and Results

Now we present our result comparing S2T performance when training on orthographies of varying degrees of phonemic informativeness, and analyzing phonemic confusing using sequence alignment techniques.

5.1 Experiments

First, we test the hypothesis that training and testing upon an orthography unmarked for vowels, as opposed to marked, yields lower ASR word error rates. Experiment 1 compares the effect of training and testing upon the Latin-based orthography and transliterated Tifinagh orthography in a set of trials listed in Table 4 (1a-c). In 1a, the Latin corpus is used for training and testing. The outputs were evaluated against Latin gold utterances in the test split. In 1b, we train in the same manner, but test by applying a transliterator to convert the Latin test set into the consonantal Tifinagh orthography without ligatures. The corpus used to train the language model (LM) is composed of the transliterated utterances of the original corpus. In the third setup (1c), we repeat experiment 1b using a transliterator that models the ligatures described in Section 3. Examples of the ligatured Tifinagh are shown in Table 3.

Secondly, we test the hypothesis that learning from an orthography marked for vowels and decoding on an orthography unmarked for vowels results in lower word error rates compared to training and testing on either of the marked or unmarked orthographies alone. In experiment 2, we test the hypothesis that training on the plene (fully marked) Latin orthography and subsequently decoding into and testing against the defective Tifinagh orthography yields lower error rates compared to both training and testing on the Tifinagh orthography. We train all components on the Latin script and obtain Latin-script output for test utterances as in 1a. However, we then transliterate the output and test against gold utterances transliterated into Tifinagh, as in 1b. Because our main goal is to study the acoustic

[7] https://github.com/berbertranslit/berbertranslit

Table 4: The impact of orthography and language modeling. Group 1: trained and tested on the same orthography types. Group 2: Latin to Tifinagh transliteration at test time given a Latin model. Group 3: the same as Group 1 but without language modeling.

Exp.	Train Orthography	Transliteration	LM	Test Orthography	CER (%)	WER (%)
1a	Latin	no	yes	Latin	29.9	49.9
1b	Tifinagh	no	yes	Tifinagh	35.8	57.9
1c	Tifinagh (ligatured)	no	yes	Tifinagh (ligatured)	33.7	57.4
2	Latin	yes	yes	Tifinagh	29.7	47.4
3a	Latin	no	no	Latin	34.9	78.3
3b	Tifinagh	no	no	Tifinagh	38.8	77.9
3c	Latin	yes	no	Tifinagh	35.6	72.1

Table 5: Alignment of the same sentence produced by different models in Table 4. * indicates a missing space in the alignment. + indicates transliterated gold sequence in Tifinagh.

Group		Raw	Alignment (in IPA representation)
3a - Latin - Latin	Gold	yuweḍ ɣer lebɣi s	j u w ə ḍ ʁ ə r * l ə b * ʁ * i * s
	Pred	yuweḍ ɣaleb ɣ is	j u w ə ḍ ʁ a - * l ə b * ʁ * i * s
3b - Tifinagh - Tifinagh	Gold+	ⵢⵓⵡⴻⴹ ⵖⵔ ⵍⴻⴱⵖⵉ ⵙ	j w ḍ ʁ r * l b * ʁ j s
	Pred	ⵢⵓⵡⴻⴹ ⵍⴻⴱ ⵙ	j w ḍ ʁ - * l b * ʁ - s
3c - Latin - Tifinagh	Gold+	ⵢⵓⵡⴻⴹ ⵖⵔ ⵍⴻⴱⵖⵉ ⵙ	j w ḍ ʁ r * l b * ʁ j s
	Pred	ⵢⵓⵡⴻⴹ ⵍⴻⴱ ⵉ ⵙ	j w ḍ ʁ - * l b * ʁ - s

model and we do not want a small LM training corpus to negatively affect the experimental result, we build the LM in DeepSpeech on all train, dev, and test utterances of the normalized CommonVoice Kabyle Latin-script data for experiments 1 and 2.

Finally, we train the S2T model without a LM as a post-process to specifically understand the sensitivity of the neural speech component. Trials 3a-c replicate 1a-c, but do not apply LM post-processing to help understand the effect of our interventions on the neural ASR component.

5.2 Results

We report the results of all three sets of trials in Table 4. 1a and 1b show that the original Kabyle input encoded in the plene Latin orthography yields lower error rates than when training and testing on the transliterated Tifinagh alone (CER: -5.9%, WER: -8%). However, this reduction is less pronounced when the ligatured Tifinagh orthography is used (1c) (CER: -3.8%, WER: -7.5%).

Trial 2 exhibits improved recognition when training on the Latin orthography and subsequently transliterating to and testing against Tifinagh. This arrangement reduces CER by 0.2% and WER by 2.5% with respect to trial 1a in which the plene orthography was used for both training and testing. Compared to training and testing in the defective orthography (1b), 2 shows a 10.5% absolute decrease in WER and 6.1% absolute decrease in CER.

Trial 3 shows that, without the language model, the WER for training upon and testing against Latin orthography (3a) is greater than when using the Tifinagh orthography (3b) by 0.4%. However, the CER for the former procedure with respect to the latter is less by 3.9%, likely due to the increased difficulty of predicting more characters. Applying a Tifinagh tansliterator to the Latin trained model (3c) resulted in a WER reduction of 6.2% and 5.8% with respect to 3a and 3b. 3c exhibits an improved CER compared to the Tifinagh-only trial (3b) (-3.2%), although it is 0.7% higher when compared to the Latin-only trial (3a).

5.3 Phonemic Confusion Analysis

To understand the orthographies' effects on the speech model we conduct an analysis by alignment between the gold utterances and the predictions from experiments 3b and 3c. This analysis is inspired by recent studies by Kong et al. (2017), Alishahi et al. (2017) and Belinkov et al. (2019), to explore the nature of neural learning of phonemic information.

More specifically we use Lingpy (List et al., 2019) package to determine phone error rates as described in Section 4.4. We translate all graphemes of the gold utterances and their predicted counterparts into sequences of G2P IPA representations and tabulate phoneme class confusions using PHOIBLE's sound classes (Moran and McCloy, 2019). Table 5 shows example aligned sentences produced by this procedure. By analyzing the aligned utterances, we tabulate estimated confusions between the gold and predicted alignments.

We count phonemic disagreements between the models as a proportion of gold target contexts of the aligned matching phoneme. To understand which model achieves better performance for word-final vowel recognition that is denoted in the Tifinagh orthography, we analyze the counts of all gold contexts in which vowels or semi-vowels appear (always word-finally) against the counts of aligned model inferences at these contexts. Table 6 shows that the model trained on the Latin orthography and subsequently transliterated (3c) achieves higher recognition of the pure vowel grapheme compared to the model trained on the unvowelled traditional Tifinagh (3b).

Table 7 compares the errors across several different phonemic classes. We do not consider the 'continuant' and 'delayedRelease' features, as the distinction between allophonic and phonemic fricativity is difficult to determine for Kabyle from graphemes alone. Although the PHOIBLE database includes these features as 'syllabic', we tally counts for the 'approximate', 'sonorant', and 'dorsal', and 'periodic glottal source' features without 'syllabic' phonemes so as to better analyze the contribution of non-syllabic features. McNemar's asymptotic test with continuity correction Edwards (1948) affirms the significance of the difference between 3b and 3c ($P < 0.025$ for all features except the 'geminate' feature).

6 Discussion

Performance when training on plene inputs (3c) to decode word-final vowels improves when compared 3b in which intra-word vowels are hidden from the model. The results suggest that sonorous phonemes benefit more from model training on the voweled text. When only one model between 3b and 3c is correct, we see that 'approximate', 'sonorant', and 'period glottal' phonemes exhibit comparatively high disagreement, surpassed only by the phonemes with positive 'lateral' and 'syllabic' features. The model may share information across these features, and in particular, voicing. All of these features record higher recognition rates in the case of 3c. While the difference in error rates for sonorous and voiced consonants between 3b and 3c does not exactly trend according to the sonority hierarchy (Ladefoged and Johnson, 2014), the number of disagreements between the models does follow this trend. These findings suggest that the model in 3c is leveraging correlates of sonority for phoneme recognition (Figure 1).

A surprising contrast was discovered in the models' differential abilities in detecting coronal and dorsal consonants. We hypothesize that this difference is a function of the differing contexts that these sounds occur in relation to vowels and geminate consonants. The improvement in the 'spread glottis' feature between 3b and 3c is notable, though it is difficult to generalize given the low prevalence of graphemes representing phonemes possessing this feature.

Our study experiments with the DeepSpeech architecture using a single set of hyperparameters for a single data set and language. Future work can investigate the interactions of model architectures, hyperparameters, data scales, G2P mappings, and statistics of orthographic informativeness on S2T performance.

7 Conclusion

Our study is the first to document S2T performance on Tifinagh inputs and shows that the choice of orthography may be consequential for S2T systems trained on graphemes. We amplify findings of prior studies focused on Semitic languages by showing that a Berber S2T model intended to output unvowelled graphemes benefits from training on fully-featured inputs. Our research suggests that ensuring data inputs are fully-featured would improve ASR model quality for languages that conventionally use consonantal orthographies, like Syriac, Hebrew, Persian, and Arabic vernaculars.

Table 6: Comparison of model performance for different word-final vowels. The columns represents phoneme pairs (Tifinagh grapheme : Latin IPA). Trial 3c shows considerably higher recognition of vowels.

	ⴰ : a/ə	ⵉ : i (j)	ⵓ : u/ (w)	All Vowels
The number of word-final vowels in gold	7,430	6,557	1,341	15,328
C_w: The portion (%) of all word-final phonemes	11.7%	10.3%	2.1%	13.0%
C_2: The portion (%) of C_w either 3b (x)or 3c is correct	23.7%	28.1%	30.4%	26.2%
C_3: Both 3b and 3c are incorrect	38.2%	46.8%	34.9%	41.6%
C_{3b}: The portion (%) of C_2 for which 3b is correct	18.5%	13.2%	13.7%	15.6%
C_{3c}: The portion (%) of C_2 for which 3c is correct	**81.5%**	**86.8%**	**86.3%**	**84.5%**

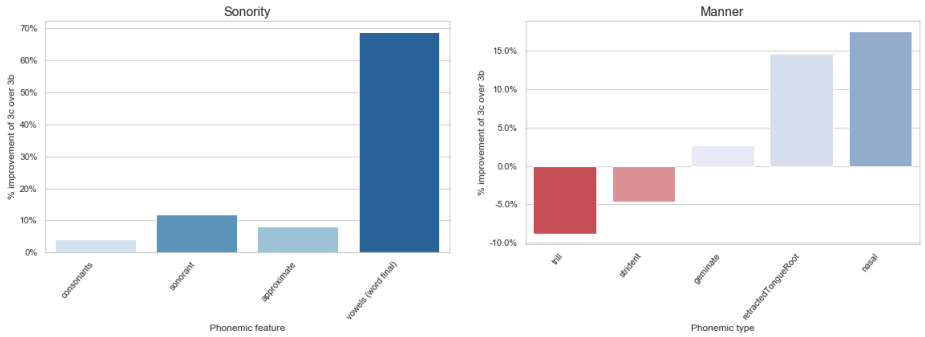

Figure 1: Comparison of the relative error difference between 3b and 3c.

Table 7: Comparison of model performance for different phonemic features. C_p represents the portion (%) of G2P mappings the feature comprises of the total number of G2P mappings in the corpus. See the definition of C_2, C_3, C_{3b} and C_{3c} in Table 6. 3c is correct for more disagreements for all features except for the coronal, strident, and trill features. We use McNemar's asymptotic test with continuity correction Edwards (1948) to test the null hypothesis that there is no difference between the performance of C_{3b} and C_{3c} with respect to different sound classes. χ^2_1 values are particularly high for voiced and syllabic phonemes. We bold the higher between C_{3b} and C_{3c} when $\chi^2_1 > 18.5$ (corresponding to P=0.001).

	C_p	C_2	C_3	C_{3b}	C_{3c}	χ^2_1
consonants	53.1%	16.6%	29.7%	47.9%	**52.1%**	38.9
sonorant (- syllabic)	24.0%	18.1%	26.2%	44.1%	**55.9%**	151.4
approximate (- syllabic)	12.6%	18.6%	28.2%	45.9%	**54.0%**	38.2
nasal	11.5%	17.6%	24.1%	42.0%	**58.0%**	130.1
retracted tongue root	2.1%	16.7%	60.0%	45.8%	54.2%	6.0
labial	11.7%	16.1%	49.8%	35.0%	**65.0%**	26.3
labiodental	1.5%	17.8%	30.3%	40.7%	**59.3%**	23.7
coronal	37.1%	16.4%	29.1%	**51.9%**	48.1%	22.3
strident	8.0%	10.5%	33.5%	**53.8%**	46.2%	12.3
lateral	4.3%	18.6%	30.0%	47.4%	52.6%	5.3
geminate	8.6%	9.0%	56.8%*	49.6%	50.4%	0.13
trill	5.0%	16.3%	30.7%	**55.7%**	44.3%	26.0
dorsal (- syllabic)	11.8%	17.7%	28.7%	38.2%	**61.8%**	294.6
periodic glottal (voiced) (- syllabic)	36.4%	18.5%	29.2%	42.2%	**57.8%**	407.9
spread glottis	0.4%	20.3%	47.1%	34.6%	**65.4%**	18.8
syllabic (vowels) (word-final)	6.1%	26.2%	41.6%	15.6%	**84.4%**	1902.8

References

Muhammad Raihan Abbas and Dr Khadim Hussain Asif. 2020a. Punjabi to iso 15919 and roman transliteration with phonetic rectification. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(2):1–20.

Muhammad Raihan Abbas and Dr. Khadim Hussain Asif. 2020b. Punjabi to iso 15919 and roman transliteration with phonetic rectification. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 19(2).

Mohamed Afify, Long Nguyen, Bing Xiang, Sherif Abdou, and John Makhoul. 2005. Recent progress in arabic broadcast news transcription at bbn. In *Ninth European Conference on Speech Communication and Technology*.

Sina Ahmadi. 2019. A rule-based kurdish text transliteration system. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(2):1–8.

Fawaz S Al-Anzi and Dia AbuZeina. 2017. The effect of diacritization on arabic speech recogntion. In *2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, pages 1–5. IEEE.

Tuka Tuka Waddah Talib Ali Al Hanai Alhanai. 2014. *Lexical and Language Modeling of Diacritics and Morphemes in Arabic Automatic Speech Recognition*. Ph.D. thesis, Massachusetts Institute of Technology.

Afra Alishahi, Marie Barking, and Grzegorz Chrupała. 2017. Encoding of phonology in a recurrent neural model of grounded speech. *arXiv preprint arXiv:1706.03815*.

Sawsan Alqahtani and Mona Diab. 2019. Investigating input and output units in diacritic restoration. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 811–817. IEEE.

Sawsan Alqahtani, Ajay Mishra, and Mona Diab. 2019. Efficient convolutional neural networks for diacritic restoration. *arXiv preprint arXiv:1912.06900*.

Eiman Alsharhan and Allan Ramsay. 2019. Improved arabic speech recognition system through the automatic generation of fine-grained phonetic transcriptions. *Information Processing & Management*, 56(2):343–353.

Mohammad Alshayeji, Sari Sultan, et al. 2019. Diacritics effect on arabic speech recognition. *Arabian Journal for Science and Engineering*, 44(11):9043–9056.

Sankaranarayanan Ananthakrishnan, Shrikanth Narayanan, and Srinivas Bangalore. 2005. Automatic diacritization of arabic transcripts for automatic speech recognition. In *Proceedings of the 4th International Conference on Natural Language Processing*, pages 47–54.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Aryaman Arora, Luke Gessler, and Nathan Schneider. 2020. Supervised grapheme-to-phoneme conversion of orthographic schwas in hindi and punjabi. *arXiv preprint arXiv:2004.10353*.

Yonatan Belinkov, Ahmed Ali, and James Glass. 2019. Analyzing phonetic and graphemic representations in end-to-end automatic speech recognition. *arXiv preprint arXiv:1907.04224*.

Mohamed Benaddy, Othmane El Meslouhi, Youssef Es-saady, and Mustapha Kardouchi. 2019. Handwritten tifinagh characters recognition using deep convolutional neural networks. *Sensing and Imaging*, 20(1):9.

Laurent Besacier, Etienne Barnard, Alexey Karpov, and Tanja Schultz. 2014. Automatic speech recognition for under-resourced languages: A survey. *Speech Communication*, 56:85–100.

Juan Luis Blanco. 2014. Tifinagh & the ircam: Explorations in cursiveness and bicameralism in the tifinagh script. *Unpublished Dissertation, University of Reading*.

Zhehuai Chen, Mahaveer Jain, Yongqiang Wang, Michael L Seltzer, and Christian Fuegen. 2019. Joint grapheme and phoneme embeddings for contextual end-to-end asr. In *INTERSPEECH*, pages 3490–3494.

Won Ik Cho, Seok Min Kim, and Nam Soo Kim. 2020. Towards an efficient code-mixed grapheme-to-phoneme conversion in an agglutinative language: A case study on to-korean transliteration. In *Proceedings of the The 4th Workshop on Computational Approaches to Code Switching*, pages 65–70.

Erica Lindsay Cooper. 2019. *Text-to-speech synthesis using found data for low-resource languages*. Ph.D. thesis, Columbia University.

Peter T Daniels and David L Share. 2018. Writing system variation and its consequences for reading and dyslexia. *Scientific Studies of Reading*, 22(1):101–116.

Kareem Darwish, Ahmed Abdelali, Hamdy Mubarak, and Mohamed Eldesouki. 2020. Arabic diacritic recovery using a feature-rich bilstm model. *arXiv preprint arXiv:2002.01207*.

Marelie Davel, Etienne Barnard, Charl van Heerden, William Hartmann, Damianos Karakos, Richard Schwartz, and Stavros Tsakalidis. 2015. Exploring minimal pronunciation modeling for low resource languages. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Aliya Deri and Kevin Knight. 2016. Grapheme-to-phoneme models for (almost) any language. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 399–408.

Allen L Edwards. 1948. Note on the "correction for continuity" in testing the significance of the difference between correlated proportions. *Psychometrika*, 13(3):185–187.

Ramada Elghamis. 2011. Le tifinagh au niger contemporain: Étude sur l'écriture indigène des touaregs. *Unpublished PhD Thesis, Leiden: Universiteit Leiden*.

Alexander Elias. 2020. Kabyle" double" consonants: Long or strong?

Jesse Emond, Bhuvana Ramabhadran, Brian Roark, Pedro Moreno, and Min Ma. 2018. Transliteration based approaches to improve code-switched speech recognition performance. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 448–455. IEEE.

Florian Eyben, Martin Wöllmer, Björn Schuller, and Alex Graves. 2009. From speech to letters-using a novel neural network architecture for grapheme based asr. In *2009 IEEE Workshop on Automatic Speech Recognition & Understanding*, pages 376–380. IEEE.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.

Mark Hasegawa-Johnson, Camille Goudeseune, and Gina-Anne Levow. 2019. Fast transcription of speech in low-resource languages. *arXiv preprint arXiv:1909.07285*.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696, Sofia, Bulgaria. Association for Computational Linguistics.

Ke Hu, Antoine Bruguier, Tara N Sainath, Rohit Prabhavalkar, and Golan Pundak. 2019. Phoneme-based contextualization for cross-lingual speech recognition in end-to-end models. *arXiv preprint arXiv:1906.09292*.

Alexandra Jaffe. 2000. Introduction: Non-standard orthography and non-standard speech. *Journal of sociolinguistics*, 4(4):497–513.

Preethi Jyothi and Mark Hasegawa-Johnson. 2017. Low-resource grapheme-to-phoneme conversion using recurrent neural networks. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5030–5034. IEEE.

Xiang Kong, Jeung-Yoon Choi, and Stefanie Shattuck-Hufnagel. 2017. Evaluating automatic speech recognition systems in comparison with human perception results using distinctive feature measures. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*.

Yotaro Kubo and Michiel Bacchiani. 2020. Joint phoneme-grapheme model for end-to-end speech recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6119–6123. IEEE.

Peter Ladefoged and Keith Johnson. 2014. *A course in phonetics*. Nelson Education.

Jeremy M Law, Astrid De Vos, Jolijn Vanderauwera, Jan Wouters, Pol Ghesquière, and Maaike Vandermosten. 2018. Grapheme-phoneme learning in an unknown orthography: A study in typical reading and dyslexic children. *Frontiers in psychology*, 9:1393.

Duc Le, Xiaohui Zhang, Weiyi Zheng, Christian Fügen, Geoffrey Zweig, and Michael L Seltzer. 2019. From senones to chenones: Tied context-dependent graphemes for hybrid speech recognition. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 457–464. IEEE.

Ngoc Tan Le and Fatiha Sadat. 2018. Low-resource machine transliteration using recurrent neural networks of asian languages. In *Proceedings of the Seventh Named Entities Workshop*, pages 95–100.

Bo Li, Yu Zhang, Tara Sainath, Yonghui Wu, and William Chan. 2019. Bytes are all you need: End-to-end multilingual speech recognition and synthesis with bytes. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5621–5625. IEEE.

Johann-Mattis List. 2014. *Sequence comparison in historical linguistics*. Ph.D. thesis, Düsseldorf University Press.

Johann-Mattis List, Simon Greenhill, Tiago Tresoldi, and Robert Forkel. 2019. Lingpy. a python library for quantitative tasks in historical linguistics.

Demri Lyes, Falek Leila, and Teffahi Hocine. 2019. Building a pronunciation dictionary for the kabyle language. In *International Conference on Speech and Computer*, pages 309–316. Springer.

Xavier Marjou. 2021. OTEANN: Estimating the transparency of orthographies with an artificial neural network. In *Proceedings of the Third Workshop on Computational Typology and Multilingual NLP*, pages 1–9, Online. Association for Computational Linguistics.

Lateefeh Maroun, Raphiq Ibrahim, and Zohar Eviatar. 2020. Visual and orthographic processing in arabic word recognition among dyslexic and typical readers. *Writing Systems Research*, pages 1–17.

Maryse Maroun and J Richard Hanley. 2017. Diacritics improve comprehension of the arabic script by providing access to the meanings of heterophonic homographs. *Reading and Writing*, 30(2):319–335.

Steven Moran and Daniel McCloy, editors. 2019. *PHOIBLE 2.0*. Max Planck Institute for the Science of Human History, Jena.

Nick Posegay. 2020. Connecting the dots: The shared phonological tradition in syriac, arabic, and hebrew vocalisation. *Studies in Semitic Vocalisation and Reading Traditions*, page 191.

A. Sean Pue. 2019. Graph transliterator: A graph-based transliteration tool. *Journal of Open Source Software*, 4(44):1717.

Yasaman Rafat, Veronica Whitford, Marc Joanisse, Mercedeh Mohaghegh, Natasha Swiderski, Sarah Cornwell, Celina Valdivia, Nasim Fakoornia, Riham Hafez, Parastoo Nasrollahzadeh, et al. 2019. First language orthography influences second language speech during reading: Evidence from highly proficient korean-english bilinguals. In *Proceedings of the International Symposium on Monolingual and Bilingual Speech*, pages 100–107.

Kanishka Rao, Fuchun Peng, Haşim Sak, and Françoise Beaufays. 2015. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4225–4229. IEEE.

Kanishka Rao and Haşim Sak. 2017. Multi-accent speech recognition with hierarchical grapheme based models. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4815–4819. IEEE.

Lamyaa Sadouk, Taoufiq Gadi, and El Hassan Essoufi. 2017. Handwritten tifinagh character recognition using deep learning architectures. In *Proceedings of the 1st International Conference on Internet of Things and Machine Learning*, pages 1–11.

Naruya Saitou and Masatoshi Nei. 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425.

Andrew Savage. 2008. Writing tuareg—the three script options. *International journal of the sociology of language*, 2008(192):5–13.

Patrick Schone. 2006. Low-resource autodiacritization of abjads for speech keyword search. In *Ninth International Conference on Spoken Language Processing*.

Lameen Souag. 2019. Kabyle in arabic script: A history without standardisation. *Creating Standards*, page 273.

Houcemeddine Turki, Emad Adel, Tariq Daouda, and Nassim Regragui. 2016. A conventional orthography for maghrebi arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC), Portoroz, Slovenia*.

Yongqiang Wang, Abdelrahman Mohamed, Due Le, Chunxi Liu, Alex Xiao, Jay Mahadeokar, Hongzhao Huang, Andros Tjandra, Xiaohui Zhang, Frank Zhang, et al. 2020. Transformer-based acoustic modeling for hybrid speech recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6874–6878. IEEE.

Yu Wang, Xie Chen, Mark JF Gales, Anton Ragni, and Jeremy Heng Meng Wong. 2018. Phonetic and graphemic systems for multi-genre broadcast transcription. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5899–5903. IEEE.

Xiang Yu, Ngoc Thang Vu, and Jonas Kuhn. 2020a. Ensemble self-training for low-resource languages: grapheme-to-phoneme conversion and morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 70–78.

Xiang Yu, Ngoc Thang Vu, and Jonas Kuhn. 2020b. Ensemble self-training for low-resource languages: Grapheme-to-phoneme conversion and morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 70–78, Online. Association for Computational Linguistics.

Ouissam Zealouk, Mohamed Hamidi, Hassan Satori, and Khalid Satori. 2020. Amazigh digits

speech recognition system under noise car environment. In *Embedded Systems and Artificial Intelligence*, pages 421–428. Springer.

Imed Zitouni. 2014. *Natural language processing of semitic languages*. Springer.