Polyglot Neural Language Models: A Case Study in Cross-Lingual Phonetic Representation Learning

Yulia Tsvetkov Sunayana Sitaram Manaal Faruqui Guillaume Lample
Patrick Littell David Mortensen Alan W Black Lori Levin Chris Dyer
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA, 15213, USA
{ytsvetko,ssitaram,mfaruqui,glample,plittell,dmortens,awb,lsl,cdyer}@cs.cmu.edu

Abstract

We introduce polyglot language models, recurrent neural network models trained to predict symbol sequences in many different languages using shared representations of symbols and conditioning on typological information about the language to be predicted. We apply these to the problem of modeling phone sequences—a domain in which universal symbol inventories and cross-linguistically shared feature representations are a natural fit. Intrinsic evaluation on held-out perplexity, qualitative analysis of the learned representations, and extrinsic evaluation in two downstream applications that make use of phonetic features show (i) that polyglot models better generalize to held-out data than comparable monolingual models and (ii) that polyglot phonetic feature representations are of higher quality than those learned monolingually.

1 Introduction

Nearly all existing language model (LM) architectures are designed to model one language at a time. This is unsurprising considering the historical importance of count-based models in which every surface form of a word is a separately modeled entity (English *cat* and Spanish *gato* would not likely benefit from sharing counts). However, recent models that use distributed representations—in particular models that share representations across languages (Hermann and Blunsom, 2014; Faruqui and Dyer, 2014; Huang et al., 2015; Lu et al., 2015, *inter alia*)—suggest universal models applicable to multiple languages are a possibility. This paper takes a step in this direction.

We introduce **polyglot language models**: neural network language models that are trained on and applied to any number of languages. Our goals with these models are the following. First, to facilitate data and parameter sharing, providing more training resources to languages, which is especially valuable in low-resource settings. Second, models trained on diverse languages with diverse linguistic properties will better be able to learn naturalistic representations that are less likely to "overfit" to a single linguistic outlier. Finally, polyglot models offer convenience in a multilingual world: a single model replaces dozens of different models.

Exploration of polyglot language models at the sentence level—the traditional domain of language modeling—requires dealing with a massive event space (i.e., the union of words across many languages). To work in a more tractable domain, we evaluate our model on phone-based language modeling, the modeling sequences of *sounds*, rather than words. We choose this domain since a common assumption of many theories of phonology is that all spoken languages construct words from a finite inventory of phonetic symbols (represented conveniently as the elements of the the International Phonetic Alphabet; IPA) which are distinguished by language-universal features (e.g., place and manner of articulation, voicing status, etc.). Although our focus is on sound sequences, our solution can be ported to the semantic/syntactic problem as resulting from adaptation to constraints on semantic/syntactic structure.

This paper makes two primary contributions: in

Proceedings of NAACL-HLT 2016, pages 1357–1366,
San Diego, California, June 12-17, 2016. ©2016 Association for Computational Linguistics

modeling and in applications. In §2, we introduce a novel polyglot neural language model (NLM) architecture. Despite being trained on multiple languages, the multilingual model is more effective (9.5% lower perplexity) than individual models, and substantially more effective than naive baselines (over 25% lower perplexity). Our most effective polyglot architecture conditions not only on the identity of the language being predicted in each sequence, but also on a vector representation of its phono-typological properties. In addition to learning representations of phones as part of the polyglot language modeling objective, the model incorporates features about linguistic typology to improve generalization performance (§3). Our second primary contribution is to show that downstream applications are improved by using polyglot-learned phone representations. We focus on two tasks: predicting adapted word forms in models of cross-lingual lexical borrowing and speech synthesis (§4). Our experimental results (§5) show that in borrowing, we improve over the current state-of-the-art, and in speech synthesis, our features are more effective than manually-designed phonetic features. Finally, we analyze the phonological content of learned representations, finding that our polyglot models discover standard phonological categories such as length and nasalization, and that these are grouped correctly across languages with different phonetic inventories and contrastive features.

2 Model

In this section, we first describe in §2.1 the underlying framework of our model—RNNLM—a standard recurrent neural network based language model (Mikolov et al., 2010; Sundermeyer et al., 2012). Then, in §2.2, we define a Polyglot LM—a modification of RNNLM to incorporate language information, both learned and hand-crafted.

Problem definition. In the phonological LM, *phones* (sounds) are the basic units. Mapping from words to phones is defined in pronunciation dictionaries. For example, “cats” [kæts] is a sequence of four phones. Given a prefix of phones $\phi_1, \phi_2, \ldots, \phi_{t-1}$, the task of the LM is to estimate the conditional probability of the next phone $p(\phi_t \mid \phi_1, \phi_2, \ldots, \phi_{t-1})$.

2.1 RNNLM

In NLMs, a vocabulary V (here, a set of phones composing all word types in the language) is represented as a matrix of parameters $\mathbf{X} \in \mathbb{R}^{d \times |V|}$, with $|V|$ phone types represented as d-dimensional vectors. $\mathbf{X}$ is often denoted as lookup table. Phones in the input sequence are first converted to phone vectors, where ϕ_i is represented by $\mathbf{x}_i$ by multiplying the phone indicator (one-hot vector of length $|V|$) and the lookup table.

At each time step t, most recent phone prefix vector[1] $\mathbf{x}_t$ and hidden state $\mathbf{h}_{t-1}$ are transformed to compute a new hidden representation:

$$\mathbf{h}_t = f(\mathbf{x}_t, \mathbf{h}_{t-1}),$$

where f is a non-linear transformation. In the original RNNLMs (Mikolov et al., 2010), the transformation is such that:

$$\mathbf{h}_t = \tanh(\mathbf{W}_{h_x}\mathbf{x}_t + \mathbf{W}_{h_h}\mathbf{h}_{t-1} + \mathbf{b}_h).$$

To overcome the notorious problem in recurrent neural networks of vanishing gradients (Bengio et al., 1994), following Sundermeyer et al. (2012), in recurrent layer we use long short-term memory (LSTM) units (Hochreiter and Schmidhuber, 1997):[2]

$$\mathbf{h}_t = \mathrm{LSTM}(\mathbf{x}_t, \mathbf{h}_{t-1}).$$

Given the hidden sequence $\mathbf{h}_t$, the output sequence is then computed as follows:

$$p(\phi_t = i \mid \phi_1, \ldots, \phi_{t-1}) = \\ \mathrm{softmax}(\mathbf{W}_{out}\mathbf{h}_t + \mathbf{b}_{out})_i,$$

where $\mathrm{softmax}(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$ ensures a valid probability distribution over output phones.

[1] We are reading at each time step the most recent n-gram context rather than—as is more common in RNNLMs—a single phone context. Empirically, this works better for phone sequences, and we hypothesize that this lets the learner rely on direct connections for local phenomena (which are abundant in phonology) and minimally use the recurrent state to model longer-range effects.

[2] For brevity, we omit the equations describing the LSTM cells; they can be found in (Graves, 2013, eq. 7–11).

2.2 Polyglot LM

We now describe our modifications to RNNLM to account for multilinguality. The architecture is depicted in figure 1. Our task is to estimate the conditional probability of the next phone given the preceding phones and the language (ℓ): $p(\phi_t \mid \phi_1, \ldots, \phi_{t-1}, \ell)$.

In a multilingual NLM, we define a vocabulary V^* to be the union of vocabularies of all training languages, assuming that all language vocabularies are mapped to a shared representation (here, IPA). In addition, we maintain V_ℓ with a special symbol for each language (e.g., $\phi_{english}$, ϕ_{arabic}). Language symbol vectors are parameters in the new lookup table $\mathbf{X}_\ell \in \mathbb{R}^{d \times |\#langs|}$ (e.g., $\mathbf{x}_{english}$, $\mathbf{x}_{arabic}$). The inputs to the Polyglot LM are the phone vectors $\mathbf{x}_t$, the language character vector $\mathbf{x}_\ell$, and the typological feature vector constructed externally $\mathbf{t}_\ell$. The typological feature vector will be discussed in the following section.

The input layer is passed to the hidden local-context layer:

$$\mathbf{c}_t = \mathbf{W}_{c_x}\mathbf{x}_t + \mathbf{W}_{c_{lang}}\mathbf{x}_{lang} + \mathbf{b}_c.$$

The local-context vector is then passed to the hidden LSTM global-context layer, similarly to the previously described RNNLM:

$$\mathbf{g}_t = \mathrm{LSTM}(\mathbf{c}_t, \mathbf{g}_{t-1}).$$

In the next step, the global-context vector $\mathbf{g}_t$ is "factored" by the typology of the training language, to integrate manually-defined language features. To obtain this, we first project the (potentially high-dimensional) $\mathbf{t}_\ell$ into a low-dimensional vector, and apply non-linearity. Then, we multiply the $\mathbf{g}_t$ and the projected language layer, to obtain a global-context-language matrix:

$$\begin{aligned}\mathbf{f}_\ell &= \tanh(\mathbf{W}_\ell \mathbf{t}_\ell + \mathbf{b}_\ell),\\ \mathbf{G}_t^\ell &= \mathbf{g}_t \otimes \mathbf{f}_\ell^\top.\end{aligned}$$

Finally, we vectorize the resulting matrix into a column vector and compute the output sequence as follows:

$$\begin{aligned}p(\phi_t = i \mid \phi_1, \ldots, \phi_{t-1}, \ell) &= \\ &\mathrm{softmax}(\mathbf{W}_{out}\mathrm{vec}(\mathbf{G}_t^\ell) + \mathbf{b}_{out})_i.\end{aligned}$$

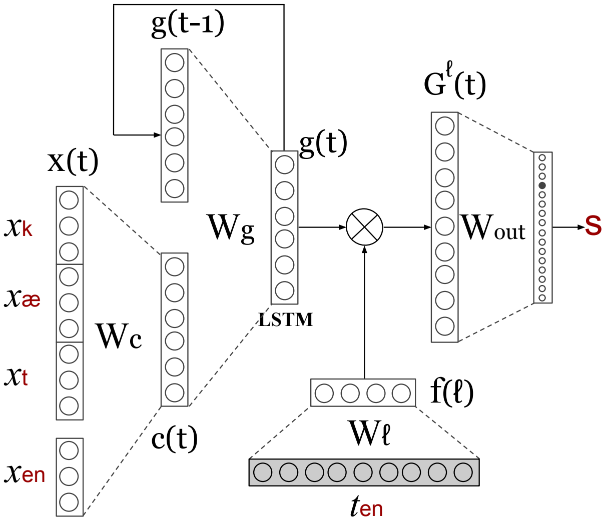

Figure 1: Architecture of the Polyglot LM.

Model training. Parameters of the models are the lookup tables $\mathbf{X}$ and $\mathbf{X}_\ell$, weight matrices $\mathbf{W}_i$, and bias vectors $\mathbf{b}_i$. Parameter optimization is performed using stochastic updates to minimize the categorical cross-entropy loss (which is equivalent to minimizing perplexity and maximizing likelihood): $H(\phi, \hat{\phi}) = -\Sigma_i \hat{\phi}_i \log \phi_i$, where ϕ is predicted and $\hat{\phi}$ is the gold label.

3 Typological features

Typological information is fed to the model via vectors of 190 binary typological features, all of which are phonological (related to sound structure) in their nature. These feature vectors are derived from data from the WALS (Dryer and Haspelmath, 2013), PHOIBLE (Moran et al., 2014), and Ethnologue (Lewis et al., 2015) typological databases via extensive post-processing and analysis.[3] The features primarily concern properties of sound inventories (i.e., the set of phones or phonemes occurring in a language) and are mostly of one of four types:

1. **Single segment represented in an inventory**;

[3] This data resource, which provides standardized phono-typological information for 2,273 languages, is available at https://github.com/dmort27/uriel-phonology/tarball/0.1. It is a subset of the URIEL database, a comprehensive database of typological features encoding syntactic and morphological (as well as phonological) properties of languages. It is available at http://cs.cmu.edu/~dmortens/uriel.html.

e.g., does language ℓ's sound inventory include /g/, a voiced velar stop?

2. **Class of segments represented in an inventory**; e.g., does language ℓ's sound inventory include voiced fricatives like /z/ and /v/?
3. **Minimal contrast represented in an inventory**; e.g., does language ℓ's sound inventory include two sounds that differ only in voicing, such as /t/ and /d/?
4. **Number of sounds representative of a class that are present in an inventory**; e.g., does language ℓ's sound inventory include exactly five vowels?

The motivation and criteria for coding each individual feature required extensive linguistic knowledge and analysis. Consider the case of tense vowels like /i/ and /u/ in "beet" and "boot" in contrast with lax vowels like /ɪ/ and /ʊ/ in "bit" and "book." Only through linguistic analysis does it become evident that (1) all languages have tense vowels—a feature based on the presence of tense vowels is uninformative and that (2) a significant minority of languages make a distinction between tense and lax vowels—a feature based on whether languages display a minimal difference of this kind would be more useful.

4 Applications of Phonetic Vectors

Learned continuous word representations—word vectors—are an important by-product of neural LMs, and these are used as features in numerous NLP applications, including chunking (Turian et al., 2010), part-of-speech tagging (Ling et al., 2015), dependency parsing (Lazaridou et al., 2013; Bansal et al., 2014; Dyer et al., 2015; Watanabe and Sumita, 2015), named entity recognition (Guo et al., 2014), and sentiment analysis (Socher et al., 2013; Wang et al., 2015). We evaluate phone vectors learned by Polyglot LMs in two downstream applications that rely on phonology: modeling lexical borrowing (§4.1) and speech synthesis (§4.2).

4.1 Lexical borrowing

Lexical borrowing is the adoption of words from another language, that inevitably happens when speakers of different languages communicate for a long period of time (Thomason and Kaufman, 2001). Borrowed words—also called *loanwords*—constitute 10–70% of most language lexicons (Haspelmath, 2009); these are content words of foreign origin that are adapted in the language and are not perceived as foreign by language speakers. Computational modeling of cross-lingual transformations of loanwords is effective for inferring lexical correspondences across languages with limited parallel data, benefiting applications such as machine translation (Tsvetkov and Dyer, 2015; Tsvetkov and Dyer, 2016).

In the process of their nativization in a foreign language, loanwords undergo primarily **phonological adaptation**, namely insertion/deletion/substitution of phones to adapt to the phonotactic constraints of the recipient language. If a foreign phone is not present in the recipient language, it is usually replaced with its closest native equivalent—we thus hypothesize that cross-lingual phonological features learned by the Polyglot LM can be useful in models of borrowing to quantify cross-lingual similarities of sounds.

To test this hypothesis, we augment the hand-engineered models proposed by Tsvetkov and Dyer (2016) with features from phone vectors learned by our model. Inputs to the borrowing framework are loanwords (in Swahili, Romanian, Maltese), and outputs are their corresponding "donor" words in the donor language (Arabic, French, Italian, resp.). The framework is implemented as a cascade of finite-state transducers with insertion/deletion/substitution operations on sounds, weighted by high-level conceptual linguistic constraints that are learned in a supervised manner. Given a loanword, the system produces a candidate donor word with lower ranked violations than other candidates, using the shortest path algorithm. In the original borrowing model, insertion/deletion/substitution operations are unweighted. In this work, we integrate transition weights in the phone substitution transducers, which are cosine distances between phone vectors learned by our model. Our intuition is that similar sounds appear in similar contexts, even if they are not present in the same language (e.g., /s^{ʕ}/ in Arabic is adapted to /s/ in Swahili). Thus, if our model effectively captures cross-lingual signals, similar sounds should have smaller distances in the vector space, which can improve the shortest path results. Figure 2 illustrates our modifications to the original framework.

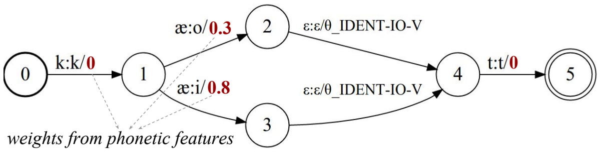

Figure 2: Distances between phone vectors learned by the Polyglot LM are integrated as substitution weights in the lexical borrowing transducers. An English word *cat* [kæt] is adapted to its Russian counterpart кот [kot]. The transducer has also an erroneous path to кит [kit] 'whale'. In the original system, both paths are weighted with the same feature IDENT-IO-V, firing on vowel substitution. Our modification allows the borrowing model to identify more plausible paths by weighting substitution operations.

4.2 Speech synthesis

Speech synthesis is the process of converting text into speech. It has various applications, such as screen readers for the visually impaired and hands-free voice based systems. Text-to-speech (TTS) systems are also used as part of speech-to-speech translation systems and spoken dialog systems, such as personal digital assistants. Natural and intelligible TTS systems exist for a number of languages in the world today. However, building TTS systems remains prohibitive for many languages due to the lack of linguistic resources and data.

The language-specific resources that are traditionally used for building TTS systems in a new language are: (1) audio recordings with transcripts; (2) pronunciation lexicon or letter to sound rules; and (3) a phone set definition. Standard TTS systems today use phone sets designed by experts. Typically, these phone sets also contain phonetic features for each phoneme, which are used as features in models of the spectrum and prosody. The phonetic features available in standard TTS systems are multidimensional vectors indicating various properties of each phoneme, such as whether it is a vowel or consonant, vowel length and height, place of articulation of a consonant, etc. Constructing these features by hand can be labor intensive, and coming up with such features automatically may be useful in low-resource scenarios.

In this work, we replace manually engineered phonetic features with phone vectors, which are then used by classification and regression trees for modeling the spectrum. Each phoneme in our phone set is assigned an automatically constructed phone vector, and each member of the phone vector is treated as a phoneme-level feature which is used in place of the manually engineered phonetic features. While prior work has explored TTS augmented with acoustic features (Watts et al., 2015), to the best of our knowledge, we are the first to replace manually engineered phonetic features in TTS systems with automatically constructed phone vectors.

5 Experiments

Our experimental evaluation of our proposed polyglot models consists of two parts: (i) an intrinsic evaluation where phone sequences are modeled with independent models and (ii) an extrinsic evaluation of the learned phonetic representations. Before discussing these results, we provide details of the data resources we used.

5.1 Resources and experimental setup

Resources. We experiment with the following languages: Arabic (AR), French (FR), Hindi (HI), Italian (IT), Maltese (MT), Romanian (RO), Swahili (SW), Tamil (TA), and Telugu (TE). In our language modeling experiments, two main sources of data are pronunciation dictionaries and typological features described in §3. The dictionaries for AR, FR, HI, TA, and TE are taken from in-house speech recognition/synthesis systems. For remaining languages, the dictionaries are automatically constructed using the Omniglot grapheme-to-IPA conversion rules.[4]

We use two types of pronunciation dictionaries: (1) AR, FR, HI, IT, MT, RO, and SW dictionaries used in experiments with lexical borrowing; and (2) EN, HI, TA, and TE dictionaries used in experiments with speech synthesis. The former are mapped to IPA, with the resulting phone vocabulary size—the number of distinct phones across IPA dictionaries—of 127 phones. The latter are encoded using the UniTran universal transliteration resource (Qian et al., 2010), with a vocabulary of 79 phone types.

From the (word-type) pronunciation dictionaries, we remove 15% of the words for development, and a further 10% for testing; the rest of the data is

[4] http://omniglot.com/writing/

	AR	FR	HI	IT	MT	RO	SW
train	1,868/18,485	238/1,851	193/1,536	988/901	114/1,152	387/4,661	659/7,239
dev	366/3,627	47/363	38/302	19/176	22/226	76/916	130/1,422
test	208/2,057	27/207	22/173	11/100	13/128	43/524	73/806

Table 1: Train/dev/test counts for IPA pronunciation dictionaries for words (phone sequences) and phone tokens, in thousands: #thousands of sequences/# thousands of tokens.

	EN	HI	TA	TE
train	101/867	191/1,523	74/780	71/690
dev	20/169	37/300	14/152	14/135
test	11/97	21/171	8/87	8/77

Table 2: Train/dev/test statistics for UniTran pronunciation dictionaries for words (phone sequences) and phone tokens, in thousands: #thousands of sequences/# thousands of tokens.

used to train the models. In tables 1 and 2 we list—for both types of pronunciation dictionaries—train/dev/test data statistics for words (phone sequences) and phone tokens. We concatenate each phone sequence with beginning and end symbols (<*s*>, <*/s*>).

Hyperparameters. We used the following network architecture: 100-dimensional phone vectors, with hidden local-context and LSTM layers of size 100, and hidden language layer of size 20. All language models were trained using the left context of 3 phones (4-gram LMs). Across all language modeling experiments, parameter optimization was performed on the dev set using the Adam algorithm (Kingma and Ba, 2014) with mini-batches of size 100 to train the models for 5 epochs.

5.2 Intrinsic perplexity evaluation

Perplexity is the standard evaluation measure for language models, which has been shown to correlate strongly with error rates in downstream applications (Klakow and Peters, 2002). We evaluated perplexities across several architectures, and several monolingual and multilingual setups. We kept the same hyper-parameters across all setups, as detailed in §5. Perplexities of LMs trained on the two types of pronunciation dictionaries were evaluated separately; table 3 summarizes perplexities of the models trained on IPA dictionaries, and table 4 summarizes perplexities of the UniTran LMs.

In columns, we compare three model architectures: *baseline* denotes the standard RNNLM architecture described in §2.1; +*lang* denotes the Polyglot LM architecture described in §2.2 with input language vector but without typological features and language layer; finally, +*typology* denotes the full Polyglot LM architecture. This setup lets us separately evaluate the contribution of modified architecture and the contribution of auxiliary set of features introduced via the language layer.

Test languages are IT in table 3, and HI in table 4. The rows correspond to different sets of training languages for the models: *monolingual* is for training and testing on the same language; +*similar* denotes training on three typologically similar languages: IT, FR, RO in table 3, and HI, TA, TE in table 4; +*dissimilar* denotes training on four languages, three similar and one typologically dissimilar language, to evaluate robustness of multilingual systems to diverse types of data. The final sets of training languages are IT, FR, RO, HI in table 3, and HI, TA, TE, EN in table 4.

	Perplexity (↓)		
training set	baseline	+lang	+typology
monolingual	4.36	–	–
+similar	5.73	4.93	**4.24** (↓ 26.0%)
+dissimilar	5.88	4.98	**4.41** (↓ 25.0%)

Table 3: Perplexity experiments with IT as test language. Training languages: monolingual: IT; +similar: IT, FR, RO; +dissimilar: IT, FR, RO, HI.

	Perplexity (↓)		
training set	baseline	+lang	+typology
monolingual	3.70	–	–
+similar	4.14	3.78	**3.35** (↓ 19.1%)
+dissimilar	4.29	3.82	**3.42** (↓ 20.3%)

Table 4: Perplexity experiments with HI as test language. Training languages: monolingual: HI; +similar: HI, TA, TE; +dissimilar: HI, TA, TE, EN.

We see several patterns of results. First, polyglot models require, unsurprisingly, information about what language they are predicting to obtain good modeling performance. Second, typological information is more valuable than letting the model learn representations of the language along with the characters. Finally, typology-augmented polyglot models outperform their monolingual baseline, providing evidence in support of the hypothesis that cross-lingual evidence is useful not only for learning cross-lingual representations and models, but monolingual ones as well.

5.3 Lexical borrowing experiments

We fully reproduced lexical borrowing models described in (Tsvetkov and Dyer, 2016) for three language pairs: AR–SW, FR–RO, and IT–MT. Train and test corpora are donor–loanword pairs in the language pairs. Corpora statistics are given in table 5 (note that these are extremely small data sets; thus small numbers of highly informative features a necessary for good generalization). We use the reproduced systems as the baselines, and compare these to the corresponding systems augmented with phone vectors, as described in §4.1.

	AR–SW	FR–RO	IT–MT
train	417	282	425
test	73	50	75

Table 5: Number of training and test pairs the the borrowing datasets.

Integrated vectors were obtained from a single polyglot model with typology, trained on all languages with IPA dictionaries. For comparison with the results in table 3, perplexity of the model on the IT dataset (used for evaluation is §5.2) is 4.16, even lower than in the model trained on four languages. To retrain the high-level conceptual linguistic features learned by the borrowing models, we initialized the augmented systems with feature weights learned by the baselines, and retrained. Final weights were established using cross-validation. Then, we evaluated the accuracy of the augmented borrowing systems on the held-out test data.

Accuracies are shown in table 6. We observe improvements of up to 5% in accuracies of FR–RO and IT–MT pairs. Effectiveness of the same polyglot model trained on multiple languages and integrated in different downstream systems supports our assumption that the model remains stable and effective with addition of languages. Our model is less effective for the AR–SW language pair. We speculate that the results are worse, because this is a pair of (typologically) more distant languages; consequently, the phonological adaptation processes that happen in loanword assimilation are more complex than mere substitutions of similar phones that we are targeting via the integration of phone vectors.

	Accuracy (↑)		
	AR–SW	FR–RO	IT–MT
baseline	**48.4**	75.6	83.3
+multilingual	46.9	**80.6**	**87.1**

Table 6: Accuracies of the baseline models of lexical borrowing and the models augmented with phone vectors. In all the experiments, we use vectors from a single Polyglot LM model trained on AR, SW, FR, RO, IT, MT.

5.4 Speech synthesis experiments

A popular objective metric for measuring the quality of synthetic speech is the Mel Cepstral Distortion (MCD) (Hu and Loizou, 2008). The MCD metric calculates an L2 norm of the Mel Frequency Cepstral Coefficients (MFCCs) of natural speech from a held out test set, and synthetic speech generated from the same test set. Since this is a distance metric, a lower value of MCD suggests better synthesis. The MCD is a database-specific metric, but experiments by Kominek et al. (Kominek et al., 2008) have shown that a decrease in MCD of 0.08 is perceptually significant, and a decrease of 0.12 is equivalent to doubling the size of the TTS database. In our experiments, we use MCD to measure the relative improvement obtained by our techniques.

We conducted experiments on the IIIT-H Hindi voice database (Prahallad et al., 2012), a 2 hour single speaker database recorded by a professional male speaker. We used the same front end (UniTran) to build all the Hindi TTS systems, with the only difference between the systems being the presence or absence of phonetic features and our vectors. For all our voice-based experiments, we built CLUSTERGEN Statistical Parametric Synthesis voices (Black, 2006) using the Festvox voice building tools (Black and Lenzo, 2003) and the Festival speech synthesis engine (Black and Taylor, 1997).

The baseline TTS system was built using no phonetic features. We also built a TTS system with standard hand-crafted phonetic features. Table 7 shows the MCD for the HI baseline, the standard TTS with hand-crafted features, and augmented TTS systems built using monolingual and multilingual phone vectors constructed with Polyglot LMs.

	MCD (↓)
baseline	4.58
+monolingual	4.40
+multilingual	**4.39**
+hand-crafted	4.41

Table 7: MCD for the HI TTS systems. Polyglot LM training languages: monolingual: HI; +multilingual: HI, TA, TE, EN.

Our multilingual vectors outperform the baseline, with a significant decrease of 0.19 in MCD. Crucially, TTS systems augmented with the Polyglot LM phone vectors outperform also the standard TTS with hand-crafted features. We found that using both feature sets added no value, suggesting that learned phone vectors are capturing information that is equivalent to the hand-engineered vectors.

5.5 Qualitative analysis of vectors

Phone vectors learned by Polyglot LMs are mere sequences of real numbers. An interesting question is whether these vectors capture linguistic (phonological) qualities of phones they are encoding. To analyze to what extent our vectors capture linguistic properties of phones, we use the QVEC—a tool to quantify and interpret linguistic content of vector space models (Tsvetkov et al., 2015). The tool aligns dimensions in a matrix of learned distributed representations with dimensions of a hand-crafted linguistic matrix. Alignments are induced via correlating columns in the distributed and the linguistic matrices. To analyze the content of the distributed matrix, annotations from the linguistic matrix are projected via the maximally-correlated alignments.

We constructed a phonological matrix in which 5,059 rows are IPA phones and 21 columns are boolean indicators of universal phonological properties, e.g. *consonant*, *voiced*, *labial*.[5] We the projected annotations from the linguistic matrix and manually examined aligned dimensions in the phone vectors from §5.3 (trained on six languages). In the maximally-correlated columns—corresponding to linguistic features *long*, *consonant*, *nasalized*—we examined phones with highest coefficients. These were: [ɐː, ʊː, iː, ɔː, ɛː] for *long*; [v, ɲ, d͡ʒ, d, f, j, t͡s, ŋ] for *consonant*; and [ɔ̃, ɛ̃, ɑ̃, œ̃] for *nasalized*. Clearly, the learned representation discover standard phonological features. Moreover, these top-ranked sounds are not grouped by a single language, e.g., /d͡ʒ/ is present in Arabic but not in French, and /ɲ, ŋ/ are present in French but not in Arabic. From this analysis, we conclude that (1) the model discovers linguistically meaningful phonetic features; (2) the model induces meaningful related groupings across languages.

6 Related Work

Multilingual language models. Interpolation of monolingual LMs is an alternative to obtain a multilingual model (Harbeck et al., 1997; Weng et al., 1997). However, interpolated models still require a trained model per language, and do not allow parameter sharing at training time. Bilingual language models trained on concatenated corpora were explored mainly in speech recognition (Ward et al., 1998; Wang et al., 2002; Fügen et al., 2003). Adaptations have been proposed to apply language models in bilingual settings in machine translation (Niehues et al., 2011) and code switching (Adel et al., 2013). These approaches, however, require adaptation to every pair of languages, and an adapted model cannot be applied to more than two languages.

Independently, Ammar et al. (2016) used a different polyglot architecture for multilingual dependency parsing. This work has also confirmed the utility of polyglot architectures in leveraging multilinguality.

Multimodal neural language models. Multimodal language modeling is integrating image/video modalities in text LMs. Our work is inspired by the neural multimodal LMs (Kiros and Salakhutdinov, 2013; Kiros et al., 2015), which defined language models conditional on visual contexts, although we use a different language model architecture (recurrent vs. log-bilinear) and a different approach to gat-

[5] This matrix is described in Littell et al. (2016) and is available at `https://github.com/dmort27/panphon/`.

ing modality.

7 Conclusion

We presented a novel *multilingual* language model architecture. The model obtains substantial gains in perplexity, and improves downstream text and speech applications. Although we focus on phonology, our approach is general, and can be applied in problems that integrate divergent modalities, e.g., topic modeling, and multilingual tagging and parsing.

Acknowledgments

This work was supported by the National Science Foundation through award IIS-1526745 and in part by the Defense Advanced Research Projects Agency (DARPA) Information Innovation Office (I2O). Program: Low Resource Languages for Emergent Incidents (LORELEI). Issued by DARPA/I2O under Contract No. HR0011-15-C-0114.

References

Heike Adel, Ngoc Thang Vu, and Tanja Schultz. 2013. Combination of recurrent neural networks and factored language models for code-switching language modeling. In *Proc. ACL*, pages 206–211.

Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2016. Many languages, one parser. *CoRR*, abs/1602.01595.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proc. ACL*.

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166.

Alan W Black and Kevin A Lenzo. 2003. Building synthetic voices. `http://festvox.org/bsv/`.

Alan W Black and Paul Taylor. 1997. The Festival speech synthesis system: system documentation. Technical report, Human Communication Research Centre, University of Edinburgh.

Alan W Black. 2006. CLUSTERGEN: a statistical parametric synthesizer using trajectory modeling. In *Proc. Interspeech*.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology. `http://wals.info/`.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In *Proc. ACL*.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proc. EACL*.

Christian Fügen, Sebastian Stuker, Hagen Soltau, Florian Metze, and Tanja Schultz. 2003. Efficient handling of multilingual language models. In *Proc. ASRU*, pages 441–446.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850.

Jiang Guo, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Revisiting embedding features for simple semi-supervised learning. In *Proc. EMNLP*.

Stefan Harbeck, Elmar Nöth, and Heinrich Niemann. 1997. Multilingual speech recognition. In *Proc. 2nd SQEL Workshop on Multi-Lingual Information Retrieval Dialogs*, pages 9–15.

Martin Haspelmath. 2009. Lexical borrowing: concepts and issues. *Loanwords in the World's Languages: a comparative handbook*, pages 35–54.

Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual Models for Compositional Distributional Semantics. In *Proc. ACL*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Yi Hu and Philipos C Loizou. 2008. Evaluation of objective quality measures for speech enhancement. *Audio, Speech, & Language Processing*, 16(1):229–238.

Kejun Huang, Matt Gardner, Evangelos Papalexakis, Christos Faloutsos, Nikos Sidiropoulos, Tom Mitchell, Partha P. Talukdar, and Xiao Fu. 2015. Translation invariant word embeddings. In *Proc. EMNLP*, pages 1084–1088.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Ryan Kiros and Ruslan Salakhutdinov. 2013. Multimodal neural language models. In *Proc. NIPS Deep Learning Workshop*.

Ryan Kiros, Ruslan Salakhutdinov, and Richard Zemel. 2015. Unifying visual-semantic embeddings with multimodal neural language models. *TACL*.

Dietrich Klakow and Jochen Peters. 2002. Testing the correlation of word error rate and perplexity. *Speech Communication*, 38(1):19–28.

John Kominek, Tanja Schultz, and Alan W Black. 2008. Synthesizer voice quality of new languages calibrated with mean Mel Cepstral Distortion. In *Proc. SLTU*, pages 63–68.

Angeliki Lazaridou, Eva Maria Vecchi, and Marco Baroni. 2013. Fish transporters and miracle homes: How compositional distributional semantics can help NP parsing. In *Proc. EMNLP*.

M Paul Lewis, Gary F Simons, and Charles D Fennig. 2015. *Ethnologue: Languages of the world*. Texas: SIL International. http://www.ethnologue.com.

Wang Ling, Tiago Luís, Luís Marujo, Ramón Fernandez Astudillo, Silvio Amir, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *Proc. NAACL*.

Patrick Littell, David Mortensen, Kartik Goyal, Chris Dyer, and Lori Levin. 2016. Bridge-language capitalization inference in Western Iranian: Sorani, Kurmanji, Zazaki, and Tajik. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC'16)*.

Ang Lu, Weiran Wang, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2015. Deep multilingual correlation for improved word embeddings. In *Proc. NAACL*.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Proc. Interspeech*, pages 1045–1048.

Steven Moran, Daniel McCloy, and Richard Wright, editors. 2014. *PHOIBLE Online*. Max Planck Institute for Evolutionary Anthropology. http://phoible.org/.

Jan Niehues, Teresa Herrmann, Stephan Vogel, and Alex Waibel. 2011. Wider context by using bilingual language models in machine translation. In *Proc. WMT*, pages 198–206.

Kishore Prahallad, E. Naresh Kumar, Venkatesh Keri, S. Rajendran, and Alan W Black. 2012. The IIIT-H Indic speech databases. In *Proc. Interspeech*.

Ting Qian, Kristy Hollingshead, Su-youn Yoon, Kyoung-young Kim, Richard Sproat, and Malta LREC. 2010. A Python toolkit for universal transliteration. In *Proc. LREC*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. EMNLP*.

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. LSTM neural networks for language modeling. In *Proc. Interspeech*.

Sarah Grey Thomason and Terrence Kaufman. 2001. *Language contact*. Edinburgh University Press Edinburgh.

Yulia Tsvetkov and Chris Dyer. 2015. Lexicon stratification for translating out-of-vocabulary words. In *Proc. ACL*, pages 125–131.

Yulia Tsvetkov and Chris Dyer. 2016. Cross-lingual bridges with models of lexical borrowing. *JAIR*, 55:63–93.

Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Guillaume Lample, and Chris Dyer. 2015. Evaluation of word vector representations by subspace alignment. In *Proc. EMNLP*. https://github.com/ytsvetko/qvec.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proc. ACL*.

Zhirong Wang, Umut Topkara, Tanja Schultz, and Alex Waibel. 2002. Towards universal speech recognition. In *Proc. ICMI*, page 247.

Xin Wang, Yuanchao Liu, Chengjie Sun, Baoxun Wang, and Xiaolong Wang. 2015. Predicting polarities of tweets by composing word embeddings with long short-term memory. In *Proc. ACL*, pages 1343–1353.

Todd Ward, Salim Roukos, Chalapathy Neti, Jerome Gros, Mark Epstein, and Satya Dharanipragada. 1998. Towards speech understanding across multiple languages. In *Proc. ICSLP*.

Taro Watanabe and Eiichiro Sumita. 2015. Transition-based neural constituent parsing. In *Proc. ACL*.

Oliver Watts, Zhizheng Wu, and Simon King. 2015. Sentence-level control vectors for deep neural network speech synthesis. In *Proc. Interspeech*.

Fuliang Weng, Harry Bratt, Leonardo Neumeyer, and Andreas Stolcke. 1997. A study of multilingual speech recognition. In *Proc. EUROSPEECH*, pages 359–362.