Phonologically Aware Neural Model for Named Entity Recognition in Low Resource Transfer Settings

Akash Bharadwaj   **David Mortensen**   **Chris Dyer**   **Jaime G. Carbonell**
{akashb, dmortens, cdyer, jgc}@cs.cmu.edu
Language Technologies Institute
Carnegie Mellon University

Abstract

Named Entity Recognition is a well established information extraction task with many state of the art systems existing for a variety of languages. Most systems rely on language specific resources, large annotated corpora, gazetteers and feature engineering to perform well monolingually. In this paper, we introduce an attentional neural model which only uses language universal phonological character representations with word embeddings to achieve state of the art performance in a monolingual setting using supervision and which can quickly adapt to a new language with minimal or no data. We demonstrate that phonological character representations facilitate cross-lingual transfer, outperform orthographic representations and incorporating both attention and phonological features improves statistical efficiency of the model in 0-shot and low data transfer settings with no task specific feature engineering in the source or target language.

1 Introduction

Named Entity Recognition (NER) (Nadeau and Sekine, 2007; Marrero et al., 2013) is an information extraction task that deals with finding and classifying entities in text into a fixed set of types of interest. It is challenging for a variety of reasons. Named Entities (NEs) can be arbitrarily synthesized (eg. people's/organization's names). NEs are often not subject to uniform cross-linguistic spelling conventions: compare *France* (English) and *Frantsiya* (Uzbek). NEs occur rarely in data which makes generalization difficult. Skewed class statistics necessitate measures to prevent models from merely favoring a majority class.

Named entities must also be annotated in context (eg. "[New York Times]$_{\text{ORG}}$" vs. "[New York]$_{\text{LOC}}$"). Lexical ambiguity (Turkey—country vs. bird), semantic ambiguity ("I work at the [New York Times]$_{\text{ORG}}$" vs. "I read the New York Times") and sparsity induced by morphology add complexity.

Despite the challenges mentioned above, competent monolingual systems that rely on having sufficient annotated data, knowledge and resources available for engineering features have been developed. A more challenging task is to design a model that retains competence in monolingual scenarios and can easily be transferred to a low resource language with minimum overhead in terms of data annotation requirements and feature engineering. This transfer setting introduces additional challenges such as varying character usage conventions across languages with same script, differing scripts, lack of NE transliteration, varying morphology, different lexicons and mutual non-intelligibility to name a few.

We propose the following changes over prior work (Lample et al., 2016) to address the challenges of the low-resource transfer setting. We use:

1. Language universal phonological character representations instead of orthographic ones.

2. Attention over characters of a word while labeling it with an NE category.

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1462–1472,
Austin, Texas, November 1-5, 2016.

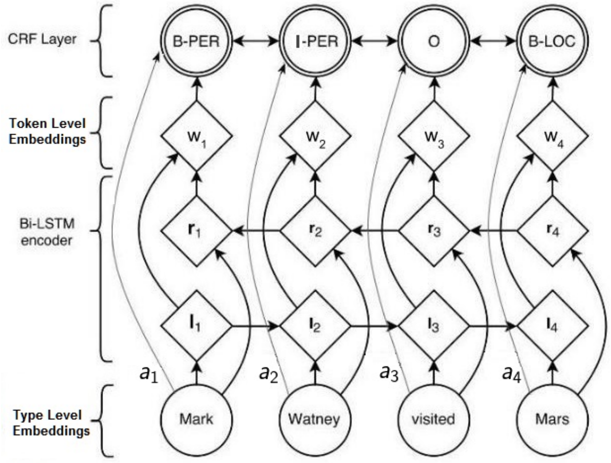

Figure 1: Attentional LSTM-CRF architecture. l_i denotes the encoding of a word and its left context (forward LSTM) while r_i includes only right context (backward LSTM). Inputs to word LSTMs are obtained using character LSTMs and word-embeddings. a_i denotes an attentional context vector concatenated with l_i and r_i.

We show that using phonological character representations instead does not negatively impact performance on two languages: Spanish and Turkish. We then show that using global phonological representations enables model transfer from one or more source languages to a target language with no extra effort, even when the languages use different scripts. We demonstrate that while attention over characters of words has marginal utility in monolingual and high resource settings, it greatly improves the statistical efficiency of the model in 0-shot and low resource transfer settings. We do require a mapping from a language's script to phonological feature space which is script specific and not task specific. This presents little or no overhead due to existence of tools like PanPhon (Littell et al., 2016).

2 Our Approach

Figure 1 provides a high level overview of our model. We model the words of a sentence at the type level and the token level. At the type level (ignorant of sentential context), we use bidirectional character LSTMs as in figure 2 to compose characters of a word to obtain its word representation and concatenate this with a word embedding that captures distributional semantics. This can memorize entities or capture morphological and suffixal clues that can help at a discriminative task like NER. We compose type level word representations with bidirectional LSTMs to obtain token level (cognizant of sentential context) representations. Using token level word representations along with an attentional context vector for each word based on the sequence of characters it contains, we generate score functions used by a Conditional Random Field (CRF) for inference. To facilitate transfer across languages with different scripts, we use Epitran [1] and PanPhon (Littell et al., 2016).

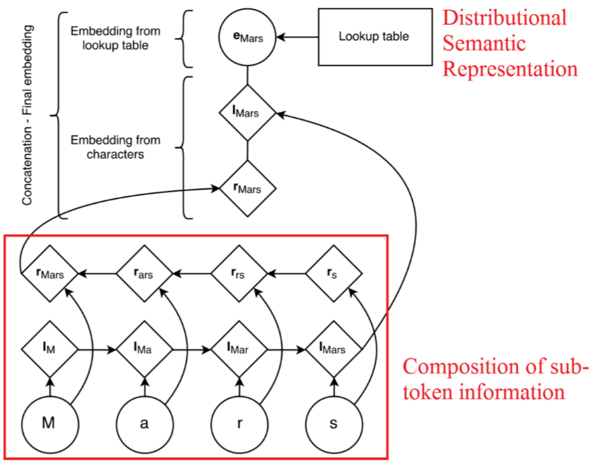
Concatenation - Final embedding
Embedding from lookup table
Embedding from characters
e_{Mars}
Lookup table
Distributional Semantic Representation
l_{Mars}
r_{Mars}
r_{Mars}
r_{ars}
r_{rs}
r_s
l_M
l_{Ma}
l_{Mar}
l_{Mars}
M
a
r
s
Composition of sub-token information

Figure 2: Type level word representations - l denotes a word prefix encoding (by forward char LSTM) while r denotes a word suffix encoding (by backward char LSTM).

Epitran is a straightforward orthography-to-IPA (International Phonetic Alphabet [language universal]) system including a collection of preprocessors and grapheme-to-phoneme mappings for a variety of language-script pairs. It converts a word from its native script into a sequence of IPA characters, each of which approximately corresponds to a phoneme. PanPhon is a database of IPA-to-phonological feature vector mappings and a library for querying, manipulating, and exploiting this database. It consumes the output of Epitran and produces feature vectors (21 dimensions) of phonological characteristics such as whether a phoneme is articulated with (accompanied by) vibration of the vocal folds (voiced), with the tongue in a high, low, back, or front position, with the lips rounded or unrounded, with tongue tip or blade (coronal), etc. Figure 3 depicts the sequence of operations applied to the same NE

[1] https://github.com/dmort27/epitran

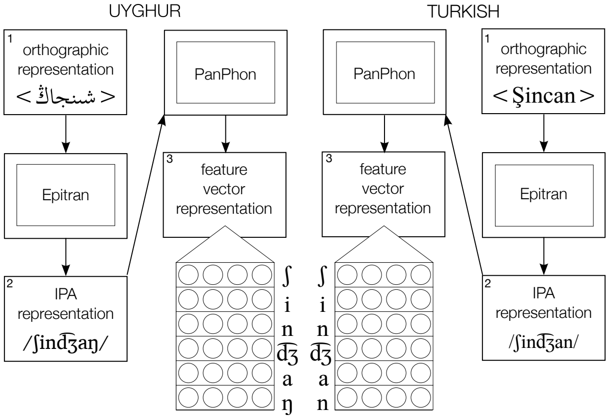

Figure 3: Use of Epitran and PanPhon to enable transfer across orthographies

in Uyghur (Perso-Arabic script) and Turkish (Latin script), thus making the equivalence across scripts apparent. We concatenate the feature vectors from PanPhon and 1-hot encodings of the corresponding IPA characters and use these as inputs to the character bi-LSTMs.

This shift to IPA space is motivated by prior work (Tsvetkov et al., 2015; Tsvetkov and Dyer, 2015) which demonstrated the value of projecting orthographic surface forms of words into a phonological space for detecting loan words, transliteration and cognates even in language pairs that exhibit significant typological, morphological and phonological differences. Our underlying assumption is that named entities are likely to be transliterated or retain pronunciation patterns across languages. Additionally, phenomena such as vowel harmony manifest explicitly in IPA representation and can potentially be helpful for NER. Foreign named entities for example, need not obey vowel harmony rules prevalent in languages like Turkish. A powerful sequence model such as a LSTM could be tolerant to the noise created by lexical aberrations, lack of spelling conventions, vowel raising etc. when given a phonological representation of an NE in different languages.

Our second proposed change is to incorporate attention over the sequence of IPA segments in a word when predicting scores for its possible labels. Attention is an unsupervised alternative to convolution or feature engineering to capture helpful localized phenomena like capitalization of first letter, presence of case markers, special characters, helpful morphological suffixes etc. or the conjunction of multiple such phenomena. Such features have been the mainstay of most prior work for NER. Most of these features are subtle and occur at the type level, whereas the CRF performs inference at the token level. We show (empirically) that attention makes the CRF more sensitive to such useful type level phenomena during inference and improves the statistical efficiency of the model in certain scenarios. Having described our intuitions, we now provide mathematical details of our model.

2.1 Model Description

2.1.1 LSTM

Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) belongs to a special breed of neural networks called Recurrent Neural Networks (RNNs) which are capable of processing sequential input of unbounded and arbitrary length. This makes them suitable for a sequence labeling task. LSTMs incorporate gating functions at each time step to allow the network to forget, remember and update contextual memory and mitigate problems like vanishing gradient. We use the following implementation:

$$
\begin{aligned}
i_t =& \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right)\\
c_t =& (1 - i_t) \odot c_{t-1} +\\
& i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)\\
o_t =& \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o\right)\\
h_t =& o_t \odot \tanh(c_t)
\end{aligned}
$$

where $\odot$ indicates element-wise product and σ indicates element-wise sigmoid function.

In practice we use bi-directional LSTMs (each with its own parameters) to mitigate cases where the LSTM may be biased towards the last few inputs it reads. This is done both at the word-level and the character level. Let the hidden state at time step **t** of the forward LSTM be denoted by $\overrightarrow{h_t}$ and the corresponding state of the backward LSTM be denoted by $\overleftarrow{h_t}$. At the character level, for a word with L characters, we only take the final hidden states in each direction i.e. $[\overrightarrow{h_L}; \overleftarrow{h_0}]$ and concatenate them to get a representation of the word that comprises these characters. At the word level, we concatenate corresponding forward and backward LSTM states for each word X_t to get $[\overrightarrow{h_t}; \overleftarrow{h_t}]$ which is the token level representation for the t^{th} word in a sentence X. We use this to generate un-normalized energy/score functions for the CRF layer.

2.2 Attention

Let w_t = $[\overrightarrow{h_t}; \overleftarrow{h_t}]$ indicate the concatenated word bi-LSTM output (of dimension d_1) at step t corresponding to the t^{th} word (X_t) in the input sequence X. Let M_t be the matrix containing the concatenated bi-directional character LSTM outputs for each character of X_t. It has dimensions (l_t, d_2) where d_2 is the dimension of the concatenated bi-directional character LSTM hidden states and l_t is the number of characters in X_t. Let m_t^i denote the i^{th} row of M_t. Let P be a parameter matrix of dimension (d_1, d_2) and p be a bias vector of length d_2. We follow (Bahdanau et al., 2014) in the formulation of attention context vector a_t:

$$\begin{aligned} w'_t =& \tanh(w_t \cdot P + p) \\ \alpha_i =& \frac{\exp(w'_t \cdot m_t^i)}{\sum_{j=1}^{l_t} \exp(w'_t \cdot m_t^j)} \\ a_t =& \sum_{i=1}^{l_t} (\alpha_i * m_t^i) \end{aligned}$$

The attentional context vector a_t is then appended to w_t to obtain concatenated vector u_t = [a_t; w_t]. We apply a linear transform U (matrix of dimension $(d_1 + d_2, k)$ where k is the number of unique NER tags). This gives us:

$$e_t = u_t \cdot U \qquad (1)$$

where e_t is a vector of un-normalized energy/score functions indicating the compatibility between word X_t and each of the *k* possible NER labels it can be given. Note that each word has a separate attention context vector obtained using the character LSTM hidden states generated by its constituent characters.

2.3 Conditional Random Field

Unlike Hidden Markov Models, CRFs do not enforce any independence assumptions among observed data and directly model the likelihood of a labeling hypothesis discriminatively. They also model adjacency compatibility between NER tags which can capture strong constraints like an 'I-label' tag not following an O tag without a 'B-label' tag in between them (see section 2.6). In our model, the CRF is parametrized as follows:

For a word sequence X = $(x_1, x_2, ...x_N)$, let **E** be a matrix of dimension (k,N) where k is the number of unique NER tags and N is the sequence length. The t^{th} column is the vector e_t obtained in equation 1. Let **T** be the square transition matrix of size (k+2, k+2) which captures transition score between the k NER tags, the start and the end symbols. Let Y = $(y_1, y_2, ...y_N)$ be the label sequence associated with the input word sequence, each y_i being an index into the ordered set of unique NER tags. Let y_0 be the start symbol and y_{N+1} be the end symbol. The score of this sequence is evaluated as:

$$S(X, Y) = \sum_{i=1}^{N} E_{y_i,i} + \sum_{j=0}^{N} T_{y_j,y_{j+1}}$$

Let $\mathcal{Y_X}$ indicate the exponential space of all possible labelings of this sequence X. The partition function associated with this CRF is then evaluated as:

$$Z = \sum_{Y \in \mathcal{Y_X}} e^{S(X,Y)}$$

The probability of a specific labeling $Y \in \mathcal{Y_X}$:

$$P(Y|X) = \frac{e^{S(X,Y)}}{Z}$$

The training objective is to maximize conditional log probability of the correct labeling sequence Y^*:

$$log(P(Y^*|X)) = S(X, Y^*) - log\,(Z) \qquad (2)$$

The decoding criteria for an input sequence X is:

$$Y^* = \underset{Y \in \mathcal{Y_X}}{\arg\max}\, S(X, Y) \qquad (3)$$

Normally, evaluating the partition function over the exponential space of all possible labelings would be intractable. However, as described in (Lafferty et al., 2001), this can be done efficiently for linear chain CRFs using the forward backward algorithm.

2.4 Word Representations

The inputs to our model are in the form of type level word representations (figure 2). Motivated by the distributional hypothesis (Harris, 1954; Firth, 1957) we use word embeddings as inputs. In the monolingual scenario, we use structured skipgram word embeddings (Ling et al., 2015a). For the transfer scenario, embeddings can optionally be trained using techniques like multi CCA described in (Ammar et al., 2016). By learning a linear transformation from a shared vector space between languages, the model may acquire some transfer capability to the target language.

We use character bi-LSTMs to handle the Out Of Vocabulary (OOV) problem as in (Ling et al., 2015b). However, just as a distributional hypothesis exists for words, prior work (Tsvetkov and Dyer, 2015; Tsvetkov et al., 2015) suggests phonological character representations capture inherent similarities between characters that are not apparent from orthogonal one-hot orthographic character representations and can serve as a language universal surrogate for character representations. This is also useful for multi-lingual named entity recognition as explained in section 2. Therefore we make use of Epitran and PanPhon as in figure 3 to obtain both 1-hot IPA character encodings and phonological feature vectors capturing similarity between IPA characters. These form the inputs to the character bi-LSTMs. The mapping from orthographic character segments to IPA is sometimes many to one (ambiguous) and unarticulated characters (like periods in NE abbreviations) and capitalization information is lost by default. Given the importance of such orthographic features for NER and the ambiguity introduced, a drop in performance may be expected by using phonological character representations.

2.5 Training

Our model parametrization is similar to (Lample et al., 2016). The weights to be trained include the the CRF transition matrix **T**, the projection parameters are used to generate matrix **E** (P and U), the LSTM parameters and word and character embedding matrices. The objective is to maximize the log probability of the correct labeling sequence as given in equation 2. This objective is fully differentiable and standard back-propagation is used to learn weights. We use Stochastic Gradient Descent with a learning rate of 0.05 and gradients clipped at 5.0 providing best performance. Using Adadelta or Adam leads to faster convergence but slightly worse performance. Word level LSTMs use a hidden layer size of 100, orthographic character LSTMs (if used) used a hidden layer of size 25 while phonological character LSTMs used a hidden layer of size 15 due to the smaller phonetic alphabet. Tuning these did not have a major effect on performance. Dropout of 0.5 is applied after concatenation of the word embeddings and character LSTM outputs. Best dropout value was chosen through ablation studies. For Spanish, we use word embeddings pre-trained on the Spanish Gigaword version 3. For transfer experiments, we use multilingual word embeddings trained using multi CCA described in (Ammar et al., 2016).

2.6 Entity Types and Tagging Schemes

In all of the datasets in table 1, 4 entity types are annotated:

1. **Persons (PER)**
   Real/fictional, living/dead people. Aliases and family names are also annotated. E.g. Barack Obama, the [Kennedys], Puff Daddy etc.
2. **Locations (LOC)**
   Geographical locations without a dedicated population and government. E.g. Nile river, Sahara desert, Mt. Everest, Asia etc.
3. **Geo-Political Entities (GPE)**
   Geographical regions with corresponding population and government. Mentions can be nominal (e.g. India, E.U., Britain etc.) or adjectival (e.g. [British] army, [French] government etc.).
4. **Organizations (ORG)**
   Names of entities with organization and managerial structure. E.g. Democratic Party, Google, JetBlue, etc.

A BIO tagging scheme is used for all annotations. All non-entity tokens are annotated as 'O'. The first token of an entity span is annotated as 'B-label' and all remaining tokens in the entity span are annotated as 'I-label'.

3 Experiments

We conduct four different experiments:

1. CoNLL 2002 Spanish NER (Sang., 2002) task. This demonstrates the monolingual competence of phonological character representations vs. orthographic representations.

2. Turkish NER using the Linguistic Data Consortium's LDC2014E115 BOLT Turkish Language Pack [2]. This demonstrates the utility of phonological character representations and attention in a morphologically rich, low resource language. We compare against a state-of-the-art monolingual model (Lample et al., 2016) that uses orthographic character LSTMs.
3. Transfer Experiments from Uzbek to Turkish using LDC2014E112 BOLT [3] data pack for Uzbek and LDC2014E115 BOLT data pack for Turkish. These two languages have similar syntax and word order (Dryer, 2013) but vary significantly in morphology, vowel harmony and phonology, can use different scripts (Uzbek-Latin/Cyrilic, Turkish-Latin) and are not mutually intelligible.
4. Transfer Experiments from Uzbek and Turkish into Uyghur using LDC2014E112 and LDC2014E115 BOLT data pack for Uzbek and Turkish respectively and Uyghur data provided as part of DARPA LORELEI[4]. These languages all have different scripts, lexicons, morphology and phonology. Results are reported by NIST [5] on an unseen test set.

3.1 Results

Tables 2 and 3 report results from monolingual experiments in Spanish. In table 3, we report the performance of our best model against other state-of-the-art models for the Spanish CoNLL 2002 NER task (Sang., 2002). Our model performs marginally better than other benchmarks with the optimal configuration of hyper-parameters and using pre-trained word embeddings. Table 2 report ablation study results, which reveal that using pre-trained word embeddings without using character LSTMs yields a very strong baseline that already out-performs various previous benchmarks. Using character LSTMs that compose orthographic character representations yields a +0.91 improvement in F1 score and a further

Language	# Sentences	# Entities
Spanish	8323	18798
Turkish	5065	4883
Uzbek	10078	7960
Uyghur	2161	2668*

Table 1: Dataset Statistics. * indicates non-gold annotations produced by a non-speaker linguist.

Phono Chars	Ortho Chars	Word Vecs	Cap+ Cat[6]	Ortho Attn	Phono Attn	F1
No	No	Yes	No	No	No	83.61
No	Yes	Yes	No	No	No	84.52
No	Yes	Yes	No	Yes	No	84.64
No	Yes	Yes	Yes	No	No	84.91
No	Yes	Yes	Yes	Yes	No	85.25
Yes	No	Yes	No	No	No	84.08
Yes	No	Yes	No	No	Yes	84.88
Yes	No	Yes	Yes	No	No	84.89
Yes	No	Yes	Yes	No	Yes	**85.81**
Yes	Yes	Yes	No	No	Yes	84.53
Yes	Yes	Yes	Yes	No	No	84.92
Yes	Yes	Yes	Yes	Yes	Yes	84.75
Yes	Yes	Yes	Yes	No	Yes	84.84
Yes	Yes	Yes	Yes	Yes	No	85.32

Table 2: Ablation Tests on Spanish CoNLL 2002. **Bold** indicates the best model.

Model	F1
Carreras et al. (2002)*	81.39
dos Santos et al. (2015)	82.21
Gillick et al. (2015)	81.83
Gillick et al. (2015)*	82.95
Lample et al. (2016)	85.75
Yang et al. (2016)	85.77
Our Best	**85.81**

Table 3: Comparison with benchmarks. * indicates a model that uses external labeled data

Phono Chars	Ortho Chars	Word vecs	Cap+ Cat	Ortho Attn	Phono Attn	F1
No	No	Yes	No	No	No	49.2
No	Yes	Yes	No	No	No	65.41
No	Yes	Yes	No	Yes	No	64.76
No	Yes	Yes	Yes	No	No	60.57
No	Yes	Yes	Yes	Yes	No	60.87
Yes	No	Yes	No	No	No	63.04
Yes	No	Yes	No	No	Yes	**66.07**
Yes	No	Yes	Yes	No	No	59.08
Yes	No	Yes	Yes	No	Yes	62.46
Yes	Yes	Yes	No	No	Yes	63.43
Yes	Yes	Yes	Yes	No	No	63.46
Yes	Yes	Yes	Yes	Yes	Yes	**66.47**

Table 4: Ablation Tests on Turkish **Bold** indicates the best transfer eligible (66.07) and transfer ineligible models (66.47)

[2] http://opencatalog.darpa.mil/BOLT.html

[3] BOLT contains newswire, discussion forum, social media and chat data

[4] http://www.darpa.mil/program/low-resource-languages-for-emergent-incidents

[5] https://www.nist.gov

[6] Sparse features for character capitalization and character type (digit, punctuation etc.)

Model	F1
Lample et al. (2016)	61.11
Lample et al. (2016) with pretrained embeddings	65.41
Our Best model	**66.47**
Our Best transfer-eligible model	**66.07**

Table 5: Comparison with state-of-the-art monolingual Turkish model

Model	F1
Lample et al. (2016)	37.1
Our best transfer model*	**51.2**

Table 6: NIST evaluations for Uyghur. * indicates transfer from Uzbek and Turkish

+0.12 F1 with attention. Using phonological character representations instead yields an improvement of +0.47 F1 and a further +0.8 F1 with attention. Thus, phonological representations benefit more from attention applied over them to beat out orthographic representations in that scenario. Using sparse features indicating the character category (alphabet vs. number vs. punctuation/non-phonetic) and capitalization in conjunction with with phonological character representations and word embeddings with attention over phonological characters yields the best configuration that slightly outperforms the best published models so far. Given that many previous benchmarks used features that rely heavily on orthography, this is an encouraging result since one would expect to lose some performance by using more abstract phonological representations as explained in section 2.4.

Tables 4 and 5 highlight results from monolingual experiments on Turkish. This dataset is much more challenging since the annotated training courpus is significantly smaller than the CoNLL 2002 shared task dataset and because Turkish is an agglutinative language exhibiting sparsity inducing morphology which leads to huge vocabulary size. As a competitive baseline, we train the LSTM CRF described in (Lample et al., 2016) due to its documented ability to obtain state-of-the-art monolingual results for many languages with minimal feature engineering. Our best model from the Turkish ablation study outperforms this baseline. We also see a stark contrast between the ablation study results for Turkish compared to Spanish. Firstly, word embeddings alone perform rather poorly due to the challenges of reliably estimating them for a large vocabulary given a small dataset. Character composed representations of words provide a significant performance boost (+17.27 F1 for the best model). Secondly, usage of sparse character features (like capitalization) seems to hurt performance in all but the last model in table 4. Thirdly, phonological and orthographic character representations are complementary in the case of Turkish, unlike Spanish. This is not too surprising since Turkish exhibits phonological phenomena like vowel harmony. Lack of vowel harmony in a word could give-away a foreign word or a named entity for example. Results show that attention is helpful as well. We would also like to point out that the only models in the ablation studies eligible for transfer are those that do not use orthographic character representations. Among these, the model that uses phonological representation with attention and word vectors performs the best and also outperforms the baseline system.

Our next experiments on model transfer are arguably the most interesting. Tables 7 and 8 document single source (Uzbek to Turkish) transfer results. We find that using sparse character category and capitalization features in conjunction with attention and phonological features yields the best 0-shot transfer performance (no training labels in the target language). Specifically, attention provides +6 F1 in 0-shot and 5% labeled-target language data scenarios. It is interesting to note that using multilingual word embeddings for the source and target languages alone accounts for very poor transfer. We also find that with as little as 20% of the data, we approach the performance of a monolingual target model that was trained on all the data. We also notice that the transfer models retain a consistent advantage over a monolingually trained target language model across all data availability scenarios. Lastly, we note that while attention provides a significant improvement in 0-shot and 5% data availability scenarios, a model without attention (or sparse features like capitalization) eventually does better with more data. This indicates that the model is competent enough to leverage transfer via phonology alone. This could also possibly be because attention patterns from Uzbek could bring in a bias that is eventually sub-optimal for Turkish due to dif-

Phono Chars	Word vecs	Cap+ Cat	Phono Attn	Uzbek Source F1	Target 0-shot F1	5% data	20% data	40% data	60% data	80% data	All data
No	Yes	No	No	41.87	2.09	23.44	35	42.75	46.32	48.81	50.34
Yes	Yes	No	Yes	61.24	11.9	34.06	47.84	56.1	53.5	64.72	65.2
Yes	Yes	No	No	60.92	15.55	39.42	**60.14**	**63.23**	**62.54**	**65.24**	**65.63**
Yes	Yes	Yes	No	64.89	22.14	41.19	54.02	57.06	59.26	60.84	61.72
Yes	Yes	Yes	Yes	61.85	**26.92**	**47.21**	58.58	60.32	60.7	62.84	63.58

Table 7: Model Transfer from Uzbek (Source) to Turkish (Target) at different target data availability thresholds

Model	0-shot	5% data	20% data	40% data	60% data	80% data	All data
LSTM-CRF (Lample et al., 2016)	0	33.44	50.61	53.25	57.41	60	61.11
S-LSTM (Lample et al., 2016)	0	15.41	39.33	42.99	51.92	51.55	56.58

Table 8: Monolingual Turkish baseline at different data availability thresholds

ferent morphology and phonology. In future work, we shall perform more insightful error analysis to explain these trends.

Table 6 documents NIST evaluation results on an unseen Uyghur test set (with gold annotations) for the best transfer model configuration jointly trained on Turkish and Uzbek gold annotations and Uyghur training annotations produced by a non-speaker linguist (non-gold). Since Uyghur lacks helpful type-level orthographic features such as capitalization, our transfer model in table 6 does not use any sparse features or attention but benefits from transfer via the phonological character representations we've proposed. Despite the noisy supervision provided in the target language, transferring from Turkish and Uzbek provides a +14.1 F1 improvement over a state of the art monolingual model trained on the same Uyghur annotations. It is worth pointing out that this transfer was achieved across 3 languages each with different scripts, morphology, phonology and lexicons.

4 Prior Work

NER is a well-studied sequence-labeling problem for which many different approaches have been proposed. Early works had a monolingual focus and relied heavily on feature engineering. Approaches include maximum entropy models (Chieu and Ng, 2003), hierarchically smoothed tries (Cucerzan and Yarowsky, 1999), decision trees (Carreras et al., 2002) and models incorporating syntactic, semantic and world knowledge (Wakao et al., 1996). Each of these models brings in a bias of its own. Florian et al. (2003) successfully tried ensembling multiple classifiers and improved performance.

Since NER is a sequence labeling problem, there are local dependencies both among NE labels associated with words and among the words themselves, that could aid the labeling process. To explicitly deal with these sequential characteristics, Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs) became very popular. (Klein et al., 2003; Florian et al., 2003; McCallum and Li, 2003; Ratinov and Roth, 2009; Chandra et al., 1981; Lin and Wu, 2009; Lample et al., 2016; Yang et al., 2016; Ma and Hovy, 2016). CRFs eventually became more popular because they are discriminative models that directly model the required posterior probability of a labeling sequence using parametrized functions of features. They do not model the probability of the observed sentence itself, avoid Markovian independence assumptions made by HMMs and avoid the label bias problem.

Most of the work cited so far makes use of hand engineered features. The following approaches minimize the use of features while still maintaining a monolingual focus. Collobert et al. (2011), Turian et al. (2010), and Ando and Zhang (2005) use unsupervised features in conjunction with engineered features capturing capitalization, character categories and gazetteer matches. Collobert et al. (2011) use a Convolutional Neural Network (CNN) over the sequence of word embeddings. Huang et al. (2015) instead use bi-directional LSTMs over the sequence of words, along with spelling and orthographic features.

The most recent work eliminates feature engineering altogether and combines CRFs with LSTMs which can model long sequences while remembering appropriate past context. Lample et al. (2016) proposed an architecture that uses both character and word level LSTMs to produce score functions for CRF inference conditioned on global context. Ma and Hovy (2016) replace the character LSTMs of Lample et al. (2016) with a CNN instead. Yang et al. (2016) follow a very similar architecture to Lample et al. (2016), replacing the LSTMs with Gated Recurrent Units (Cho et al., 2014). However, Yang et al. (2016) also tackle multi task and multi-lingual joint training scenarios.

Most of the models cited so far are monolingual either because they use hand crafted features and language specific resources or because of deep-seated assumptions. For example a change in orthography, lexicon, script, word order or addition of complex morphology makes transfer impossible. This is the central challenge that we tackle. There has been much less work catering to this scenario. Kim et al. (2012) use weak annotations from Wikipedia metadata and parallel data for multi lingual NER. Yang et al. (2016) addresses the use case of multilingual joint training, which assumes there is sufficient data available in all languages. Nothman et al. (2013) also operate under the assumption of availability of Wikipedia data.

To the best of our knowledge, a scenario involving transfer of a model trained in one (or more) source language(s) to another language with little or no labeled data, different script, different morphology, different lexicon, lack of transliteration, non-mutual intelligibility etc. has not been addressed recently.

5 Conclusion

In this paper, we presented two improvements over a state-of-the-art monolingual model to address Named Entity Recognition in transfer settings. The first seeks to reconcile various dimensions of variability between languages such as varying script, orthographic conventions, phonological phenomena etc. by representing words as sequences of IPA (International Phonetic Alphabet) segments consistent across all languages, rather than sequences of characters specific to a particular language. Secondly, we exploit the one-to-one mapping between input sequence words and output labels and advocate for the use of attention over the character/IPA sequence of a word when predicting its label. We show empirically that these two improvements 1) achieve at least state-of-the-art performance on a monolingual NER task in Spanish, 2) handle complex morphology in languages such as Turkish, Uzbek and Uyghur better than state of the art, 3) provide 0-shot performance in a transfer scenario to a related new language, well above that possible using multilingual word embeddings alone, and 4) are capable of generalizing to a new language with much less training data than a monolingually trained model. Moreover, all of this is achieved without any extra feature engineering specific to the task or language, apart from mapping characters in that language to IPA. We believe these results to be encouraging and look forward to replicating these results on more language pairs in different language families and further automating the process of obtaining phonological character representations.

6 Acknowledgement

This work is sponsored by Defense Advanced Research Projects Agency Information Innovation Office (I2O). Program: Low Resource Languages for Emergent Incidents (LORELEI). Issued by DARPA/I2O under Contract No. HR0011-15-C-0114. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A Smith. 2016. Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925*.

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *The Journal of Machine Learning Research*, 6:1817–1853.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Xavier Carreras, Lluis Marquez, and Lluís Padró. 2002. Named entity extraction using adaboost. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–4. Association for Computational Linguistics.

Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. *Journal of the Association for Computing Machinery*, 28(1):114–133.

Hai Leong Chieu and Hwee Tou Ng. 2003. Named entity recognition with a maximum entropy approach. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 160–163. Edmonton, Canada.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Silviu Cucerzan and David Yarowsky. 1999. Language independent named entity recognition combining morphological and contextual evidence. In *Proceedings of the 1999 Joint SIGDAT Conference on EMNLP and VLC*, pages 90–99.

Cıcero dos Santos, Victor Guimaraes, RJ Niterói, and Rio de Janeiro. 2015. Boosting named entity recognition with neural character embeddings. In *Proceedings of NEWS 2015 The Fifth Named Entities Workshop*, page 25.

Matthew S. Dryer, 2013. *Order of Subject, Object and Verb*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

John Rupert Firth. 1957. A synopsis of linguistic theory. In *In Studies in Linguistic Analysis*. Oxford: Philological Societ.

Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. 2003. Named entity recognition through classifier combination. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 168–171. Edmonton, Canada.

Dan Gillick, Cliff Brunk, Oriol Vinyals, and Amarnag Subramanya. 2015. Multilingual language processing from bytes. *arXiv preprint arXiv:1512.00103*.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

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

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Sungchul Kim, Kristina Toutanova, and Hwanjo Yu. 2012. Multilingual named entity recognition using parallel data and metadata from wikipedia. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 694–702. Association for Computational Linguistics.

Dan Klein, Joseph Smarr, Huy Nguyen, and Christopher D. Manning. 2003. Named entity recognition with character-level models. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 180–183. Edmonton, Canada.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Dekang Lin and Xiaoyun Wu. 2009. Phrase clustering for discriminative learning. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1030–1038. Association for Computational Linguistics.

Wang Ling, Chris Dyer, Alan Black, and Isabel Trancoso. 2015a. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Wang Ling, Tiago Luís, Luís Marujo, Ramón Fernandez Astudillo, Silvio Amir, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015b. Finding function in form: Compositional character models for open vocabulary word representation. *arXiv preprint arXiv:1508.02096*.

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 (LREC16)*.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. 2013. Named entity recognition: fallacies, challenges and opportunities. *Computer Standards & Interfaces*, 35(5):482–489.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 188–191. Association for Computational Linguistics.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R Curran. 2013. Learning multilingual named entity recognition from wikipedia. *Artificial Intelligence*, 194:151–175.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155. Association for Computational Linguistics.

Erik F. Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: Languageindependent named entity recognition. *In Proc. CoNLL*.

Yulia Tsvetkov and Chris Dyer. 2015. Cross-lingual bridges with models of lexical borrowing. *JAIR*.

Yulia Tsvetkov, Waleed Ammar, and Chris Dyer. 2015. Constraint-based models of lexical borrowing. *NAACL*.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics.

Takahiro Wakao, Robert Gaizauskas, and Yorick Wilks. 1996. Evaluation of an algorithm for the recognition and classification of proper names. In *Proceedings of the 16th conference on Computational linguistics-Volume 1*, pages 418–423. Association for Computational Linguistics.

Zhilin Yang, Ruslan Salakhutdinov, and William Cohen. 2016. Multi-task cross-lingual sequence tagging from scratch. *arXiv preprint arXiv:1603.06270*.