Siamese convolutional networks based on phonetic features for cognate identification

Taraka Rama
Seminar für Sprachwissenschaft
University of Tübingen
`taraka-rama.kasicheyanula@uni-tuebingen.de`

Abstract

In this paper, we explore the use of convolutional networks (ConvNets) for the purpose of cognate identification. We compare our architecture with binary classifiers based on string similarity measures on different language families. Our experiments show that convolutional networks achieve competitive results across concepts and across language families at the task of cognate identification.

1 Introduction

Cognates are words that are known to have descended from a common ancestral language. In historical linguistics, identification of cognates is an important step for positing relationships between languages. Historical linguists apply the comparative method (Trask, 1996) for positing relationships between languages.

In NLP, automatic identification of cognates is associated with the task of determining if two words are descended from a common ancestor or not. There are at least two ways to achieve automatic identification of cognates.

One way is to modify a well-known string alignment technique such as Longest Common Subsequence or Needleman-Wunsch algorithm (Needleman and Wunsch, 1970) to weigh the alignments differentially (Kondrak, 2001; List, 2012). The weights are determined through the linguistic knowledge of the sound changes that occurred in the language family.

The second approach employs a machine learning perspective that is widely employed in NLP. The cognate identification is achieved by training a linear classifier or a sequence labeler on a set of labeled positive and negative examples; and then employ the trained classifier to classify new word pairs. The features for a classifier consist of word similarity measures based on number of shared bigrams, edit distance, and longest common subsequence (Hauer and Kondrak, 2011; Inkpen et al., 2005).

The above procedures provide an estimate of the similarity between a pair of words and cannot directly be used to infer a phylogeny based on models of trait evolution. The pairwise judgments have to be converted into multiple cognate judgments so that the multiple judgments can be supplied to a automatic tree building program for inferring a phylogeny for the languages under study.

It has to be noted that the Indo-European dating studies (Bouckaert et al., 2012; Chang et al., 2015) employ human expert cognacy judgments for inferring phylogeny and dates of a very well-studied language family. Hence, there is a need for developing automated cognate identification methods that can be applied to under-studied languages of the world.

2 Related work

The earlier computational effort of (Jäger, 2013; Rama et al., 2013) employs Pointwise Mutual Information (PMI) to compute transition matrices between sounds. Both Jäger (2013) and Rama et al. (2013) employ undirectional sound correspondence based scorer to compute word similarity. The general approach is to align word pairs using vanilla edit distance and impose a cutoff to extract potential cognate pairs. The aligned sound symbols are then employed to compute the PMI scoring matrix that is used to realign the pairs. The PMI scoring matrix is recounted from the realigned pairs. This procedure is repeated until convergence.

Jäger (2013) imposes an additional cutoff based on the PMI scoring matrix. Further, Jäger (2013)

also employs the PMI scoring matrix to infer family trees for new language families and compares those trees with the *expert* trees given in *Glottolog* (Nordhoff and Hammarström, 2011). Rama et al. (2013) take a slightly different approach, in that, the authors compute a PMI matrix independently for each language family and evaluate its performance at the task of pair-wise cognate identification. In this work, we also compare the convolutional networks against PMI based binary classifier.

Previous works of cognate identification such as (Bergsma and Kondrak, 2007; Inkpen et al., 2005) supply string similarity measures as features for training different classifiers such as decision trees, maximum-entropy, and SVMs for the purpose of determining if a given word pair is cognate or not.

In another line of work, List (2012) employs a transition matrix derived from historical linguistic knowledge to align and score word pairs. This approach is algorithmically similar to that of Kondrak (2000) who employs articulation motivated weights to score a sound transition matrix. The weighted sound transition matrix is used to score a word pair.

The work of List (2012) known as Sound-Class Phonetic Alignment (SCA) approach reduces the phonemes to historical linguistic motivated sound classes such that transitions between some classes are less penalized than transitions between the rest of the classes. For example, the probability of velars transitioning to palatals is a well-attested sound change across the world. The SCA approach employs a weighted directed graph to model directionality and proportionality of sound changes between sound classes. For example, a direct change between velars and dentals is unattested and would get a zero weight. Both Kondrak (2000) and List (2012) set the weights and directions in the sound transition graph to suit the reality of sound change.

All the above outlined approaches employ a scoring matrix that is derived automatically or manually; or, employ a SVM to train form similarity based features for the purpose of cognate identification.

3 Convolutional networks

This article is the first to apply convolutional networks (ConvNets) to phonemes by treating each phoneme as a vector of binary valued phonetic features. This approach has the advantage that it does not require explicit feature engineering, alignments, and a sound transition matrix. The approach requires cognacy statements and phonetic descriptions of sounds used to transcribe the words. The cognacy statements can be obtained from etymological dictionaries and the quality of the phonemes can be obtained from Ladefoged and Maddieson (1998).

Collobert et al. (2011) proposed ConvNets for NLP tasks in 2011 and were since applied for sentence classification (Kim, 2014; Johnson and Zhang, 2015; Kalchbrenner et al., 2014; Zhang et al., 2015), part-of-speech tagging (Santos and Zadrozny, 2014), and information retrieval (Shen et al., 2014).

Kim (2014) applied convolutional networks to pre-trained word embeddings in a sentence for the task of sentence classification. Johnson and Zhang (2015) train their convolutional network from scratch by using a one-hot vector for each word. The authors show that their convolutional network performs better than a SVM classifier trained on bag-of-words features. Santos and Zadrozny (2014) use character embeddings to train their POS-tagger. The authors find that the POS-tagger performs better than the accuracies reported in (Manning, 2011).

In a recent work, Zhang et al. (2015) treat documents as a sequence of characters and transform each document into a sequence of one-hot character vectors. The authors designed and trained two 9-layer convolutional networks for the purpose of sentiment classification. The authors report competitive or state-of-the art performance on a wide range of benchmark sentiment classification datasets.

4 Character convolutional networks

Chopra et al. (2005) extended the traditional ConvNets to classify if two images belong to the same person. These ConvNets are known as Siamese Networks (inspired from Siamese twins) and share weights for independent but identical layers of convolutional networks. Siamese networks and their variants have been employed for identifying if two images are from the same person or different persons (Zagoruyko and Komodakis, 2015); and for recognizing if two speech segments belong to the same word class (Kamper et al., 2015).

4.1 Word as image

Historical linguists perform cognate identification based on regular correspondences which are described as changes in phonetic features of phonemes. For instance, Grimm's law $b^h \sim b$ is described as loss of aspiration; $p \sim f$ is described as change from plosives to fricatives; and devoicing $d \sim t$ in English *ten* $\sim$ Latin *decem*.

Learning criteria for cognacy through phonetic features from a set of training examples implies that there is no need for explicit alignment and design/learning of sound scoring matrices. In this article, we represent each phoneme as a binary-valued vector of phonetic features and then perform convolution on the two-dimensional matrix.

4.2 Siamese network

Intuitively, a network should learn a similarity function such that words that diverged due to accountable sound shifts are placed close to one another than two words that are not cognates. And, Siamese networks are suitable for this task since, they learn a similarity function that has a higher similarity between cognates as compared to non-cognates. The weight tying ensures that two cognate words sharing similar phonetic features in a local context tend to be get higher weights than words that are not cognate.

4.3 Phoneme vectorization

In this article, we work with the ASJP alphabet (Brown et al., 2013). The ASJP alphabet is coarser than IPA but is designed with the aim to capture highly frequent sounds in the world's languages. The ASJP database has word lists for 60% of the world's languages but only has cognate judgments for some selected families (Wichmann and Holman, 2013).

We composed a binary vector for each phoneme based on the description given in table 1. In total, there are 16 binary valued features. We also reduced all vowels to a single vowel that has a value of 1 for voicing feature and 0 for the rest of the features. The main motivation for such decision is that vowels are diachronically unstable than consonants (Kessler, 2007).

A word such as "fat" would be represented as 3×16 matrix where each column provides a binary value for the phonetic feature (cf. table 2).

p = voiceless bilabial stop and fricative [IPA: p, ɸ]
b = voiced bilabial stop and fricative [IPA: b, β]
m = bilabial nasal [IPA: m]
f = voiceless labiodental fricative [IPA: f]
v = voiced labiodental fricative [IPA: v]
8 = voiceless and voiced dental fricative [IPA: θ, ð]
4 = dental nasal [IPA: n̪]
t = voiceless alveolar stop [IPA: t]
d = voiced alveolar stop [IPA: d]
s = voiceless alveolar fricative [IPA: s]
z = voiced alveolar fricative [IPA: z]
c = voiceless and voiced alveolar affricate [IPA: ts, dz]
n = voiceless and voiced alveolar nasal [IPA: n]
S = voiceless postalveolar fricative [IPA: ʃ]
Z = voiced postalveolar fricative [IPA: ʒ]
C = voiceless palato-alveolar affricate [IPA: tʃ]
j = voiced palato-alveolar affricate [IPA: dʒ]
T = voiceless and voiced palatal stop [IPA: c, ɟ]
5 = palatal nasal [IPA: ɲ]
k = voiceless velar stop [IPA: k]
g = voiced velar stop [IPA: g]
x = voiceless and voiced velar fricative [IPA: x, ɣ]
N = velar nasal [IPA: ŋ]
q = voiceless uvular stop [IPA: q]
G = voiced uvular stop [IPA: ɢ]
X = voiceless and voiced uvular fricative, voiceless and voiced pharyngeal fricative [IPA: χ, ʁ, ħ, ʕ]
7 = voiceless glottal stop [IPA: ʔ]
h = voiceless and voiced glottal fricative [IPA: h, ɦ]
l = voiced alveolar lateral approximate [IPA: l]
L = all other laterals [IPA: ʟ, ʎ]
w = voiced bilabial-velar approximant [IPA: w]
y = palatal approximant [IPA: j]
r = voiced apico-alveolar trill and all varieties of "r-sounds" [IPA: r, R, etc.]
! = all varieties of "click-sounds" [IPA: !, ʘ, ‖, ǂ]

Table 1: ASJP consonants. ASJP has 6 vowels which we collapsed to a single vowel *V*.

4.4 ConvNet Models

In this subsection, we describe the ConvNet models used in our experiments.

Siamese ConvNet Siamese networks takes a pair of inputs and minimizes the distance between the output representations. Each branch of the Siamese network is composed of a convolutional network. The Euclidean distance D between the representations of each branch is then used to train a contrastive-loss function $yD + (1 - y)max\{0, m - D\}$ where m is the margin and y is the true label. We only describe the architecture since this forms the basis for the rest of our experiments with Siamese architectures.[1]

Manhattan Siamese ConvNet The second ConvNet is also a Siamese network where the Euclidean distance is replaced by a element-wise absolute difference layer followed by a fully con-

---

[1] The results were slightly better than a majority class classifier and were not reported in the article

p	0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0
b	1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0
f	0 1 1 0 0 0 0 0 0 1 0 0 0 0 0 0
v	1 1 1 0 0 0 0 0 0 1 0 0 0 0 0 0
m	1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0
8	1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0
4	1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0
t	0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0
d	1 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0
s	0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0
z	1 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0
c	1 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0
n	1 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0
S	0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0
Z	1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0
C	0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0
j	1 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0
T	1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0
5	0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0
k	0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0
g	1 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0
x	1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0
N	1 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0
q	0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0
G	1 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0
X	1 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0
7	0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
h	1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0
l	1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0
L	1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
w	1 1 0 0 0 1 0 0 0 0 0 0 0 1 0 0
y	1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0
r	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
!	1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
V	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Table 2: Binarized ASJP alphabet used in our experiments. Each column corresponds to the following features: Voiced, Labial, Dental, Alveolar, Palatal/Post-alveolar, Velar, Uvular, Glottal, Stop, Fricative, Affricate, Nasal, Click, Approximant, Lateral, and Rhotic.

nected layer (cf. figure 1). To the best of our knowledge, only Zagoruyko and Komodakis (2015) added two fully connected layers to the concatenated outputs of the Siamese network and trained a system that predicts if two image patches belong to the same image or different images. We refer this architecture as a Manhattan Siamese ConvNet due to the difference layer's similarity to Manhattan distance.

2-channel Convnet Until now, each word is treated as a separate input. Zagoruyko and Komodakis (2015) introduced a 2-channel architecture which treats a pair of image patches as a 2-channel image. This can also be applied to words. The 2-channel ConvNet has two convolutional layers, a maxpooling layer, and a fully connected layer with 8 units.
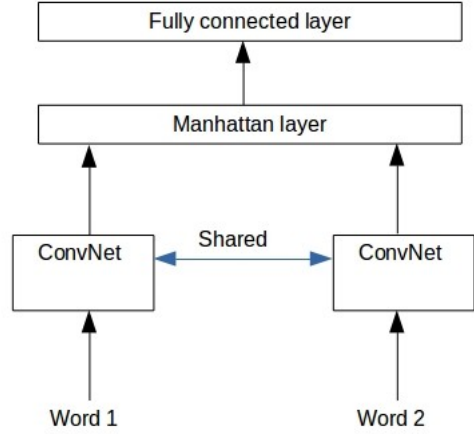

Figure 1: Siamese network with fully connected layer. The weighted are shared between the two convolutional networks.

The number of feature maps in each convolutional layer is fixed at 10 with a kernel size of 2×3. The max-pooling layer halves the output of the previous convolutional layer. We also inserted a dropout layer with 0.5 probability (Srivastava et al., 2014) after a fully-connected layer to avoid over-fitting. The convolutional layers were trained with ReLU non-linearity.

We zero-padded each word to obtain a length of 10 for all the words to apply the filter equally about a word. We used adadelta optimizer (Zeiler, 2012) with learning rate of 1.0, $\rho = 0.95$, and $\epsilon = 10^{-6}$. We fixed the mini-batch size to 128 in all our experiments. We experimented with different batch sizes ($[32, 64, 128, 256]$) and did not observe any significant deviation in the validation loss. Both, Manhattan and 2-stream ConvNets were trained using the log-loss function. Both our architectures are relatively shallow (3) as compared to the text classification architecture of Zhang et al. (2015). We trained all our networks using Keras (Chollet, 2015) and Theano (Bergstra et al., 2010).

5 Comparison methods

We compare the ConvNet architectures with SVM classifiers trained with different string similarities as features.

Other sound classes/alphabets Apart from ASJP alphabet, there are two other alphabets that have been designed by historical linguists for the purpose of modeling sound change. As mentioned before, the main idea behind the design of sound classes is to discourage transitions between partic-

ular classes of sounds but allow transitions within a sound class. Dolgopolsky (1986) proposed a ten sound class system based on the empirical data of 140 languages. SCA alphabet (List, 2012) has a size of 25 and attempts to address some issues with the ASJP alphabet (lack of tones) and also extend Dolgopolsky's sound classes based on evidence from more number of languages.

Orthographic measures as features We converted all the datasets into all the three sound classes and computed the following string similarity scores:

- Edit distance.
- Common number of bigrams.
- Length of the longest common subsequence.
- Length of longest common prefix.
- Common number of trigrams.
- Global alignment based on Needlman-Wunch algorithm (Needleman and Wunsch, 1970).
- Local alignment score based on Smith-Waterman algorithm (Smith and Waterman, 1981).
- Semi-global alignment score is a compromise between global and local alignments (Durbin et al., 2002).[2]
- Common number of skipped bigrams (XDICE).
- A positional extension of XDICE known as XXDICE (Brew and McKelvie, 1996).

Pair-wise Mutual Information (PMI) We also computed a PMI score for a pair of ASJP transcribed words using the PMI scoring matrix developed by Jäger (2013). This system is referred to as PMI system.

We included length of each word and the absolute difference in length between the words as features for both the Orthographic and PMI systems. The sound class orthographic scores system attempts to combine the previous cognate identification systems developed by (Inkpen et al., 2005; Hauer and Kondrak, 2011) and the insights from applying string similarities to sound classes for language comparison (Kessler, 2007).

6 Datasets

In this section, we describe the datasets used in our experiments.

IELex database The Indo-European Lexical database is created by Dyen et al. (1992) and curated by Michael Dunn.[3] The transcription in IELex database is not uniformly IPA and retains many forms transcribed in the Romanized IPA format of Dyen et al. (1992). We cleaned the IELex database of any non-IPA-like transcriptions and converted part of the database into ASJP format.

Austronesian vocabulary database The Austronesian Vocabulary Database (Greenhill and Gray, 2009) has word lists for 210 Swadesh concepts and 378 languages.[4] The database does not have transcriptions in a uniform IPA format. We removed all symbols that do not appear in the standard IPA and converted the lexical items to ASJP format.[5]

Family	Concepts	Languages	Training	Testing
Austronesian	210	100	334807	140697
Mayan	100	30	28222	12344
Indo-European	206	50	117740	49205
Mixed dataset	–	–	176889	–

Table 3: The number of languages, concepts, training, and test examples in our datasets. We do not test on the mixed database and only use it for training purpose.

Short word lists with cognacy judgments Wichmann and Holman (2013) and List (2014) compiled cognacy wordlists for subsets of families from various scholarly sources such as comparative handbooks and historical linguistics' articles. The details of this compilation is given below. For each dataset, we give the number of languages/the number of concepts in parantheses. This dataset is henceforth referred to as "Mixed dataset".

- Wichmann and Holman (2013): Afrasian (21/40), Kadai (12/40), Kamasau (8/36), Lolo-Burmese (15/40), Mayan (30/100), Miao-Yao (6/36), Mixe-Zoque (10/100), Mon-Khmer (16/100).

- List (2014): Bai dialects (9/110), Chinese dialects (18/180), Huon (14/84), Japanese (10/200), ObUgrian (21/110; Hungarian excluded from Ugric sub-family), Tujia (5/107; Sino-Tibetan).

---

[2]The global, local, and alignment scores were computed using LingPy library (List and Moran, 2013).

[3]`ielex.mpi.nl`

[4]`http://language.psy.auckland.ac.nz/austronesian/`

[5]For computational reasons, we work with a subset of 100 languages.

We performed two experiments with these datasets. In the first experiment, we randomly selected 70% of concepts from IELex, ABVD, and Mayan datasets for training and the rest of the 30% concepts for testing. The motivation behind this experiment is to test if ConvNets can learn phonetic feature patterns across concepts. In the second experiment, we trained on the Mixed dataset but tested on the Indo-European and Austronesian datasets. The motivation behind this experiment is to test if ConvNets can learn general patterns of sound change across language families. The number of training and testing examples in each dataset is given in table 3.

7 Results

In this section, we report the results of our cross-concepts and cross-family experiments.

SVM training and evaluation metrics We used a linear kernel and optimized the SVM hyperparameter (C) through ten-fold cross-validation and grid search on the training data. We report accuracies, class-wise F-scores (positive and negative), combined F-score, and average precision score for each system on concepts dataset in table 4. The average precision score corresponds to the area under the precision-recall curve and is an indicator of the robustness of the model to thresholds.

7.1 Cross-Concept experiments

Effect of size and width of fully connected layers We observed that both the depth and width of the fully connected layers do not affect the performance of the ConvNet models. We used a fully connected network of size 8 in all our experiments. We increased the number of neurons from 8 to 64 in multiples of two and observed that increasing the number of neurons hurts the performance of the system.

Effect of filter size Zhang and Wallace (2015) observed that the size of the filter patch can affect the performance of the system. We experimented with different filter sizes of dimensions $m \times k$ where, $m \in [1, 2]$ and $k \in [1, 3]$. We did not find any change in the performance in concepts experiments. We report the results for $m = 2, k = 3$ filter size for cross-concept experiments.

7.2 Cross-Family experiments

Effect of filter size Unlike the previous experiment, the filter size has a effect on the performance of the ConvNet system. We observed that the best results were obtained with a filter size of 1×3.

We did not include the results of the 2-channel ConvNet because of its worse performance at the task of cross-family cognate identification. The results of our experiments are given in table 5.

Dataset	Orthographic	PMI	Manhattan ConvNet
Austronesian	0.766	0.78	0.746
Indo-European	0.815	0.804	0.804
Austronesian	0.821	0.837	0.820
	0.661	0.656	0.570
	0.759	0.768	0.728
Indo-European	0.876	0.873	0.871
	0.631	0.569	0.590
	0.806	0.786	0.791
Austronesian	0.771	0.795	0.707
Indo-European	0.731	0.692	0.691

Table 5: Testing accuracies, class-wise and combined F-scores, average precision score of each system on Indo-European and Austronesian families.

8 Discussion

The Manhattan ConvNet competes with PMI and orthographic models at cross-concept cognate identification task. The Manhattan ConvNet performs better than PMI and orthographic models in terms of overall accuracy in all the three language families. In terms of averaged F-scores, Manhattan ConvNet performs slightly better than orthographic model and only performs worse than the other models at Austronesian language family.

The Manhattan ConvNet shows mixed performance at the task of cross-family cognate identification. The Manhattan ConvNet does not turn up as the best system across all the evaluation metrics in a single language family. The ConvNet performs better than PMI but is not as good as Orthographic measures at Indo-European language family. In terms of accuracies, the ConvNet comes closer to PMI than the orthographic system.

These experiments suggest that ConvNets can compete with a classifier trained on different orthographic measures and different sound classes. ConvNets can also compete with a data driven method like PMI which was trained in an EM-like fashion on millions of word pairs. ConvNets can certainly perform better than a classifier trained

Language family	Orthographic	PMI	Manhattan ConvNet	2-Channel ConvNet
Austronesian	77.92%	78%	79.04%	76.1%
Indo-European	80%	78.58%	83.43%	81.7%
Mayan	83.66%	85.25%	87.1%	82.1%
	0.833	0.836	0.861	0.830
Austronesian	0.675	0.665	0.576	0.595
	0.783	0.782	0.776	0.760
	0.863	0.854	0.894	0.883
Indo-European	0.628	0.598	0.618	0.585
	0.808	0.794	0.830	0.813
	0.866	0.885	0.888	0.865
Mayan	0.791	0.795	0.756	0.734
	0.84	0.853	0.842	0.819
Austronesian	0.749	0.74	0.683	0.643
Indo-European	0.729	0.678	0.681	0.64
Mayan	0.88	0.892	0.871	0.805

Table 4: Each system is trained on cognate and non-cognate pairs on 145 concepts in Indo-European and Austronesian families; and tested on the rest of the concepts. For Mayan family, the number of training concepts is 70 and the number of concepts in testing data is 30. For each family, numbers correspond to the following metrics: Accuracies, F-scores (negative, positive, combined), Average precision score.

on word similarity scores at cross-concept experiments.

The Orthographic system and PMI system show similar performance at the Austronesian cross-concept task. However, ConvNets do not perform as well as orthographic and PMI systems. The reason for this could be due to the differential transcriptions in the database.

9 Conclusion

In this article, we explored the use of phonetic feature convolutional networks for the task of pairwise cognate identification. Our experiments with convolutional networks show that phonetic features can be directly used for classifying if two words are related or not.

In the future, we intend to work directly with speech recordings and include language relatedness information into ConvNets to improve the performance. We are currently working towards building a larger database of word lists in IPA transcription.

References

[Bergsma and Kondrak2007] Shane Bergsma and Grzegorz Kondrak. 2007. Alignment-based discriminative string similarity. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 656–663, Prague, Czech Republic, June. Association for Computational Linguistics.

[Bergstra et al.2010] James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. 2010. Theano: a cpu and gpu math expression compiler. In *Proceedings of the Python for scientific computing conference (SciPy)*, volume 4, page 3. Austin, TX.

[Bouckaert et al.2012] Remco Bouckaert, Philippe Lemey, Michael Dunn, Simon J. Greenhill, Alexander V. Alekseyenko, Alexei J. Drummond, Russell D. Gray, Marc A. Suchard, and Quentin D. Atkinson. 2012. Mapping the origins and expansion of the Indo-European language family. *Science*, 337(6097):957–960.

[Brew and McKelvie1996] Chris Brew and David McKelvie. 1996. Word-pair extraction for lexicography. In *Proceedings of the Second International Conference on New Methods in Language Processing*, pages 45–55. Ankara.

[Brown et al.2013] Cecil H. Brown, Eric W. Holman, and Søren Wichmann. 2013. Sound correspondences in the world's languages. *Language*, 89(1):4–29.

[Chang et al.2015] Will Chang, Chundra Cathcart, David Hall, and Andrew Garrett. 2015. Ancestry-constrained phylogenetic analysis supports the Indo-European steppe hypothesis. *Language*, 91(1):194–244.

[Chollet2015] Franois Chollet. 2015. Keras.

[Chopra et al.2005] Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 539–546. IEEE.

[Collobert et al.2011] 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.

[Dolgopolsky1986] Aron B. Dolgopolsky. 1986. A probabilistic hypothesis concerning the oldest relationships among the language families of northern Eurasia. In Vitalij V. Shevoroshkin and Thomas L. Markey, editors, *Typology, Relationship, and Time: A Collection of Papers on Language Change and Relationship by Soviet Linguists*, pages 27–50. Karoma, Ann Arbor, MI.

[Durbin et al.2002] Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. 2002. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge.

[Dyen et al.1992] Isidore Dyen, Joseph B. Kruskal, and Paul Black. 1992. An Indo-European classification: A lexicostatistical experiment. *Transactions of the American Philosophical Society*, 82(5):1–132.

[Greenhill and Gray2009] Simon J. Greenhill and Russell D. Gray. 2009. Austronesian language phylogenies: Myths and misconceptions about Bayesian computational methods. *Austronesian Historical Linguistics and Culture History: A Festschrift for Robert Blust*, pages 375–397.

[Hauer and Kondrak2011] Bradley Hauer and Grzegorz Kondrak. 2011. Clustering semantically equivalent words into cognate sets in multilingual lists. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 865–873, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

[Inkpen et al.2005] Diana Inkpen, Oana Frunza, and Grzegorz Kondrak. 2005. Automatic identification of cognates and false friends in French and English. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 251–257.

[Jäger2013] Gerhard Jäger. 2013. Phylogenetic inference from word lists using weighted alignment with empirically determined weights. *Language Dynamics and Change*, 3(2):245–291.

[Johnson and Zhang2015] Rie Johnson and Tong Zhang. 2015. Effective use of word order for text categorization with convolutional neural networks. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 103–112.

[Kalchbrenner et al.2014] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, June.

[Kamper et al.2015] Herman Kamper, Weiran Wang, and Karen Livescu. 2015. Deep convolutional acoustic word embeddings using word-pair side information. *CoRR*, abs/1510.01032.

[Kessler2007] Brett Kessler. 2007. Word similarity metrics and multilateral comparison. In *Proceedings of the Ninth Meeting of the ACL Special Interest Group in Computational Morphology and Phonology*, pages 6–14. Association for Computational Linguistics.

[Kim2014] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.

[Kondrak2000] Grzegorz Kondrak. 2000. A new algorithm for the alignment of phonetic sequences. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 288–295.

[Kondrak2001] Grzegorz Kondrak. 2001. Identifying cognates by phonetic and semantic similarity. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, pages 1–8. Association for Computational Linguistics.

[Ladefoged and Maddieson1998] Peter Ladefoged and Ian Maddieson. 1998. The sounds of the world's languages. *Language*, 74(2):374–376.

[List and Moran2013] Johann-Mattis List and Steven Moran. 2013. An open source toolkit for quantitative historical linguistics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 13–18, Sofia, Bulgaria, August. Association for Computational Linguistics.

[List2012] Johann-Mattis List. 2012. SCA: phonetic alignment based on sound classes. In *New Directions in Logic, Language and Computation*, pages 32–51. Springer.

[List2014] J.-M. List. 2014. *Sequence comparison in historical linguistics*. Dsseldorf University Press, Dsseldorf.

[Manning2011] Christopher D Manning. 2011. Part-of-speech tagging from 97% to 100%: is it time

for some linguistics? In *Computational Linguistics and Intelligent Text Processing*, pages 171–189. Springer.

[Needleman and Wunsch1970] Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453.

[Nordhoff and Hammarström2011] Sebastian Nordhoff and Harald Hammarström. 2011. Glottolog/Langdoc: Defining dialects, languages, and language families as collections of resources. In *Proceedings of the First International Workshop on Linked Science*, volume 783.

[Rama et al.2013] Taraka Rama, Prasant Kolachina, and Sudheer Kolachina. 2013. Two methods for automatic identification of cognates. *QITL*, 5:76.

[Santos and Zadrozny2014] Cicero D Santos and Bianca Zadrozny. 2014. Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1818–1826.

[Shen et al.2014] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 101–110. ACM.

[Smith and Waterman1981] Temple F. Smith and Michael S. Waterman. 1981. Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197.

[Srivastava et al.2014] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

[Trask1996] Robert Lawrence Trask. 1996. *Historical Linguistics*. Oxford University Press, London.

[Wichmann and Holman2013] Søren Wichmann and Eric W Holman. 2013. Languages with longer words have more lexical change. In *Approaches to Measuring Linguistic Differences*, pages 249–281. Mouton de Gruyter.

[Zagoruyko and Komodakis2015] Sergey Zagoruyko and Nikos Komodakis. 2015. Learning to compare image patches via convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June.

[Zeiler2012] Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

[Zhang and Wallace2015] Ye Zhang and Byron Wallace. 2015. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*.

[Zhang et al.2015] Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 649–657. Curran Associates, Inc.