Phonetic Vector Representations for Sound Sequence Alignment

Pavel Sofroniev and **Çağrı Çöltekin**
Department of Linguistics
University of Tübingen
pavel.sofroniev@student.uni-tuebingen.de, ccoltekin@sfs.uni-tuebingen.de

Abstract

This study explores a number of data-driven vector representations of the IPA-encoded sound segments for the purpose of sound sequence alignment. We test the alternative representations based on the alignment accuracy in the context of computational historical linguistics. We show that the data-driven methods consistently do better than linguistically-motivated articulatory-acoustic features. The similarity scores obtained using the data-driven representations in a monolingual context, however, performs worse than the state-of-the-art distance (or similarity) scoring methods proposed in earlier studies of computational historical linguistics. We also show that adapting representations to the task at hand improves the results, yielding alignment accuracy comparable to the state of the art methods.

1 Introduction

Most studies in computational linguistics or natural language processing treat the phonetic segments as categorical units, which prevents analyzing or exploiting the similarities or differences between these units. Alignment of sound sequences, a crucial step in a number of different fields of inquiry, is one of the tasks that suffers if the segments are treated as distinct symbols with no notion of similarity. As a result, alignment algorithms commonly employed in practice (e.g., Needleman and Wunsch, 1970) use a scoring function based on similarity of the individual units.

The tasks that require or benefit from aligning sequences are prevalent in computational linguistics, as well as relatively unrelated fields such as bioinformatics. In this study, we focus on aligning phonetically transcribed parallel word lists in the context of computational historical linguistics, where alignment of sound sequences is interesting either on its own (demonstrating differences between language varieties) or as a necessary step in a larger application, for example, for inferring the cognacy of these words or finding synchronic or diachronic sound correspondences.

The use of similarities between the sound segments has been common in computational studies of historical linguistics (Covington, 1996, 1998; Kondrak, 2000; Kondrak and Hirst, 2002; Kondrak, 2003; List, 2012; Jäger, 2013; Jäger and Sofroniev, 2016). These studies rely on scoring functions most of which are based on the linguistic knowledge about the sound changes that typically occur across languages. Another trend shared by all of the earlier studies is the use of a reduced alphabet for representing the sound segments. Even though the standard way to encode sound sequences is the International Phonetic Alphabet (IPA), using a smaller set of symbols, such as ASJP (Brown et al., 2013; Wichmann et al., 2016), seem to help creating scoring functions that are more useful for historical linguistics.

In the present study, we explore a number of methods that learn vector representations for IPA tokens from multi-lingual word-lists, either using the words in a monolingual context or making use of the fact that words represent the same concept in different languages. We use a standard similarity metric over vectors (cosine similarity) for determining the similarities between the segments, and, in turn, use these similarities for aligning IPA-transcribed sequences.

Besides providing a more principled method for measuring distances, compared to only distance information, vector representations are more useful for further analysis, and may yield better results in other computational tasks relying on supervised or unsupervised machine learning techniques. Vector representations for phonetic, phonological or orthographic units have been used successfully in

earlier research, e.g., for word segmentation (Ma et al., 2016), transfer learning of named entity recognition (Mortensen et al., 2016) and morphological inflection (Silfverberg et al., 2018).

We compare our methods to a one-hot-encoding baseline (which is equivalent to symbolic representations), linguistically-motivated vectors, and alignments produced using state-of-the-art scoring methods. We compare the alignment performance of these methods on a manually-annotated gold-standard corpus, using the same alignment algorithm and the same training data where applicable.

2 Methods

Our aim is to learn and use vector representations for the purposes of sound sequence alignment. Once we have vector representations, we align the two sequences with Needleman-Wunsch algorithm using the cosine similarity between the phonetic vectors as the similarity function.

2.1 Baseline Representations

One-hot encoding is a common method for representing categorical data. Under one-hot encoding, given a vocabulary of N distinct segments, each segment would be represented as a distinct binary vector of size N, such that exactly one of its dimensions has value 1 and all other dimensions have value 0. The method does not yield useful distance measures as each segment is equidistant from all the others. We use one-hot encoding as a proxy for a purely symbolic baseline.

PHOIBLE Online is an ongoing project aiming to compile a comprehensive database of the world languages' phonological inventories (Moran et al., 2014). The project also maintains a table of phonological features, effectively mapping each segment encountered in the database to a unique ternary feature vector. Feature values are assigned based on Hayes (2009) and Moisik and Esling (2011), and indicate either the presence, absence, or non-applicability of an articulatory-acoustic feature for each IPA symbol. PHOIBLE feature vectors serve as a linguistically-informed baseline.

2.2 Data-driven Vector Representations

Our proposed methods include three data-driven methods to learn vector representations for IPA-encoded sound segments.

phon2vec embeddings are the well-known word2vec method (Mikolov et al., 2013) applied to IPA-encoded phonetic segments. The method learns dense vector representations that maximize the similarity of segments that appear in similar contexts. As in original word2vec models, the context is treated as a bag of words, ignoring the relative position of each context element.

Position sensitive neural network embeddings (NN embeddings) are obtained using a simple feed-forward neural network architecture. Similar to word2vec skip-gram method, the neural network tries to predict the context of a word from the word itself. The hidden layer representations are, then, used as the representations for the word. Unlike word2vec, however, the context is not treated as a bag of phonetic segments. The position of the elements in the context is significant.

RNN embeddings are obtained using a sequence-to-sequence recurrent neural network (Cho et al., 2014). Given a pair of sequences, the network encodes the first sequence into a vector which is then decoded into an output sequence. The first layer of the network is an embeddings layer which converts the input categories to dense vector representations with a smaller number of dimensions. The network is trained to 'translate' words (as sequences of IPA tokens) between the languages in the training set, while, in the process, learning useful representations for IPA tokens. Once the network is trained, we are interested in the representations build for each IPA-token by the embedding layer.

Unlike the other data-driven methods described above, the RNN embeddings require, and make use of, multi-lingual nature of the data. However, crucially, the method does not require any explicit alignment of the sequences in advance.

2.3 State-of-the-art Scoring Functions

We compare the alignment performance of our methods to two state-of-the-art scoring functions. The first one, the sound-class-based phonetic alignment (SCA, List, 2012) employs a set of 28 sound classes. It operates on IPA sequences by converting the segments into their respective sound classes, aligning the sound class tokens, and then converting these back into IPA. The scoring function is hand-crafted to reflect the perceived probabilities of sound change transforming a segment of one class into a segment of another.

We also compare our results with the alignments obtained using the method proposed by Jäger (2013), which uses the ASJP database (Wichmann et al., 2016) to calculate the pairwise mutual information (PMI) scores for each pair of ASJP segments. The method starts with an initial alignment, and re-aligns the corpus iteratively for obtaining the final PMI-based scores. The method is data-driven, but heavily optimized for the task. Since it does not work with IPA-encoded sequences, we first convert the IPA sequences to ASJP alphabet, and convert them back to IPA after alignment.[1]

3 Experiments and Results

3.1 Data

In order to evaluate the performance of the methods put forward in the previous section, we use the Benchmark Database for Phonetic Alignments (BDPA, List and Prokić, 2014). The database contains 7198 aligned pairs of IPA sequences collected from 12 source datasets, covering languages and dialects from 6 language families (detailed information about the data set is provided in the Appendix). The database also features the small set of 82 selected pairs used by Covington (1996) to evaluate his method, encoded in IPA.

Our training data is sourced from NorthEuraLex, a comprehensive lexicostatistical database that provides IPA-encoded lexical data for languages of, primarily but not exclusively, Northern Eurasia (Dellert and Jäger, 2017). At the time of writing the database covers 1016 concepts from 107 languages, resulting in 121 614 IPA transcriptions.

3.2 Experimental Setup

Obtaining vector representations with the phon2vec and neural network methods involves settings the models' hyperparameters and training on a data set of IPA sequences (or pairs thereof).

We tokenize the input sequences using an open source Python package developed during this study.[2] The phon2vec and NN embeddings are trained on the set of all tokenised transcriptions in the training set. For training the RNN, we need cognates, pairs of words in different languages that share a common root. As our training set does not include cognacy information, the RNN embeddings are trained on the set of tokenised transcriptions of the word pairs constituting probable cognates — pairs in which the words belong to different languages, are linked to the same concept, and have normalised Levenshtein distance lower than 0.5. We have also experimented with thresholds of 0.4 and 0.6, but setting the cutoff at 0.5 yields better-performing embeddings.

For each method, we run the respective model with the Cartesian product of common values for each hyperparameter, practically performing a random search of the hyperparameter space. The values we have experimented with, as well as the best-performing combinations thereof, are summarized in the Appendix. Note that the models are optimized for the respective prediction task they perform, not for good alignment performance.

The implementation is realized in the Python programming language, and makes use of a number of libraries, including NumPy (Walt et al., 2011), SciPy (Jones et al., 2001), scikit-learn (Pedregosa et al., 2011), Gensim (Řehůřek and Sojka, 2010), and Keras (Chollet et al., 2015). The source code used for the experiments reported here is publicly available.[3]

3.3 Evaluation

In order to quantify the methods' performance, we employ an intuitive evaluation scheme similar to the one used by Kondrak and Hirst (2002): if, for a given word pair, m is the number of alternative gold-standard alignments and n is the number of correctly predicted alignments, the score for that pair would be $\frac{n}{m}$. In the common word pair case of a single gold-standard alignment and a single predicted alignment, the latter would yield 1 point if it is correct and 0 points otherwise; partially correct alignment do not yield points. The percentage scores are obtained by dividing the points by the total number of pairs.

3.4 Results and Discussion

The alignment performance of our baselines, proposed methods, as well as PMI and SCA on the BDPA data sets is summarized in Table 1.

The first point we would like to draw attention to is that the one-hot encoding scores are consistently lower than those in the other columns. This is expected because, unlike the other methods, one-

[1] The IPA to ASJP conversion is lossy. However, the alignments are not affected since the source IPA symbols are known during ASJP to IPA conversion.

[2] https://pypi.python.org/pypi/ipatok.

[3] https://github.com/pavelsof/ipavec.

	one-hot	phoible	phon2vec	nn	rnn	pmi	sca
Andean	85.66	87.31	97.25	99.34	99.50	95.21	99.67
Bai	52.55	62.77	61.25	74.72	75.52	–	83.45
Bulgarian	60.54	80.54	77.98	82.55	86.70	81.70	89.34
Dutch	14.16	25.65	26.00	32.50	32.50	36.67	42.20
French	42.94	62.92	68.94	74.30	77.04	71.98	80.90
Germanic	39.93	51.78	54.59	71.83	72.55	75.32	83.48
Japanese	53.56	65.04	73.74	62.71	71.08	68.26	82.19
Norwegian	59.39	78.87	73.69	83.53	89.06	78.11	91.77
Ob-Ugrian	59.58	77.87	73.35	78.04	82.55	82.09	86.04
Romance	40.48	71.28	63.16	76.37	77.55	84.51	95.62
Sinitic	27.34	28.57	30.75	72.46	74.04	–	98.95
Slavic	76.96	90.73	84.22	89.89	96.81	89.36	94.15
Global	51.83	66.64	66.99	75.88	78.45	77.36	84.84
Covington	60.61	82.42	80.18	82.52	82.52	87.80	90.24

Table 1: Scores, as percentage of total alignment pairs. Global scores does not include Covington. PMI method does not handle tonal languages, and its global score is based on the non-tonal language groups.

hot encoding cannot represent the degree of phonetic similarity between IPA segments. Viewing the one-hot encoding scores as a baseline, we conclude that the other methods' distance measures do indeed contribute to sequence alignment.

The PHOIBLE feature vectors are roughly on par with the phon2vec embeddings, yielding better results than the NN embeddings on two of the datasets (Japanese and Slavic), and are otherwise outperformed by the NN and the RNN embeddings, as well as PMI and SCA. Part of the low performance of the PHOIBLE's vectors can be due to the fact that PHOIBLE does not provide feature vectors for all IPA segments in the BDPA datasets. However, the similar performance between PHOIBLE vectors and phon2vec and, clearly better performance achieved by the NN embeddings indicates that we can learn (more) useful linguistic generalizations in a data-driven manner.

Of the data-driven methods, phon2vec yields the lowest scores, being outperformed by both neural network models in all datasets except Japanese. Given that both the phon2vec and the NN embeddings are trained on the same data, the consistent performance difference between phon2vec and NN embeddings points to usefulness of to the sequential order of IPA segments. The better performance of the RNN embeddings over other data driven methods is not surprising, as they capture useful information from the multi-lingual data set. Furthermore, the performance of RNN embeddings is similar to the PMI method, yielding better results in many data sets.

For all but the Slavic dataset, SCA yields higher scores than other methods compared in this study. The score differences exhibit considerable variance — from less than 1 percent point for the Andean dataset up to 26 percent points for the Sinitic dataset. A possible explanation for this variance is the fact that not all IPA segments found in the benchmark datasets are found in the training data. For example, NorthEuraLex includes a single tonal language, Mandarin Chinese, and the models cannot produce meaningful embeddings for most of the tones encountered in the Sinitic and Bai datasets. Arguably, a larger training dataset featuring a richer set of IPA segments would produce better-performing embeddings.

4 Conclusion

In this study we have proposed, implemented, and evaluated three methods for obtaining vector representations of IPA segments for the purposes of pairwise IPA sequence alignment. Our method outperforms a linguistically-informed baseline, as well as a trivial one-hot representation, performs comparably to a state-of-the-art data driven method. However, the performances of data driven methods, including ours, seem to be behind a linguistically-informed system, SCA. Nevertheless, the results of the data-driven methods are not too far off the mark, and we believe that they could be significantly improved by using larger and more diverse training data, and better tuning of the data-driven methods. This constitutes one direction for future experiments; another possibility is to train and use embeddings specific to a particular language family or macro-area. Further investigation is also needed with respect to comparing and evaluating the methods, especially in the context of a larger application, such as cognacy identification or phylogenetic inference.

References

Bryan Allen. 2007. *Bai Dialect Survey*. SIL International.

Jørn Almberg and Kristian Skarbø. 2011. Nordavinden og sola. En norsk dialektprøvedatabase på nettet.

Cecil H. Brown, Eric W. Holman, and Søren Wichmann. 2013. Sound Correspondences in the World's Languages. *Language*, 89(1):4–29.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv:1406.1078 [cs, stat]*.

François Chollet et al. 2015. Keras.

Michael A. Covington. 1996. An Algorithm to Align Words for Historical Comparison. *Computational Linguistics*, 22(4):481–496.

Michael A. Covington. 1998. Alignment of Multiple Languages for Historical Comparison. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*, COLING '98, pages 275–279, Stroudsburg, PA, USA. Association for Computational Linguistics.

Johannes Dellert and Gerhard Jäger, editors. 2017. *NorthEuraLex (version 0.9)*. Eberhard Karls Universität Tübingen, Tübingen.

Rick Derksen, editor. 2008. *Etymological dictionary of the Slavic inherited lexicon*. Number 4 in Leiden Indo-European Etymological Dictionary Series. Brill, Leiden and Boston.

Louis Gauchat, Jules Jeanjaquet, and Ernest Tappolet, editors. 1925. *Tableaux phonétiques des patois suisses romands.* Attinger, Neuchâtel.

Harald Hammarström, Sebastian Bank, Robert Forkel, and Martin Haspelmath, editors. 2018. *Glottolog 3.2*. Max Planck Institute for the Science of Human History, Jena.

Bruce Hayes. 2009. *Introductory Phonology*. Blackwell.

Paul Heggarty. 2006. Sounds of the Andean languages.

Jīngyī Hóu, editor. 2004. *Xiàndài Hànyǔ fāngyán yīnkù*. Shànghǎi Jiàoyù, Shànghǎi.

Eric Jones, Travis Oliphant, Pearu Peterson, et al. 2001. SciPy: Open source scientific tools for Python.

Gerhard Jäger. 2013. Phylogenetic Inference from Word Lists Using Weighted Alignment with Empirically Determined Weights. *Language Dynamics and Change*, 3(2):245–291.

Gerhard Jäger and Pavel Sofroniev. 2016. Automatic cognate classification with a support vector machine. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, pages 128–134.

Grzegorz Kondrak. 2000. A New Algorithm for the Alignment of Phonetic Sequences. In *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*, NAACL 2000, pages 288–295, Stroudsburg, PA, USA. Association for Computational Linguistics.

Grzegorz Kondrak. 2003. Phonetic alignment and similarity. *Computers and the Humanities*, 37(3):273–291.

Grzegorz Kondrak and Graeme Hirst. 2002. *Algorithms for language reconstruction*, volume 63. University of Toronto Toronto.

Johann-Mattis List. 2012. SCA: Phonetic Alignment based on sound classes. *New Directions in Logic, Language and Computation*, pages 32–51.

Johann-Mattis List and J Prokić. 2014. A benchmark database of phonetic alignments in historical linguistics and dialectology. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 288–294.

Jianqiang Ma, Çağrı Çöltekin, and Erhard Hinrichs. 2016. Learning phone embeddings for word segmentation of child-directed speech. In *Proceedings Workshop on Cognitive Aspects of Computational Language Learning*, pages 53–63.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*.

Scott R. Moisik and John H. Esling. 2011. The 'whole larynx' approach to laryngeal features. In *Proceedings of the International Congress of Phonetic Sciences (ICPhS XVII)*, pages 1406–1409.

Steven Moran, Daniel McCloy, and Richard Wright, editors. 2014. *PHOIBLE Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori S. Levin. 2016. PanPhon: A resource for mapping IPA segments to articulatory feature vectors. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3475–3484. ACL.

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.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos,

D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jelena Prokić, John Nerbonne, Vladimir Zhobov, Petya Osenova, Kiril Simov, Thomas Zastrow, and Erhard Hinrichs. 2009. The computational analysis of Bulgarian dialect pronunciation. *Serdica journal of computing*, 3(3):269–298.

Colin Renfrew and Paul Heggarty. 2009. Languages and origins in Europe.

Georges de Schutter, Boudewijn van den Berg, Ton Goeman, and Thera de Jong. 2005. Morfologische atlas van de Nederlandse dialecten.

Hattori Shirō. 1973. Japanese dialects. *Diachronic, areal and typological linguistics*, pages 368–400.

Miikka P. Silfverberg, Lingshuang Mao, and Mans Hulden. 2018. Sound Analogies with Phoneme Embeddings. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2018*, pages 136–144.

Stefan van der Walt, S. Chris Colbert, and Gael Varoquaux. 2011. The NumPy array: A structure for efficient numerical computation. *Computing in Science and Engineering*, 13:22–30.

Feng Wang. 2006. *Comparison of languages in contact. The distillation method and the case of Bai*. Institute of Linguistics Academia Sinica, Taipei.

Søren Wichmann, Eric W. Holman, and Cecil H. Brown, editors. 2016. *The ASJP Database (version 17)*. Available at http://asjp.clld.org/.

M. Zhivlov. 2011. Annotated Swadesh wordlists for the Ob-Ugrian group (Uralic family). *The Global Lexicostatistical Database*.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.

Supplementary Material

	phon2vec	experimental values
model architecture	CBOW	CBOW, skip-gram
embedding size	15	5, 15, 30
context size	2	0, 2, 4
negative samples	1	0, 1, 2
epochs	5	5, 10, 20

Table 2: Hyperparameter values, phon2vec

	nn	rnn	experimental values
embedding size	64	64	32, 64
dense layer size	128	-	64, 128
rnn layer size	-	128	64, 128
epochs	10	5	5, 10, 20
batch size	128	128	32, 64, 128
optimisation	sgd	rmsprop	

Table 3: Hyperparameter values, neural networks

	pairs	langs	family	source
Andean	619	20	Aymaran, Quechuan	Heggarty (2006)
Bai	889	17	Sino-Tibetan	Wang (2006), Allen (2007)
Bulgarian	1519	196	Indo-European	Prokić et al. (2009)
Dutch	500	62	Indo-European	Schutter et al. (2005)
French	712	62	Indo-European	Gauchat et al. (1925)
Germanic	1110	45	Indo-European	Renfrew and Heggarty (2009)
Japanese	219	10	Japonic	Shirō (1973)
Norwegian	501	51	Indo-European	Almberg and Skarbø (2011)
Ob-Ugrian	444	21	Uralic	Zhivlov (2011)
Romance	297	8	Indo-European	Renfrew and Heggarty (2009)
Sinitic	200	38	Sino-Tibetan	Hóu (2004)
Slavic	188	5	Indo-European	Derksen (2008)

Table 4: The BDPA datasets: numbers of pairs and languages, family affiliations as per Hammarström et al. (2018), and sources