Phonetic, Semantic, and Articulatory Features in Assamese-Bengali Cognate Detection

Abhijnan Nath[1], **Rahul Ghosh**[2], and **Nikhil Krishnaswamy**[1]

[1]Department of Computer Science, Colorado State University, Fort Collins, CO, USA
[2]Fossil Ridge High School, Fort Collins, CO, USA[†]
{abhijnan.nath,nkrishna}@colostate.edu

Abstract

In this paper, we propose a method to detect if words in two similar languages, Assamese and Bengali, are cognates. We mix phonetic, semantic, and articulatory features and use the cognate detection task to analyze the relative informational contribution of each type of feature to distinguish words in the two similar languages. In addition, since support for low-resourced languages like Assamese can be weak or nonexistent in some multilingual language models, we create a monolingual Assamese Transformer model and explore augmenting multilingual models with monolingual models using affine transformation techniques between vector spaces.

1 Introduction

Lexical cognates are words that are inherited by direct descent from a common etymological ancestor. Due to sound change and semantic shift, cognates may or may not be easy to detect without rigorous application of the comparative method. For example, English "two" is cognate with Armenian *erku*, as both are descended from Proto-Indo-European _*dwóh_$_1$, with _*dw->>tw-_ and _*dw->>erk-_ being regular, if non-intuitive, parallel sound changes.

Unlike loanwords, cognates are inherited and not borrowed, and are therefore necessarily subject to diachronic sound change. Application of the comparative method to cognates can be used to discern the evolutionary paths of related languages, making them very useful for historical linguists, but first cognates must be distinguished from other classes of words like ordinary translations or words that simply sound similar.

In this paper we focus on cognate detection between two closely-related languages: Bengali (ISO code bn) and Assamese (ISO code as). Bengali (262 million speakers) and Assamese (15 million speakers) are two languages of eastern India and Bangladesh. They are both official languages of India (with most speakers located in the states of West Bengal and Assam, respectively), while Bengali is also the national language of Bangladesh. They share a common descent from Early Indo-Aryan via Magadhi Prakrit, and are both typically written using Bengali or Eastern Nagari script. The Bengali-Assamese languages (or Gauda-Kamarupa languages) is the subgrouping of Eastern Indo-Aryan that contains both these languages and related dialects. They share certain grammatical features like classifying affixes (e.g., Asm. -zɔn, Beng. -dʒɔn, referring to persons), as well as certain common phonetic innovations (such as the evolution of Sanskrit /ə/→/ɔ/).

Despite the similarities, the two languages have some important differences, particularly in their sound patterns. Table 1 shows Assamese and Bengali consonants that are pronounced differently despite being written with the same letter. For instance, Assamese lenited Sanskrit /s/ to /x/ whereas Bengali palatalized it to /ʃ/. However both sounds are now written with the same letters in their respective languages—স, শ, or ষ—usually transcribed as <s> or <sh>.

Assamese	**Bengali**
s,s,z,z	tɕ,tɕʰ,dʑ,dʑʱ
t,tʰ,d,dʱ	ʈ/t̪,ʈʰ/t̪ʰ,ɖ/d̪,ɖʱ/d̪ʱ
x,ɹ	ʃ,r

Table 1: Assamese-Bengali sound correspondences.

Therefore between these two languages, phonetic features, orthographic features, semantic features, or alignment of articulatory sequences may be more or less useful in determining cognate status, depending on the specific words in question. The word এক (/ek/), meaning "one" in both languages, is a clear case of common inheritance

[†]This work conducted during an internship with the Colorado State University Department of Computer Science.

from Sanskrit with the same sound changes applied; one need only look at the orthographic and phonetic forms to see this. But for Assamese অকল (/ɔkɔl/), meaning “only,” the Bengali cognate is actually একলা (/ekla/). In Bengali, অকল is actually an Arabic loan meaning “wisdom.”

In this paper we explore the contributions of phonetic, semantic, orthographic, and articulatory alignment features to the task of cognate detection between Assamese and Bengali. We use heuristic edit distance metrics, embedding vectors from various large multilingual language models (MLMs), and neural networks to learn alignments between phonetic sequences. We also use an affine transformation technique to augment the embedding spaces of MLMs with Assamese-specific data. With combinations of features, we are able to achieve up to ∼94% F1 on cognate detection. Our results also show that embeddings from a smaller monolingual BERT variant can be mapped using affine transformations into the embedding space of larger multilingual models, which can improve both precision (up to 30%) and recall (up to 20%) in detecting Assamese cognates in Bengali.

2 Related Work

Cognate detection has been approached from many angles in the NLP community. Kondrak (2001) identifies cognates in Algonquian using phonetic and semantic similarity. Mulloni and Pekar (2006) infer orthographic changes between cognates across languages. Jäger (2018) evaluates PMI and SVM-based methods in cognate detection over the Automated Similarity Judgment Project database (Brown et al., 2008). List (2014) finds relationships between data size and genetic relatedness in automated cognate detection between English, German, Dutch, and French. Bloodgood and Strauss (2017) explore using global constraints in this task. Dellert (2018) explores sequence alignment and sound correspondence features in cognate detection in Northern European languages; these are two of the feature types we also explore here. Rama et al. (2018) and Rama and List (2019) explore the application of automated cognate detection methods to phylogenetic reconstruction and inference, and Kanojia et al. (2021a) utilize WordNets to perform orthographic similarity-based cognate detection in various Indian languages, but notably not Assamese.

Bharadwaj et al. (2016) and Rijhwani et al. (2019) suggest that phonologically-aware articulatory representations from PanPhon (Mortensen et al., 2016) can either be used natively as embeddings or as features in attention-based neural models for downstream NLP tasks such as NER or entity linking for low-resource languages. Labat and Lefever (2019) and Lefever et al. (2020) suggest that adding semantic information to orthographic features works well for cognate detection in resource-rich languages like English and Dutch (90% F1). Similarly, Kanojia et al. (2021b) suggests that adding large multilingual model embeddings to cognitive features like gaze improves cognate detection in low-resource languages like Hindi and Marathi (86% F1). Work in translation lexicons (e.g., Schafer and Yarowsky (2002)) is also relevant here, for the hybrid approach to similarity metrics used. We combine multiple approaches which, to our knowledge, have never before been used all together. Works such as Ganesan et al. (2021) and Artetxe et al. (2018a,b) improve bilingual lexical induction using either linear or non-linear word embedding maps, but they use non-contextual embeddings like fastText or word2vec. We extend such research to cognate detection using contextualized embeddings from Transformer-based models to leverage additional monolingual representations in this task.

3 Datasets

Cognates in Bengali and Assamese must share a common descent from an ancestor language[1]; the best-documented of these is Sanskrit. However, many descendants of Sanskrit make scholarly reborrowings from Sanskrit (*tatsama*) that are fully reincorporated Sanskrit forms adapted to fit the modern phonology. These exist alongside *tadbhava* words inherited from Old Indo-Aryan with concomitant sound changes in the Middle Indo-Aryan phase.

For this data collection, we turned to Wiktionary. Namely, we scraped the categories of the form `[Descendant]_terms_derived_from_Sanskrit` for each of the two descendants.[2]. We took the union of these two sets and then took the subset of the union where both the Assamese and Bengali forms had the same documented Sanskrit ancestor. Checking against com-

[1] We do not adopt the definition of cognate that subsumes loanwords (e.g., Kondrak (2001)); we use the linguistic definition that treats loanwords and cognates as distinct.

[2] e.g., https://en.wiktionary.org/wiki/Category:Assamese_terms_derived_from_Sanskrit

mon ancestry filters out loanwords from the cognate datasets. Table 2 shows the number of cognates retrieved for each language. We should note that despite the union-intersection operations being symmetrical, this does not result in equally-sized datasets for the two languages; because Bengali has more overall entries in the English Wiktionary, there are more cases where multiple Bengali words have the same ancestor as a single documented Assamese word.

Descendant	Ancestor	# Cognates
Assamese	Sanskrit	205
Bengali	Sanskrit	335

Table 2: Cognate pair counts per language.

We then convert every word in every pair to its phonetic representation in the International Phonetic Alphabet (IPA). This is done using the Epitran package (Mortensen et al., 2018). The available Epitran distribution does not support certain low-resourced languages, among them Assamese, but the format is easily extensible, and so we wrote an Epitran graph-to-phoneme mapping for Assamese using resources like Omniglot[3] and Wikiwand/Assamese[4], as well as native speaker guidance for verification.

Having gathered positive examples of cognates, we complete the datasets with word pairs that are not examples of cognates. These may be: i) **hard negatives**: phonetically similar non-cognates; ii) **synonyms**: semantically similar words, like ordinary non-cognate translations; iii) **randoms**: pairs where the two words have no discernible phonetic or semantic relationship.

To collect **hard negative** examples, we use the PanPhon package (Mortensen et al., 2016) and calculate six different edit distances between the IPA transcription of every gathered cognate in one language, and the IPA transcription for every lemma in the other language (the list of lemmas was also scraped from Wiktionary). For each edit distance, we select the word that has the lowest edit distance to the cognate in question. This returns up to six hard negatives per cognate (less if more than one edit distance metric returns the same nearest neighbor). Example: Asm. কথা (/kɔtʰa/) “word”, Beng. কটা (/kɔʈa/) “how many”.

To collect **synonyms**, we adapted our Wiktionary scraper to exploit the metadata organization of Wiktionary pages, and retrieved synonyms for each word in the collected cognates list where available. Example: Asm. কুটুম (/kutum/) “family”, Beng. রিশতাদার (/riʃt̪ad̪ar/) “relatives.”

Finally we generate the **randoms** pairings by pairing each cognate with a random word in the other language. As a final cleanup step, we remove any intersections between these three datasets and between these and the cognates dataset.

We then concatenated these subsets into three different datasets. 1) `Assamese-Bengali`, where the Assamese word is the baseline comparand to which the Bengali word is compared. 2) `Bengali-Assamese`, where the reverse is true. This is a small and subtle difference. The order of the words in word pairs between this dataset and the previous one are simply flipped, so the edit distances are symmetric, but because alignment score is calculated using a deep neural network estimator trained on randomized splits of the data, alignment scores between two reversed word pairs are similar but often not identical. 3) `All-languages`. This is a bidirectional dataset consisting of the concatenation of the previous two. In training and inference this allows the final classifier to learn from similarity metrics that flow in both directions.

The full dataset creation process for data of this size can be completed within an day, including native speaker verification. Table 11 in the Appendix gives the total train and test size of each category.

4 Methodology

Here we discuss the orthographic and phonetic features we extract from the data, our methods of assessing alignment between phonetic sequences, how we extract semantic similarity features from various language models, and how these different features combine in the cognate classification task.

4.1 Orthographic and Phonetic Similarity

Orthographic similarity is simply the Levenshtein edit distance (Levenshtein et al., 1966) between two strings. Since Assamese and Bengali use the same script with small modifications, we want to explore the importance of a simple string similarity metric as a feature in our classification task. Because of differences in the sound patterns of the two languages (see Sec. 1), phonetic distance is also important. We calculate phonetic similarity using 6 different edit distances from PanPhon

[3] https://omniglot.com/writing/assamese.htm

[4] https://www.wikiwand.com/en/Help:IPA/Assamese

over the IPA transcriptions of the word pairs in our dataset. These edit distances are: Fast Levenshtein Distance, Dolgo Prime Distance, Feature Edit Distance, Hamming Feature Distance, Weighted Feature Distance, Partial Hamming Feature Distance, all normalized by the maximum length of the two words in the pair. We hypothesize that these distance metrics collectively capture some important information about phonetic similarity between Assamese and Bengali cognate pairs.

4.2 Alignment-Scoring Network

To account for different phonotactics, epenthesis, elision, and metathesis between Assamese and Bengali, we build a model to align phonemes in the pair. This provides a more informative measure than simple edit distances.

We convert the IPA transcriptions to 21 subsegmental articulatory features using PanPhon[5]. These features include place and manner of articulation, voicing, etc., and the feature vectors were padded to the maximum length of a vector in the cognate pair. The features for word pairs in our datasets were then concatenated for input to the alignment-scoring network.

The alignment network is a two-layer deep feed-forward neural network with 512 neurons in each layer, all with ReLU activation and followed by 10% dropout. We trained for 5,000 epochs on the aforementioned concatenated features of the `All-languages` dataset (see Sec. 3), using a 80:20 train/validation split. The network was trained against the cognate/non-cognate binary label. This is not to predict cognate status directly, since we do not include any semantic information at this step, but the label acts as an rough indicator of "phonetically aligned" or not. A positive prediction means the model predicts that the two words in the pair are strongly phonetically-aligned according to the articulatory features. During inference, we get the pre-sigmoid logit value as a holistic alignment score between the two words.

4.3 Semantic Similarity

Even though cognates do not need to have similar meaning, many do preserve semantic similarity. Work such as Turton et al. (2021) suggest that contextual semantic information at the word level can be extracted from BERT and variants as embeddings. As such, we extract semantic information from both word-level and sentence-level embeddings from large multilingual Transformer-based models such as XLM-R (Conneau et al., 2020) and MBERT (Devlin et al., 2018), as well as from some smaller, Indian language-focused models: IndicBERT (Kakwani et al., 2020) and Muril (Khanuja et al., 2021).

XLM-R (100 languages) and MBERT (104 languages) are trained on multiple languages from across the globe. MBERT includes Bengali in its training data but not Assamese. XLM-R was trained with data from both languages but the Assamese training data size is a relatively small 5 million tokens, whereas the Bengali training data is over 100 times larger (and the training data of a well-resourced language like English is 100 times larger still). IndicBERT and MuRIL are focused on Indian languages and so have a larger relative training data size for languages like Assamese and Bengali. IndicBERT and MuRIL also outperform XLM and MBERT against several semantic downstream NLP task benchmarks like IndicGLUE (Kakwani et al., 2020), cross-lingual XTREME (Hu et al., 2020), etc.

4.3.1 Monolingual Assamese Model

In order to provide our cognate classifier with a potentially stronger representation of Assamese semantics, and to investigate how much information a much smaller monolingual Transformer model might be able to contribute, we trained a "light" ALBERT (`albert-base-v2`) model for 305,700 epochs with a vocabulary size of 32,000 on four publicly-available Assamese datasets: Assamese Wikidumps[6], OSCAR (Suárez et al., 2019)[7], PMIndia (Haddow and Kirefu, 2020)[8] and the Common Crawl (CC100) Assamese corpus (Conneau et al., 2020)[9] (in total, after preprocessing, around 14 million Assamese tokens) with the BERT Masked Language Model (Devlin et al., 2018) loss function. See Table 5 in the Appendix for model configuration.

4.3.2 Affine Transformations Between Embedding Spaces

Since embeddings are vectors that preserve similarity relations across dimensions, only embeddings retrieved from the same model architecture are guaranteed to be directly comparable. Absent this condition, differences in training data, training

[5] PanPhon does not contain suprasegmental or tonal information but both Bengali and Assamese are non-tonal languages.

[6] https://archive.org/details/aswiki-20220120

[7] https://oscar-corpus.com

[8] https://paperswithcode.com/dataset/pmindia

[9] https://paperswithcode.com/dataset/cc100

regime, and model architecture mean that embeddings retrieved from different models are likely to be orthogonal in most dimensions.

However, recent work in the vision community (McNeely-White et al., 2020, 2022) has demonstrated that by fitting affine matrices $M_{A \to B}$ and $M_{B \to A}$ between paired features denoting equivalent samples extracted from models A and B, features from one embedding space can be transformed to another embedding space with high fidelity. This entails solving for a mapping function $f(x; W)$ where $W \in \mathbb{R}^{d_A} \times \mathbb{R}^{d_B}$, between equivalent information samples (i.e., paired embedding vectors) from two models, using ridge regression. The aforementioned work has been applied to CNN architectures, and here we use this task to explore the application of similar principles to Transformer architectures.

The paired vectors we use to compute mappings between embedding spaces come in the form of word-level and sentence-level embeddings from the aforementioned large language models: IndicBERT, XLM-R, MBERT, MuRIL, and our Assamese ALBERT variant (Sec. 4.3.1).

Sentence-sensitive embeddings We took our list of extracted cognates and had a native speaker of each language manually create simple sentences for each word that were direct translations of each other. Sentences were of a form that was appropriate for the part of speech, left the sense of the word as unambiguous as possible, and were as simple as possible (e.g., see Table 3).

Language	Sentence	IPA
Bengali	এটি একটি টাং	eʈi ekʈi ʈaŋ
Assamese	এইটো এটা ঠেং	eitʊ eta t^{h}ɛŋ
English	This is a foot/leg	

Table 3: Sample equivalent sentences with cognate words (and English translations) underlined.

Two additional special tokens (`<m>` and `</m>`) were added to the models' vocabularies. Before getting the sentence embeddings, the cognate words were surrounded by these tokens to account for subword tokenization potentially breaking up the cognate words. We then generate binary vectors for the cognates using the indices of the special tokens in the sentence. Our model attends to these binary maps by an element-wise tensor multiplication in the forward function and outputs a contextual representation of the word. For instance, when preprocessed, the Bengali sample sentence "this is a valley" is input to the model as এটি একটি **<m>**উপত্যকা**</m>**. Sentence-sensitive embeddings were generated only from MBERT and our ALBERT variant, as the other models all have at least some support for Assamese already.

Word-level embeddings For each of the five models, we input a "sentence" formatted as `[CLS]<word>[SEP]` and use the `[CLS]` token's `last_hidden_state` to get representations for each token in each sequence of the batch from the last layer of the model, which often encodes more semantic information. Jawahar et al. (2019) and Tenney et al. (2019) suggest that BERTs later layers encode comparatively more high-level semantic information than the middle layers. The `[CLS]` token here serves the same purpose as the `<m>` tokens in the sentence-sensitive embeddings: to account for potential subword tokenization effects.

Having extracted the different embeddings from each model, we use the native embeddings from each model to find cosine similarities between the words in every pair in the data. These cosine similarities are input features into the final evaluation.

Affine mapping procedure Native model embeddings are independently useful for downstream NLP tasks, but their utility may be degraded when the language model does not robustly support the language in question. E.g., in the case of MBERT, which was not trained on Assamese, many Assamese words may be treated as out of vocabulary items and broken up into subwords that do not capture the semantics of the original word. Therefore in this case, we explore if and how linearly mapping one set of embeddings from its native space to a target model space can still act as an effective feature in this cognate detection task.

To construct the mapping, we take the word or sentence embeddings from one model as inputs, and equivalent word or sentence embeddings from another model as outputs, and fit them to each other using scikit-learn's ridge regressor. The resulting $d_A \times d_B$ transformation matrix[10] computed from a set of paired vectors serves as a bridge transformation from one embedding space to another by minimizing the distance between paired points in $\mathbb{R}^{d_A} \times \mathbb{R}^{d_B}$ feature space that share equivalent semantics. Multiplying a source embedding by this precomputed bridge matrix should result

[10] All embeddings used here are 768 dimensions, except embeddings from XLM-R, which are 1280 dimensions.

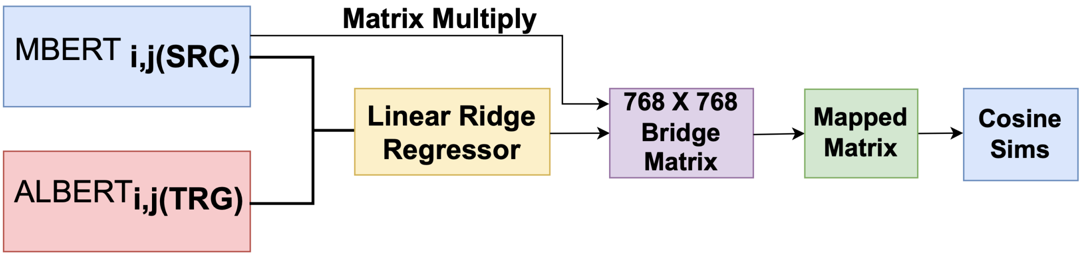

Figure 1: Cross-embedding space mapping pipeline resulting in directly comparable vector representations (MBERT→ALBERT used as example).

in approximately the same semantics in the target embedding space, meaning that a transformed embedding and one native to the target embedding space are now directly comparable using metrics like cosine similarity. Fig. 1 shows this procedure. We construct bridge matrices between the four MLMs mentioned previously, and our Assamese ALBERT variant. Like the word and sentence-sensitive embeddings, the cosine similarities between embeddings of word pairs after the mapping transformations are added to the dataset as input features to the final classification task, so we can examine all semantic similarity computations.

4.4 Evaluation

Having collected a variety of phonetic, semantic, and articulatory alignment metrics for all the paired words in our datasets, our task is now to train a classifier model to discriminate cognates from non-cognates in the data, using these features. We train two types of classification models: a logistic regressor (**LR**) and a neural network (**NN**). The NN consists of 3 layers of 512, 256, and 128 hidden units respectively, all with ReLU activation and followed by 10% dropout, and a final sigmoid activation, and is trained for 5,000 epochs with Adam optimization and BCE loss. The LR is more interpretable but the NN is better performing.

We train three versions of the model: one trained on the `All-languages` dataset, and evaluated on the test splits of that dataset and of the `Assamese-Bengali` and `Bengali-Assamese` datasets; and one each trained and evaluated only on the `Assamese-Bengali/Bengali-Assamese` datasets (pair-specific models, which are herein denoted in tables and charts with an asterisk (*) or additional label `train_ev`).

We trained all classifiers multiple times using different feature combinations to assess the contribution of different types of features. Table 4 shows the abbreviations we use in the following discussion for the different classes of features.

Abbr.	Features
`ped`	Phonetic Edit distances (PED)
`dl`	DNN logits (alignment score)
`ed`	PED with textual Levenstein dist.
`b`	All native MLMs (BERT variants)
`m`	All mappings w/o native MLMs
`ab-am`	All MLMs w/ word-level maps
`ab-sm`	All MLMs with sentence maps
`sm`	Sentence maps

Table 4: Abbreviations for feature combinations.

*`sm` - sentence maps from MBERT to ALBERT space.
*`b` - native MLM embeddings without cross-embedding space mappings (word or sentence).
*`ab-am` - includes native MLM embeddings along with word embedding maps without sentence maps

5 Results and Discussion

We achieve 94% F1, 93% recall, and 95% precision when using all features. The alignment score feature provides the greatest single boost, and we find that adding semantic information to phonetic features provides as much additional performance as adding orthographic features, though specific false positives and negatives diverge significantly.

Fig. 5 shows positive precision, recall, and F1 for the neural network classifier using all features.

	all	*bn-as*	*as-bn*	*bn-as**	*as-bn**
P(+)	95	97	94	90	90
R(+)	93	94	92	88	87
F1(+)	94	95	93	89	88

Table 5: NN classifier results (as %) for the `ed-dl-ab-am` feature combination (full feature set).

We can also see that the classifier performs very slightly better using Bengali as the baseline language than using Assamese. Similar results hold for other feature subsets: using the "bidirectional" `All-languages` model, feature sets `ed-dl-m`, `ped-dl-ab-am`, and `ed-dl-b` all show 94%

F1(+) for Bengali-Assamese but 93% F1(+) for Assamese-Bengali.

One possible reason for this is that Bengali forms are on average somewhat more conservative, tending to preserve consonant clusters more than Assamese, and in fact if we look at the false negatives for this result, we find many cases where one cognate has a consonant cluster and the other does not (see Table 6). Another possible reason may be the slightly higher number of Bengali baseline pairs in the dataset (see Sec. 3).

Bengali	Assamese
সাঁঝ (/ʃãdʑ^{ɦ}/)	সন্ধিয়া (/xɔnd^{ɦ}ija/)
শিক্ষা (/ʃikk^{h}a/)	শিকোৱা (/xikʊwa/)
মিষ্টি (/miʃʈi/)	মিঠা (/mit^{h}a/)

Table 6: Sample false negatives.

We also see that the model trained on the bidirectional data outperforms in each direction models trained on that direction alone.

The NN classifier outperforms the LR by ∼4% in all metrics. This suggests that for detecting bilingual cognates using multiple feature types, the non-linear decision boundary of a multi-layer perceptron system is better-suited to this task than the linear decision boundary of the LR.

5.1 Influence of Features

By comparing the performance of different feature subsets we can expose what features are most important to the cognate detection task and when. We also add a layer of interpretability to the results by cross-checking against the weights assigned to the different features by the LR classifier.

5.1.1 Alignment Features

The alignment score (dl) is the singular feature that most increases performance (Table 7). Adding alignment scores to just edit distances (ed) causes performance to rise approximately 17%. The logistic regressor for the ed-dl feature set gives the alignment score feature a weight of ∼3.2, making it strongly correlated with cognate status. It also performs best using the bidirectional data; with addition of alignment score, the pair-specific models perform about 4-6% lower.

5.1.2 Phonetic vs. Orthographic Features

When using only phonetic edit distances (ped), performance drops to 43% F1 in most evaluations (51% on the Assamese-Bengali pair-specific model). This is because many times Assamese-Bengali cognates are pronounced differently even if spelled similarly. Adding a textual Levenshtein distance metric (ed) can identify correspondence where phonetic edit distance struggles. The ed LR classifier gives textual Levenshtein distance a weight of ∼-2.7, a strong inverse correlation.

Feat.	*all*	*bn-as*	*as-bn*	*bn-as**	*as-bn**
ed	76	76	76	76	76
ed-dl	**93**	**93**	**92**	**86**	**88**
ped	43	43	43	42	51

Table 7: F1(+) as % with and without alignment score (dl) and Levenshtein distance features.

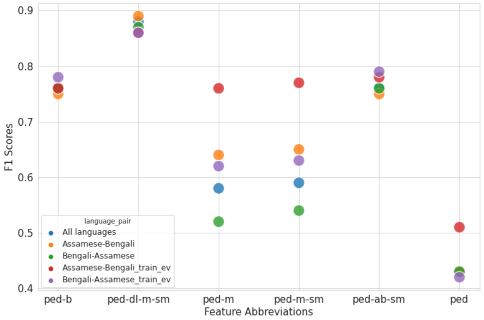

Figure 2: Influence of different semantic feature sets compared to phonetic edit distance baseline (ped).

5.1.3 Semantic Features

Addition of all the available semantic features to the ed-dl feature set results in a performance boost of only a few percentage points (cf. Tables 5 and 7). Nonetheless, by conducting further ablation tests, we can show where the semantic features actually provide important information.

Fig. 2 shows the effects of different subsets of semantic features—cosine similarities between native MLM embeddings, and between embeddings mapped from Assamese ALBERT to each MLM embedding space at the word and sentence level—compared to the lowest performing feature set, phonetic edit distances.

Adding any semantic information to phonetic features alone substantially improves performance of the neural network classifier on cognate detection. For instance, adding cosine similarities from the different pretrained MLMs (ped-b) brings performance back up to ∼76%, or on par with the inclusion of textual Levenshtein distance. For this feature set, XLM cosine similarity has the highest weight: ∼1.0, while MBERT cosine similarity is next: ∼0.4 (MuRIL: ∼0.3; IndicBERT: ∼0.06).

In terms of overall performance, adding semantic similarly to phonetic edit distance is as good as adding textual edit distance, but the specific misclassified examples in each case are quite different. Table 8 shows the breakdown of false positives by negative example type using these two different feature sets. Feature set ed has a much higher false positive rate, and also that in most cases when semantic information is used instead of textual edit distance, the proportion of false positives that are synonyms goes down, suggesting that including semantic information from MLMs improves cognate detection by mitigating misclassification of synonyms. The exception to this is in the ped-b feature set for the Assamese-Bengali pair-specific model, where 60% of false positives are synonyms, pointing to the relative weakness of Assamese semantic representations in MLMs.

	all		***bn-as****		***as-bn****	
	ed	ped-b	ed	ped-b	ed	ped-b
HN	18	12	12	11	6	4
Syn.	18	5	8	1	5	6
Rnd.	4	1	2	0	1	0

Table 8: Number of false positives using ed vs. ped-b feature sets broken down by negative example type (hard negative, synonym, random). Bidirectional and pair-specific models shown.

Word-level mappings Adding cosine similarities taken after mapping Assamese ALBERT word-level embeddings into the embedding spaces of the MLMs (ped-m) also improves performance, but the effect is more nuanced than when using native cosine similarities. For most data splits, the performance boost is not as pronounced (e.g., an appreciable but modest increase from 43% to 54% F1 on the bidirectional model evaluated against Bengali-Assamese data), but a dramatic increase in performance is seen on the Assamese-Bengali pair-specific model, where positive F1 rises to 76%, equaling the performance of the same model using the native MLM similarities. We see that the LR weight assigned to cosine similarities between the mapped Assamese ALBERT embeddings and Bengali XLM embeddings is ∼1.0 while the equivalent weight for Assamese ALBERT-Bengali MBERT mappings is ∼0.4. These weights are nearly the same as those assigned to the native XLM and MBERT cosine similarities; this and the similar NN performance indicate that these mappings are contributing the same level of information. However, weights assigned to mappings into IndicBERT or MuRIL space are both close to 0. This may be due to the larger size of the MBERT and XLM training corpora. The resultant embedding vectors in MBERT/XLM space are more dispersed, and perhaps closer to isotropic (Ethayarajh, 2019), whereas IndicBERT and MuRIL vectors appear to be clustered in a tight high-dimensional cone. This means there is more "space" in MBERT and XLM to transfer in useful semantic information through techniques like affine mapping. This is particularly interesting in the case of MBERT, which did not train on Assamese data, yet the embedding space appears able to accommodate meaningful information from Assamese embeddings.

Sentence-level mappings Adding MBERT-Assamese ALBERT cosine similarities computed after mapping the MBERT embeddings into ALBERT space using the sentence-level transformation matrix (ped-m-sm) gives a further slight boost to the neural network model. The Assamese-Bengali pair-specific model reaches 77% F1. Adding sentence-level mappings alone to phonetic edit distances increases performance over ped by only ∼6%; the combination of word and sentence-level mappings is what provides this final small boost to the Assamese-Bengali pair-specific models. Adding sentence-level mapping information also further boosts the other data splits and models by a small amount.

Examining the effect of adding sentence mappings to ped-b (ped-ab-sm), we see that this time the two pair-specific models see an appreciable improvement from 76% to 78% (Assamese-Bengali_train_ev) and 79% (Bengali-Assamese_train_ev), suggesting that similarities computed after sentence-level mappings can help language-specific models more than language-agnostic or multilingual ones.

Table 9 shows the breakdown of false positives by type of negative example using these two feature sets. Table 10 shows the breakdown of false *negatives* for ped, ped-m and ped-m-sm.

	bn-as*			***as-bn****		
	ped	pm	psm	ped	pm	psm
HN	31	48	45	47	10	15
Syn.	0	4	4	6	8	6
Rnd.	0	7	2	0	2	0

Table 9: Number of false positives in pair-specific model outputs using ped, ped-m (pm), and ped-m-sm (psm) feature sets broken down by negative example type (hard negative, synonym, random).

When compared to the phonetic edit distance baseline, the Assamese-Bengali model sees a dra-

	*bn-as**			*as-bn**		
	`ped`	`pm`	`psm`	`ped`	`pm`	`psm`
FN	212	140	138	182	106	100

Table 10: Number of false negatives (undetected cognates) in pair-specific model outputs using `ped`, `ped-m (pm)` and `ped-m-sm (psm)` feature sets.

matic reduction in false positives, mostly due to reduction in misclassified hard negatives (phonetic neighbors). Since hard negatives are semantically distant from their phonetic-neighbor cognates, introducing Assamese semantic information helps semantically disambiguate cognates from hard negatives. Adding mapped sentence-sensitive embedding similarities slightly increases the number of hard negative false positives, while also slightly reducing synonym false positives, eliminating random false positives, and further reducing false negatives. The Bengali-Assamese model actually sees *more* false positives with mappings added. This model's overall performance boost is due to fewer false negatives, while with sentence mapping the Assamese-Bengali model reduces both false positives and negatives.

The trends in Tables 8–10 show that using semantic similarities from models with relatively strong support for Bengali helps Bengali-Assamese performance, while adding mapped embedding similarities help Assamese-Bengali performance by bringing in more Assamese-specific information through affine transformation.

6 Conclusion and Future Work

We have presented here a method for detecting cognates between Bengali and Assamese that uses a mixture of phonetic, orthographic, articulatory alignment, and semantic features. The choice of these languages was motivated by their relatedness and the relative dearth of NLP work particularly on Assamese, but we believe the methods presented herein are applicable to cognate detection and other types of heterogeneous similarity-based tasks on potentially any language pair.

We found that our articulatory alignment score was by far the most informative feature. We also introduced a technique to map representations between embedding spaces and used it to introduce semantic features from a monolingual Assamese model into four large multilingual models. Adding semantic features to phonetic features alone is interesting on multiple levels—particularly using mapped instead of native embeddings.

Our ablation tests on different types of semantic representations suggest that i) linearly transforming vectors from one model's embedding space to another's carries certain semantic information with high fidelity, and ii) a model trained on a low-resource setting can be mapped to a richer model's space. If these hypotheses hold, transformed embeddings from a low-resourced LM can not only reduce the computational cost involved in training large multilingual language models but also improve downstream NLP tasks.

NLP for minority languages may benefit from being able to detect cognates in better-resourced languages, both for computational historical linguistics, and for corpus building. For instance, other languages of Assam (e.g., Mishing, Bodo) are not Indo-Aryan, but have loanwords cognate to Indo-Aryan words, alongside vocabulary cognate to other families, like Sino-Tibetan. Our phonetic and alignment techniques may facilitate creating semantic models for these severely low-resourced languages unsupported by LLMs.

Collecting putative cognates is an essential step in most applications of computational historical linguistics, allowing finding regular sound correspondences (for which our alignment method could be adapted, e.g., by training individual attention weights over a sequence), identifying shared innovations, and reconstructing earlier word forms that could be used to reconstruct proto-languages a la Bouchard-Côté et al. (2013) and Jäger (2019).

The affine mapping technique we use warrants more exploration. Not every affine map is a linear map, and other techniques like shear and rotation mapping may expose how simple a transformation can be used. Other semantic techniques we wish to explore include pairwise scoring of cognate pair embeddings using a neural network. This has been shown to work well for coreference resolution and may be applicable for cognate detection. Lastly, we would like to improve our monolingual Assamese ALBERT model and evaluate it on other downstream tasks like question answering.

Acknowledgments

This research was supported in part by a subcontract on grant award FA8750-18-2-0016 from the U.S. Defense Advanced Research Projects Agency (DARPA). Views expressed herein do not reflect the policy or position of the Department of Defense or the U.S. Government. All errors are the responsibility of the authors.

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Akash Bharadwaj, David R Mortensen, Chris Dyer, and Jaime G Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource transfer settings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1462–1472.

Michael Bloodgood and Benjamin Strauss. 2017. Using Global Constraints and Reranking to Improve Cognates Detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1983–1992.

Alexandre Bouchard-Côté, David Hall, Thomas L Griffiths, and Dan Klein. 2013. Automated reconstruction of ancient languages using probabilistic models of sound change. *Proceedings of the National Academy of Sciences*, 110(11):4224–4229.

Cecil H Brown, Eric W Holman, Søren Wichmann, and Viveka Velupillai. 2008. Automated classification of the world s languages: a description of the method and preliminary results. *Language Typology and Universals*, 61(4):285–308.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Johannes Dellert. 2018. Combining information-weighted sequence alignment and sound correspondence models for improved cognate detection. In *Proceedings of the 27th international conference on computational linguistics*, pages 3123–3133.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina N Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Kawin Ethayarajh. 2019. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65.

Ashwinkumar Ganesan, Francis Ferraro, and Tim Oates. 2021. Learning a Reversible Embedding Mapping using Bi-Directional Manifold Alignment. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3132–3139.

Barry Haddow and Faheem Kirefu. 2020. PMIndia–A Collection of Parallel Corpora of Languages of India. *arXiv preprint arXiv:2001.09907*.

Richard W Hamming. 1950. Error detecting and error correcting codes. *The Bell system technical journal*, 29(2):147–160.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Gerhard Jäger. 2018. Global-scale phylogenetic linguistic inference from lexical resources. *Scientific Data*, 5(1):1–16.

Gerhard Jäger. 2019. Computational historical linguistics. *Theoretical Linguistics*, 45(3-4):151–182.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of EMNLP*.

Diptesh Kanojia, Kevin Patel, Pushpak Bhattacharyya, Malhar Kulkarni, and Gholamreza Haffari. 2021a. Utilizing wordnets for cognate detection among indian languages. *arXiv preprint arXiv:2112.15124*.

Diptesh Kanojia, Prashant Sharma, Sayali Ghodekar, Pushpak Bhattacharyya, Gholamreza Haffari, and Malhar Kulkarni. 2021b. Cognition-aware Cognate Detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3281–3292.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. 2021. MuRIL: Multilingual representations for Indian languages. *arXiv preprint arXiv:2103.10730*.

Grzegorz Kondrak. 2001. Identifying cognates by phonetic and semantic similarity. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

Sofie Labat and Els Lefever. 2019. A Classification-Based Approach to Cognate Detection Combining Orthographic and Semantic Similarity Information. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 602–610, Varna, Bulgaria. INCOMA Ltd.

Els Lefever, Sofie Labat, and Pranaydeep Singh. 2020. Identifying cognates in English-Dutch and French-Dutch by means of orthographic information and cross-lingual word embeddings. In *PROCEEDINGS OF THE 12TH INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION (LREC 2020)*, pages 4096–4101. European Language Resources Association (ELRA).

Vladimir I Levenshtein et al. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Soviet Union.

Johann-Mattis List. 2014. Investigating the impact of sample size on cognate detection. *Journal of Language Relationship*, 11(1):91–102.

David McNeely-White, Ben Sattelberg, Nathaniel Blanchard, and Ross Beveridge. 2022. Canonical Face Embeddings. *IEEE Transactions on Biometrics, Behavior, and Identity Science*.

David McNeely-White, Benjamin Sattelberg, Nathaniel Blanchard, and Ross Beveridge. 2020. Exploring the interchangeability of CNN embedding spaces. *arXiv preprint arXiv:2010.02323*.

David R Mortensen, Siddharth Dalmia, and Patrick Littell. 2018. Epitran: Precision G2P for many languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

David R Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori 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.

Andrea Mulloni and Viktor Pekar. 2006. Automatic detection of orthographics cues for cognate recognition. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC06)*.

Taraka Rama and Johann-Mattis List. 2019. An automated framework for fast cognate detection and bayesian phylogenetic inference in computational historical linguistics. Association for Computational Linguistics.

Taraka Rama, Johann-Mattis List, Johannes Wahle, and Gerhard Jäger. 2018. Are automatic methods for cognate detection good enough for phylogenetic reconstruction in historical linguistics? *arXiv preprint arXiv:1804.05416*.

Shruti Rijhwani, Jiateng Xie, Graham Neubig, and Jaime Carbonell. 2019. Zero-shot neural transfer for cross-lingual entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6924–6931.

Mousmita Sarma and Kandarpa Sarma. 2014. *Sounds of Assamese Language*, pages 77–93.

Charles Schafer and David Yarowsky. 2002. Inducing translation lexicons via diverse similarity measures and bridge languages. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*. Leibniz-Institut für Deutsche Sprache.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Jacob Turton, Robert Elliott Smith, and David Vinson. 2021. Deriving Contextualised Semantic Features from BERT (and Other Transformer Model) Embeddings. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 248–262.

A Appendices

A.1 Sample Breakdown by Label

Table 11 gives breakdown of the Assamese-Bengali and Bengali-Assamese train/test splits based on their labels. Since we distinguish cognates from loanwords but otherwise do not single out loanwords in our datasets, loanwords may exist in the other categories. Given the phonetic similarity between loanwords and their sources, where loanwords do exist in our data, they are overwhelmingly likely to be in the hard negative category.

	as-bn		***bn-as***	
	`train`	`test`	`train`	`test`
Cog.	306	303	306	300
HN	776	769	721	716
Syn.	329	327	317	316
Rnd.	304	301	304	299
Total	1715	1700	1648	1631

Table 11: Number of Hard-Negatives (HN), Synonyms (Syn.), Cognates (Cog.), and Random pairs (Rnd.) in Assamese-Bengali and Bengali-Assamese train/test sets.

A.2 ALBERT (Monolingual Assamese Configuration)

Table 12 gives configuration details of the monolingual Assamese Transformer model that we trained for this research.

A.3 Further Details on Effects of Phonetic Features

Of the 6 phonetic edit distances we used, Hamming Feature Distance (divided by maximum length) and Partial Hamming Distance (divided by maximum length) appear to be the most correlated with cognate status according to the weights assigned to them by the logistic regressor. This suggests that Hamming distance's (Hamming, 1950) focus on using the minimum number of substitutions to transform one string into another works well for similar languages like Assamese and Bengali where most individual phonemes are largely preserved between cognate words.

Interestingly, the Dolgo Prime Distance variant gets a low (usually negative) weight in almost all feature combinations. This is interesting and suggests that Dolgo Prime Distance is not useful here due to it unduly conflating multiple phonemes into the same class. The Dolgopolsky-inspired stable phoneme classes used by PanPhon places /ʃ/ in the "coronal fricatives" class, while /x/ is in the "velar/postvelar obstruents" class. The unvoiced velar fricative /x/ is unique to Assamese and rare among Indian languages (Sarma and Sarma, 2014) and we know well that Bengali and Assamese have a regular /ʃ/-/x/ sound correspondence. So, as Dolgo Prime distance splits these up into different classes, when using this metric cognate words containing these corresponding sounds will have phonetic distance added to them when in fact they are regularly corresponding.

Parameters	**Config**
architecture	AlbertForMaskedLM
attention_probs_dropout_prob	0.1
bos_token_id	2
classifier_dropout_prob	0.1
embedding_size	128
eos_token_id	3
hidden_act	gelu
hidden_dropout_prob	0.1
hidden_size	768
initializer_range	0.02
inner_group_num	1
intermediate_size	3072
layer_norm_eps	1e-05
max_position_embeddings	514
num_attention_heads	12
num_hidden_groups	1
num_hidden_layers	6
position_embedding_type	“absolute"
transformers_version	“4.18.0"
vocab_size	32001

Table 12: ALBERT Model configuration trained on monolingual Assamese corpus.