Deciphering Undersegmented Ancient Scripts Using Phonetic Prior

Jiaming Luo
CSAIL, MIT
j_luo@csail.mit.edu

Frederik Hartmann
University of Konstanz
frederik.hartmann@uni-konstanz.de

Enrico Santus
Bayer
enrico.santus@bayer.com

Regina Barzilay
CSAIL, MIT
regina@csail.mit.edu

Yuan Cao
Google Brain
yuancao@google.com

Abstract

Most undeciphered lost languages exhibit two characteristics that pose significant decipherment challenges: (1) the scripts are not fully segmented into words; (2) the closest known language is not determined. We propose a decipherment model that handles both of these challenges by building on rich linguistic constraints reflecting consistent patterns in historical sound change. We capture the natural phonological geometry by learning character embeddings based on the International Phonetic Alphabet (IPA). The resulting generative framework jointly models word segmentation and cognate alignment, informed by phonological constraints. We evaluate the model on both deciphered languages (Gothic, Ugaritic) and an undeciphered one (Iberian). The experiments show that incorporating phonetic geometry leads to clear and consistent gains. Additionally, we propose a measure for language closeness which correctly identifies related languages for Gothic and Ugaritic. For Iberian, the method does not show strong evidence supporting Basque as a related language, concurring with the favored position by the current scholarship.[1]

1 Introduction

All the known cases of lost language decipherment have been accomplished by human experts, oftentimes over decades of painstaking efforts. At least a dozen languages are still undeciphered today. For some of those languages, even the most fundamental questions pertaining to their origins and connections to known languages are shrouded in mystery, igniting fierce scientific debate among humanities scholars. Can NLP methods be helpful in bringing some clarity to these questions? Recent work has already demonstrated that algorithms can successfully decipher lost languages like Ugaritic and Linear B (Luo et al., 2019), relying only on non-parallel data in known languages—Hebrew and Ancient Greek, respectively. However, these methods are based on assumptions that are not applicable to many undeciphered scripts.

The first assumption relates to the knowledge of language family of the lost language. This information enables us to identify the closest living language, which anchors the decipherment process. Moreover, the models assume significant proximity between the two languages so that a significant portion of their vocabulary is matched. The second assumption presumes that word boundaries are provided that uniquely define the vocabulary of the lost language.

One of the famous counterexamples to both of these assumptions is Iberian. The Iberian scripts are undersegmented with inconsistent use of word dividers. At the same time, there is no definitive consensus on its close known language—over the years, Greek, Latin, and Basque were all considered as possibilities.

In this paper, we introduce a decipherment approach that relaxes the above assumptions. The model is provided with undersegmented inscriptions in the lost language and the vocabulary in a known language. No assumptions are made about the proximity between the lost and the known languages and the goal is to match spans in the lost texts with known tokens. As a byproduct of this model, we propose a measure of language closeness that drives the selection of the best target language from the wealth of world languages.

[1]Code and data available at https://github.com/j-luo93/DecipherUnsegmented/.

Transactions of the Association for Computational Linguistics, vol. 9, pp. 69–81, 2021. https://doi.org/10.1162/tacl_a_00354
Action Editor: Hinrich Schütze. Submission batch: 7/2020; Revision batch: 8/2020; Published 02/2021.

Given the vast space of possible mappings and the scarcity of guiding signal in the input data, decipherment algorithms are commonly informed by linguistic constraints. These constraints reflect consistent patterns in language change and linguistic borrowings. Examples of previously utilized constraints include skewness of vocabulary mapping, and monotonicity of character-level alignment within cognates. We further expand the linguistic foundations of decipherment models, and incorporate phonological regularities of sound change into the matching procedure. For instance, a velar consonant [k] is unlikely to change into a labial [m]. Another important constraint in this class pertains to sound preservation, that is, the size of phonological inventories is largely preserved during language evolution.

Our approach is designed to encapsulate these constraints while addressing the segmentation issue. We devise a generative framework that jointly models word segmentation and cognate alignment. To capture the natural phonological geometry, we incorporate phonological features into character representations using the International Phonetic Alphabet (IPA). We introduce a regularization term to explicitly discourage the reduction of the phonological system and employ an edit distance-based formulation to model the monotonic alignment between cognates. The model is trained in an end-to-end fashion to optimize both the quality and the coverage of the matched tokens in the lost texts.

The ultimate goal of this work is to evaluate the model on an undeciphered language, specifically Iberian. Given how little is known about the language, it is impossible to directly assess prediction accuracy. Therefore, we adopt two complementary evaluation strategies to analyze model performance. First, we apply the model to deciphered ancient languages, Ugaritic and Gothic, which share some common challenges with Iberian. Second, we consider evaluation scenarios that capitalize on a few known facts about Iberian, such as personal names, and report the model's accuracy against these ground truths.

The results demonstrate that our model can robustly handle unsegmented or undersegmented scripts. In the Iberian personal name experiment, our model achieves a top 10 accuracy score of 75.0%. Across all the evaluation scenarios, incorporating phonological geometry leads to clear and consistent gains. For instance, the model informed by IPA obtains 12.8% increase in Gothic-Old Norse experiments. We also demonstrate that the proposed unsupervised measure of language closeness is consistent with historical linguistics findings on known languages.

2 Related Work

Non-parallel Machine Translation At a high level, our work falls into research on non-parallel machine translation. One of the important recent advancements in that area is the ability to induce accurate crosslingual lexical representations without access to parallel data (Lample et al., 2018b,a; Conneau and Lample, 2019). This is achieved by aligning embedding spaces constructed from large amounts of monolingual data. The size of data for both languages is key: High-quality monolingual embeddings are required for successful matching. This assumption, however, does not hold for ancient languages, where we can typically access a few thousands of words at most.

Decoding Cipher Texts Man-made ciphers have been the focal point for most of the early work on decipherment. They usually use EM algorithms, which are tailored towards these specific types of ciphers, most prominently substitution ciphers (Knight and Yamada, 1999; Knight et al., 2006). Later work by Nuhn et al. (2013), Hauer et al. (2014), and Kambhatla et al. (2018) addresses the problem using a heuristic search procedure, guided by a pretrained language model. To the best of our knowledge, these methods developed for tackling man-made ciphers have so far not been successfully applied to archaeological data. One contributing factor could be the inherent complexity in the evolution of natural languages.

Deciphering Ancient Scripts Our research is most closely aligned with computational decipherment of ancient scripts. Prior work has already featured several successful instances of ancient language decipherment previously done by human experts (Snyder et al., 2010; Berg-Kirkpatrick and Klein, 2013; Luo et al., 2019). Our work incorporates many linguistic insights about the structure of valid alignments introduced in prior work, such as monotonicity. We further expand the linguistic foundation by incorporating phonetic regularities that have been beneficial in early, pre-neural decipherment work (Knight et al.,

2006). However, our model is designed to handle challenging cases not addressed by prior work, where segmentation of the ancient scripts is unknown. Moreover, we are interested in dead languages without a known relative and introduce an unsupervised measure of language closeness that enables us to select an informative known language for decipherment.

3 Model

We design a model for the automatic extraction of cognates[2] directly from unsegmented or undersegmented texts (detailed setting in Section 3.1). In order to properly handle the uncertainties caused by the issue of segmentation, we devise a generative framework that composes the lost texts using smaller units—from characters to tokens, and from tokens to inscriptions. The model is trained in an end-to-end fashion to optimize both the quality and the coverage of the matched tokens.

To help the model navigate the complex search space, we consider the following linguistic properties of sound change, including phonology and phonetics in our model design:

- **Plausibility of sound change**: Similar sounds rarely change into drastically different sounds. This pattern is captured by the natural phonological geometry in human speech sounds and we incorporate relevant phonological features into the representation of characters.

- **Preservation of sounds**: The size of phonological inventories tends to be largely preserved over time. This implies that total disappearance of any sound is uncommon. In light of this, we use a regularization term to discourage any sound loss in the phonological system of the lost language.

- **Monotonicity of alignment**: The alignment between any matched pair is predominantly monotonic, which means that character-level alignments do not cross each other. This property inspires our edit distance-based formulation at the token level.

[2]Throughout this paper, the term *cognate* is liberally used to also include loanwords, as the sound correspondences in cognates and loanwords are both regular, although usually different.

To reason about phonetic proximity, we need to find character representation that explicitly reflects its phonetic properties. One such representation is provided by the IPA, where each character is represented by a vector of phonological features. As an example, consider IPA representation for two phonetically close characters [b] and [p] (See Figure 3), which share two identical coordinates. To further refine this representation, the model learns to embed these features into a new space, optimized for the decipherment task.

3.1 Problem Setting

We are given a list of *unsegmented* or *undersegmented* inscriptions $\mathcal{X} = \{X\}$ in the lost language, and a vocabulary, that is, a list of tokens $\mathcal{Y} = \{y\}$ in the known language. For each lost text X, the goal is to identify a list of non-overlapping spans $\{x\}$ that correspond to cognates in $\mathcal{Y}$. We refer to these spans as **matched spans** and any remaining character as **unmatched spans**.

We denote the character sets of the lost and the known languages by $C^{\mathbf{L}} = \{c^{\mathbf{L}}\}$ and $C^{\mathbf{K}} = \{c^{\mathbf{K}}\}$, respectively. To exploit the phonetic prior, IPA transcriptions are used for $C^{\mathbf{K}}$, while orthographic characters are used for $C^{\mathbf{L}}$. For this paper, we only consider alphabetical scripts for the lost language.[3]

3.2 Generative Framework

We design the following generative framework to handle the issue of segmentation. It jointly models segmentation and cognate alignment, which requires different treatments for matched spans and unmatched spans. An overview of the framework is provided in Figure 1 and a graphical model representation in Figure 2.

For matched spans, we introduce two latent variables: y representing the corresponding cognate in the known language and a indicating the character alignment between x and y (see the *Token* box in Figure 1). More concretely, $a = \{a_\tau\}$ is a sequence of indices, with a_τ representing the aligned position for y_τ in x. The lost token is generated by applying a character-level mapping to y according to the alignment

[3]Given that the known side uses IPA, an alphabetical system, having an alphabetical system on the lost side makes it much easier to enforce the linguistic constraints in this paper. For other types of scripts, it requires more thorough investigation, which is beyond the scope of this work.

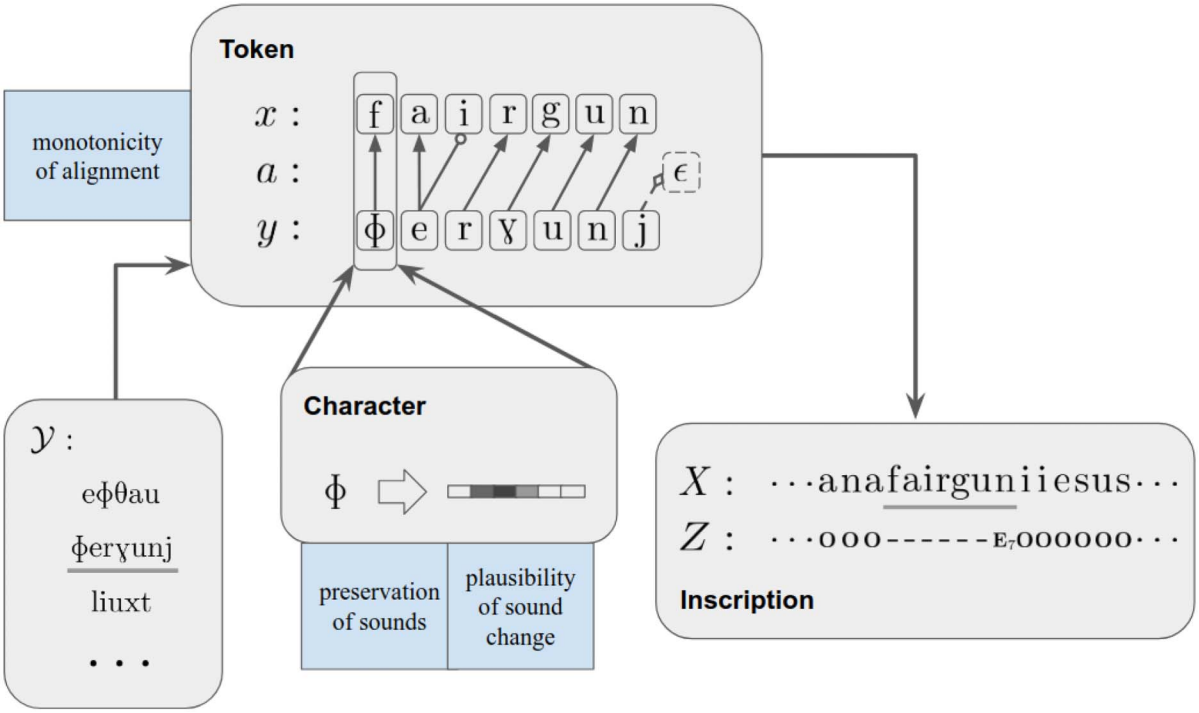

Figure 1: An overview of our framework, which generates the lost texts from smaller units—from characters to tokens and from tokens to inscriptions. Character mappings are first performed on the phonetic alphabet of the known language. Based on these mappings, a token y in the known vocabulary $\mathcal{Y}$ is converted into a token x in the lost language according to the latent alignment variable a. Lastly, all generated tokens, together with characters in unmatched spans, are concatenated to form a lost inscription. Blue boxes display the corresponding linguistic properties associated with each level of modeling.

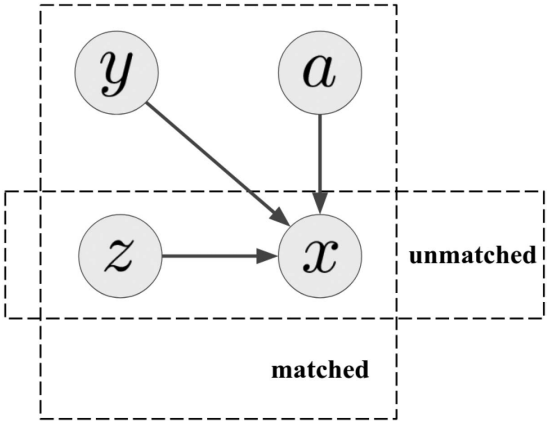

Figure 2: A graphical model representation for our framework to generate a span x. Characters in unmatched spans are generated in an independent and identically distributed fashion whereas matched spans are additionally conditioned on two latent variables: y representing a known cognate and a the character-level alignment between x and y.

provided by a. For unmatched spans, we assume each character is generated in an independent and identically distributed fashion under a uniform distribution $p_0 = \frac{1}{|C^{\text{L}}|}$.

Whether a span is matched or not is indicated by another latent variable z, and the corresponding span is denoted by x_z. More specifically, each character in an unmatched span is tagged by $z = \mathbf{O}$, whereas the entirety of a matched span of length l is marked by $z = \mathbf{E_l}$ at the end of the span (see the *Inscription* box in Figure 1). All spans are then concatenated to form the inscription, with a corresponding (sparse) tag sequence $Z = \{z\}$.

Under this framework, we arrive at the following derivation for the marginal distribution for each lost inscription X:

$$\Pr(X) = \sum_{Z} \Big[\prod_{z \in Z} \Pr(z) \Big] \Big[\prod_{\substack{z \in Z \\ z = \mathbf{O}}} p_0 \Big] \Big[\prod_{\substack{z \in Z \\ z \neq \mathbf{O}}} \Pr(x_z | z) \Big], \tag{1}$$

where $\Pr(x_z|z \neq \mathbf{O})$ is further broken down into individual character mappings:

$$\begin{aligned}\Pr(x_z|z \neq \mathbf{O}) &= \sum_{y\in\mathcal{Y}}\sum_{a\in\mathcal{A}} \Pr(y)\Pr(a)\cdot\Pr(x_z|y,z,a) \\ &\propto \sum_{y\in\mathcal{Y}}\sum_{a\in\mathcal{A}} \Pr(x_z|y,z,a) \\ &\approx \sum_{y\in\mathcal{Y}} \max_a \Pr(x_z|y,z,a) \\ &= \sum_{y\in\mathcal{Y}} \max_a \prod_\tau \Pr(x_{a_\tau}|y_\tau), \end{aligned} \tag{2}$$

Note that we assume a uniform prior for both y and a, and use the maximum to approximate the sum of $\Pr(x_z|y,z,a)$ over the latent variable a. $\mathcal{A}$ is the set of valid alignment values to be detailed in § 3.2.2.

3.2.1 Phonetics-aware Parameterization

The character mapping distributions are specified as follows:

$$\begin{aligned}&\Pr_\theta(x_{a_\tau} = c_j^{\mathbf{L}}|y_\tau = c_i^{\mathbf{K}}) \\ &\propto \exp\left(\frac{E^{\mathbf{L}}(c_j^{\mathbf{L}})\cdot E^{\mathbf{K}}(c_i^{\mathbf{K}})}{T}\right),\end{aligned} \tag{3}$$

where T is a temperature hyperparameter, $E^{\mathbf{L}}(\cdot)$ and $E^{\mathbf{K}}(\cdot)$ are the embedding functions for the lost characters and the known characters, respectively, and θ is the collection of all trainable parameters (i.e., the embeddings).

In order to capture the similarity within certain sound classes, we use IPA embeddings to represent each IPA character in the known language. More specifically, each IPA character is represented by a vector of phonological features. The model learns to embed these features into a new space and the full IPA embedding for $c^{\mathbf{K}}$ is composed by concatenating all of its relevant feature embeddings. For the example in Figure 3, the phone [b] can be represented as the concatenation of the `voiced` embedding, the `stop` embedding, and the `labial` embedding.

This compositional structure encodes the natural geometry existent in sound classes (Stevens, 2000) and biases the model towards utilizing such a structure. By design, the representations for [b] and [p] are close as they share the same values for two out of three feature groups. This structural bias is crucial for realistic character mappings.

For the lost language, we represent each character $c_j^{\mathbf{L}}$ as a weighted sum of IPA embeddings

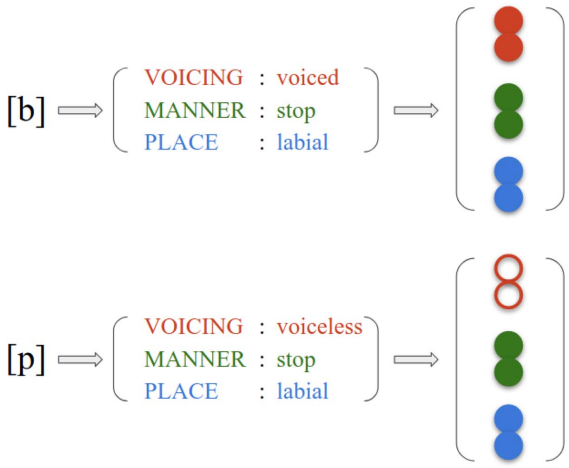

Figure 3: An illustration of IPA embeddings. Each phone is first represented by a vector of phonological features. The model first embeds each feature and then IPA embedding is obtained by concatenating all its relevant feature embeddings. For instance, the phone [b] can be represented as the concatenation of the `voiced`, `stop`, and the `labial` embeddings.

on the known side. Specifically,

$$E^{\mathbf{L}}(c_j^{\mathbf{L}}) = \sum_i w_{i,j}\cdot E^{\mathbf{K}}(c_i^{\mathbf{K}}), \tag{4}$$

where $\{w_{i,j}\}$ are learnable parameters.

3.2.2 Monotonic Alignment and Edit Distance

Individual characters in the known token y are mapped to a lost token x according to the alignment variable a. The monotonic nature of character alignment between cognate pairings motivates our design of an edit distance-based formulation to capture the dominant mechanisms involved in cognate pairings: substitutions, deletions, and insertions (Campbell, 2013). In addition to a_τ taking the value of an integer signifying the substituted position, a_τ can be ϵ, which indicates that y_τ is deleted. To model insertions, $a_\tau = (a_{\tau,1}, a_{\tau,2})$ can be two[4] adjacent indices in x.

This formulation inherently defines a set $\mathcal{A}$ of valid values for the alignment. Firstly, they are monotonically increasing with respect to τ, with the exception of ϵ. Secondly, they cover every index of x, which means every character in x is accounted for by some character in y. The *Token* box in Figure 1 showcases such an example with

[4]Insertions of even longer character sequences are rare.

all three types of edit operations. More concretely, we have the following alignment model:

$$\Pr(x_{a_\tau}|y_\tau) = \begin{cases} cc\Pr_\theta(x_{a_\tau}|y_\tau) & \text{(substitution)} \\ \Pr_\theta(\epsilon|y_\tau) & \text{(deletion)} \\ \Pr_\theta(x_{a_{\tau,1}}|y_\tau) & \\ \cdot\alpha\Pr_\theta(x_{a_{\tau,2}}|y_\tau) & \text{(insertion)} \end{cases}$$

where $\alpha \in [0, 1]$ is a hyperparameter to control the use of insertions.

3.3 Objective

Given the generative framework, our training objective is designed to optimize the quality of the extracted cognates, while matching a reasonable proportion of the text.

Quality We aim to optimize the *quality* of matched spans under the posterior distribution $\Pr(Z|X)$, measured by a scoring function $\Phi(X, Z)$. $\Phi(X, Z)$ is computed by aggregating the likelihoods of these matched spans normalized by length. The objective is defined as follows:

$$Q(X) = \mathbb{E}_{Z\sim\Pr(Z|X)}\Phi(X, Z), \quad (5)$$

$$\Phi(X, Z) = \sum_{\substack{z\in Z \\ z\neq\mathbf{O}}} \phi(x_z, z), \quad (6)$$

$$\phi(x_z, z) = \Pr(x_z|z)^{\frac{1}{|x_z|}}. \quad (7)$$

This term encourages the model to explicitly focus on improving the probability of generating the matched spans.

Regularity and Coverage The regularity of sound change, as stated by the Neogrammarian hypothesis (Campbell, 2013), implies that we need to find a reasonable number of matched spans. To achieve this goal, we incur a penalty if the expected coverage ratio of the matched characters under the posterior distribution falls below a given threshold r_{cov}:

$$\Omega_{\text{cov}}(\mathcal{X}) = \max\left(r_{\text{cov}} - \frac{\sum_{X\in\mathcal{X}} \textsc{cov}(X)}{|\mathcal{X}|}, 0.0\right)$$

$$\textsc{cov}(X) = \mathbb{E}_{Z\sim\Pr(Z|X)}\Psi(X, Z), \quad (8)$$

$$\Psi(X, Z) = \sum_{\substack{z\in Z \\ z\neq\mathbf{O}}} \psi(x_z, z) = \sum_{\substack{z\in Z \\ z\neq\mathbf{O}}} |x_z|. \quad (9)$$

Note that the ratio is computed on the entire corpus $\mathcal{X}$ instead of individual texts X because the coverage ratio can vary greatly for different individual texts. The hyperparameter r_{cov} controls the expected overlap between two languages, which enables us to apply the method even when languages share some loanwords but are not closely related.

Preservation of Sounds The size of phonological inventories tends to be largely preserved over time. This implies that total disappearance of any sound is uncommon. To reflect this tendency, we introduce an additional regularization term to discourage any sound loss. The intuition is to encourage any lost character to be mapped to exactly one[5] known IPA symbol. Formally, we have the following term

$$\Omega_{\text{loss}}(C^{\mathbf{L}}, C^{\mathbf{K}}) = \sum_{c^{\mathbf{L}}} \Big(\sum_{c^{\mathbf{K}}} \Pr(c^{\mathbf{L}}|c^{\mathbf{K}}) - 1.0\Big)^2.$$

Final Objective Putting the terms together, we have the following final objective:

$$\mathcal{S}(\mathcal{X}; C^{\mathbf{L}}, C^{\mathbf{K}}) = \sum_{X\in\mathcal{X}} Q(X) + \lambda_{\text{cov}}\Omega_{\text{cov}}(\mathcal{X}) + \lambda_{\text{loss}}\Omega_{\text{loss}}(C^{\mathbf{L}}, C^{\mathbf{K}}), \quad (10)$$

where λ_{cov} and λ_{loss} are both hyperparameters.

3.4 Training

Training with the final objective involves either finding the best latent variable, as in Equation (2), or computing the expectation under a distribution that involves one latent variable, as in Equation (5) and Equation (8). In both cases, we resort to dynamic programming to facilitate efficient computation and end-to-end training. We refer interested readers to Appendix A.1 for more detailed derivations. We illustrate one training step in Algorithm 1.

4 Experimental Setup

Our ultimate goal is to evaluate the decipherment capacity for unsegmented lost languages, without information about a known counterpart. Iberian fits both of these criteria. However, our ability to evaluate decipherment of Iberian is limited because a full ground truth is not known. Therefore, we supplement our evaluation on Iberian with more complete evaluation on lost languages with known translation, such as Gothic and Ugaritic.

[5]We experimented with looser constraints (e.g., with *at least* instead of *exactly* one correspondence), but obtained worse results.

Algorithm 1: One training step for our decipherment model

Input: One batch of lost inscriptions $\tilde{\mathcal{X}}$, entire known vocabulary $Y = \{y\}$
Parameters: Feature embeddings θ

1: $\Pr(c_j^{\mathbf{L}}|c_i^{\mathbf{K}}) \leftarrow ComputeCharDistr(\theta)$ ▷ Compute character mapping distributions (Section 3.2.1)
2: $\Pr(x|y) \leftarrow EditDistDP\big(x, y, \Pr(c_j^{\mathbf{L}}|c_i^{\mathbf{K}})\big)$ ▷ Compute token alignment probability (Section 3.2.2)
3: $\mathcal{S}(\tilde{\mathcal{X}}; C^{\mathbf{L}}, C^{\mathbf{K}}) \leftarrow \textit{WordBoundaryDP}\big(\Pr(x|y)\big)$ ▷ Compute final objective (Section 3.3)
4: $\theta \leftarrow \textit{SGD}(\mathcal{S})$ ▷ Backprop and update parameters

Language	Family	Source	#Tokens	Segmentation Situation	Century
Gothic	Germanic	Wulfila†	40,518	unsegmented	3–10 AD
Ugaritic	Semitic	Snyder et al. (2010)	7,353††	segmented	14–12 BC
Iberian	unclassified	Hesperia‡	3,466‡‡	undersegmented	6–1 BC

† `http://www.wulfila.be/gothic/download/`.
†† `http://hesperia.ucm.es/`. Iberian language is semi-syllabic, but this database has already transliterated the inscriptions into Latin scripts.
‡ This dataset directly provides the Ugaritic vocabulary, i.e., each word occurs exactly once.
‡‡ Since the texts are undersegmented and we do not know the ground truth segmentations, this represents the number of unsegmented *chunks*, each of which might contain multiple tokens.

Table 1: Basic information about the lost languages.

4.1 Languages

We focus our description on the Gothic and Iberian corpora that we compiled for this paper. Ugaritic data was reused from the prior work on decipherment (Snyder et al., 2010). Table 1 provides statistics about these languages. To evaluate the validity for our proposed language proximity measure, we additionally include six known languages: Spanish (Romance), Arabic (Semitic), Hungarian (Uralic), Turkish (Turkic), classical Latin (Latino-Faliscan), and Basque (isolate).

Gothic Several features of Gothic make it an ideal candidate for studying decipherment models. Because Gothic is fully deciphered, we can compare our predictions against ground truth. Like Iberian, Gothic is unsegmented. Its alphabet was adapted from a diverse set of languages: Greek, Latin, and Runic, but some characters are of unknown origin. The latter were in the center of decipherment efforts on Gothic (Zacher, 1855; Wagner, 2006). Another appealing feature of Gothic is its relatedness to several known Germanic languages that exhibit various degree of proximity to Gothic. The closest is its reconstructed ancestor Proto-Germanic, with Old Norse and Old English being more distantly related to Gothic. This variation in linguistic proximity enables us to study the robustness of decipherment methods to the historical change in the source and the target.

Iberian Iberian serves as a real test scenario for automatic methods—it is still undeciphered, withstanding multiple attempts over at least two centuries. Iberian scripts present two issues facing many undeciphered languages today: undersegmentation and lack of a well-researched relative. Many theories of origin have been proposed in the past, most notably linking Iberian to Basque, another non-Indo-European language on the Iberian peninsula. However, due to a lack of conclusive evidence, the current scholarship favors the position that Iberian is not genetically related to any living language. Our knowledge of Iberian owes much to the phonological system proposed by Manuel Gómez Moreno in the mid 20th century, based on fragmentary evidences such as bilingual coin legends (Sinner and Velaza, 2019). Another area with a broad consensus relates to Iberian personal names, thanks to a key Latin epigraph, *Ascoli Bronze*, which recorded the grant of Roman citizenship to Iberian soldiers who had fought for Rome (Martí et al., 2017). We use these personal names recorded in Latin as the known vocabulary.

WR†	Known language			
	Proto-Germanic (PG)	Old Norse (ON)	Old English (OE)	avg††
0%	0.820 / 0.749 / 0.863	0.213 / 0.397 / 0.597	0.046 / 0.204 / 0.497	0.360 / 0.450 / 0.652
25%	0.752 / 0.734 / 0.826	0.312 / 0.478 / 0.610	0.128 / 0.328 / 0.474	0.398 / 0.513 / 0.637
50%	0.752 / 0.736 / 0.848	0.391 / 0.508 / 0.643	0.169 / 0.404 / 0.495	0.438 / 0.549 / 0.662
75%	0.761 / 0.732 / 0.866	0.435 / 0.544 / 0.682	0.250 / 0.447 / 0.533	0.482 / 0.574 / 0.693
avg‡	0.771 / 0.737 / 0.851	0.338 / 0.482 / 0.633	0.148 / 0.346 / 0.500	0.419 / 0.522 / 0.661

† Short for *whitespace ratio*.
‡ Averaged over all whitespace ratio values.
†† Averaged over all known languages.

Table 2: Main results on Gothic in a variety of settings using A@10 scores. All scores are reported in the format of triplets, corresponding to `base` / `partial` / `full` models. In general, more phonological knowledge about the lost language, more segmentations improve the model performance. The choice of the known language also plays a significant role as Proto-Germanic has a noticeably higher score than the other two choices.

4.2 Evaluation

Stemming and Segmentation Our matching process operates at the *stem* level for the known language, instead of *full words*. Stems are more consistently preserved during language change or linguistic borrowings. While we always assume that gold stems are provided for the known language, we estimate them for the lost language.

The original Gothic texts are only segmented into sentences. To study the effect of having varying degrees of prior knowledge about the word segmentations, we create separate datasets by randomly inserting ground truth segmentations (i.e., whitespaces) with a preset probability to simulate undersegmentation scenarios.

Model Variants In multiple decipherment scenarios, partial information about phonetic assignments is available. This is the case with both Iberian and Gothic. Therefore, we evaluate performance of our model with respect to available phonological knowledge for the lost language. The `base` model assumes no knowledge while the `full` model has full knowledge of the phonological system and therefore the character mappings. For the Gothic experiment, we additionally experiment with a `partial` model that assumes that we know the phonetic values for the characters *k*, *l*, *m*, *n*, *p*, *s*, and *t*. The sound values of these characters can be used as prior knowledge as they closely resemble their original counterparts in Latin or Greek alphabets. These known mappings are incorporated through an additional term which encourages the model to match its predicted distributions with the ground truths.

In scenarios with full segmentations where it is possible to compare with previous work, we report the results for the `Bayesian` model proposed by Snyder et al. (2010) and `NeuroCipher` by Luo et al. (2019).

Metric We evaluate the model performance using top K accuracy (A@K) scores. The prediction (i.e., the stem-span pair) is considered correct if and only if the stem is correct and the the span is the prefix of the ground truth. For instance, the ground truth for the Gothic word *garda* has the stem *gard* spanning the first four letters, matching the Old Norse stem *garð*. We only consider the prediction as correct if it correctly matches *garð* and the predicted span starts with the first letter.

5 Results

Decipherment of Undersegmentated Texts Our main results on Gothic in Table 2 demonstrate that our model can effectively extract cognates in a variety of settings. Averaged over all choices of whitespace ratios and known languages (bottom right), our `base`/`partial`/`full` models achieve A@10 scores of 0.419/0.522/0.661, respectively. Not surprisingly, access to additional knowledge either about phonological mappings and/or segmentation lead to improved performance. See Table 5 for an example of model predictions.

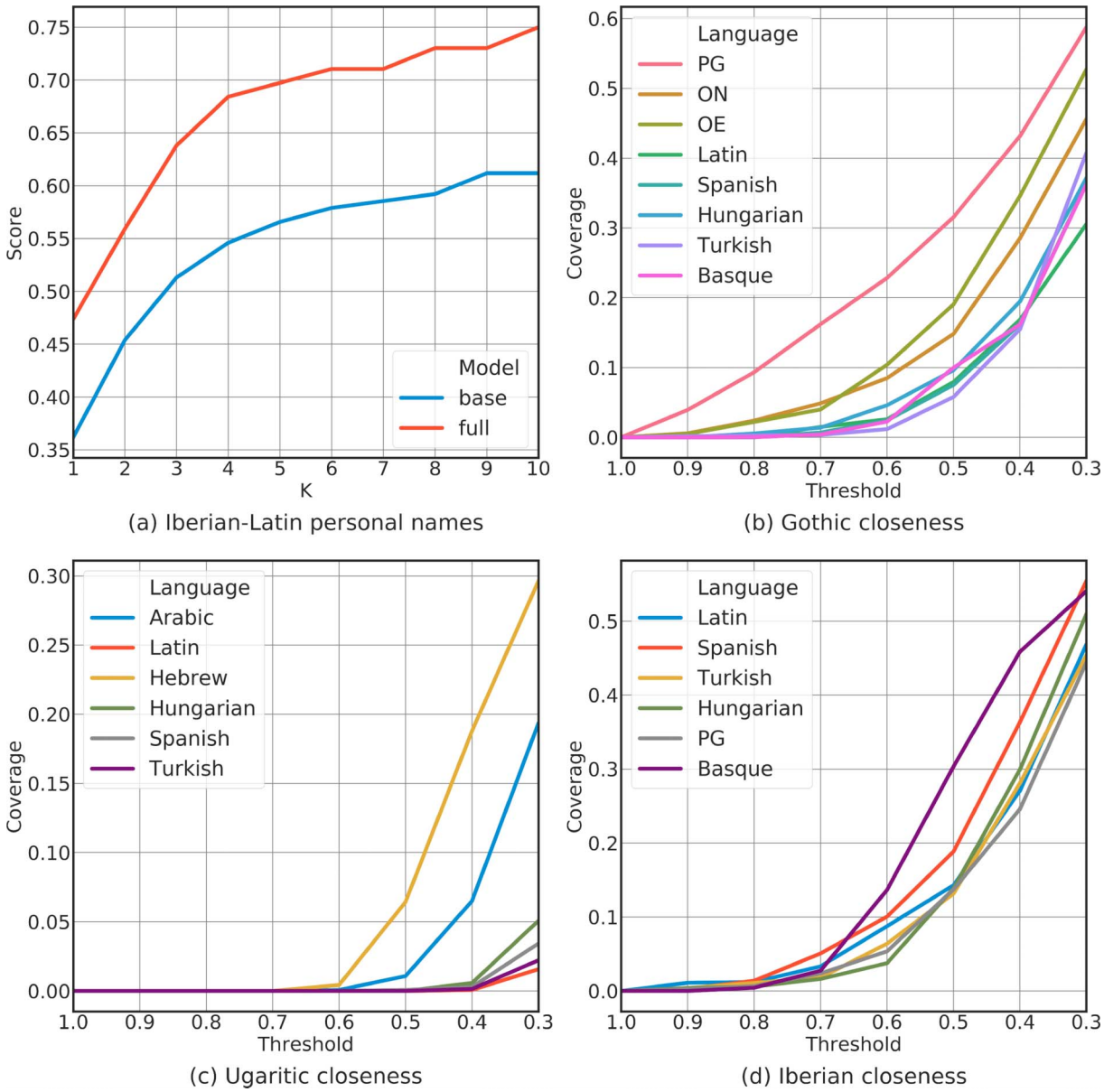

Figure 4: (a) A@K scores on Iberian using personal name recorded in Latin; (b), (c), and (d): Closeness plots for Gothic, Ugaritic and Iberian, respectively.

The choice of the known language also plays a significant role. On the closest language pair Gothic-PG, A@10 reaches 75% even without assuming any phonological knowledge about the lost language. As expected, language proximity directly impacts the complexity of the decipherment tasks which in turn translates into lower model performance on Old English and Old Norse. These results reaffirm that choosing a close known language is vital for decipherment.

The results on Iberian shows that our model performs well on a real undeciphered language with undersegmented texts. As shown in Figure 4a, `base` model reaches 60% in P@10 while `full` model reaches 75%. Note that Iberian is non-Indo-European with no genetic relationship with Latin, but our model can still discover regular correspondences for this particular set of personal names.

Ablation Study To investigate the contribution of phonetic and phonological knowledge, we conduct an ablation study using Gothic/Old Norse (Table 4). The IPA embeddings consistently improve all the model variants. As expected, the gains are most noticeable (+12.8%) for the hardest matching scenario where no prior information is available (`base` model). As expected, Ω_{loss} is vital

Lost / Known	Ugaritic† Hebrew	Gothic PG	Gothic ON	Gothic OE
Bayesian	0.604	–	–	–
NeuroCipher	0.659	0.753	0.543	0.313
base	**0.778**	**0.865**	**0.558**	**0.472**

† A@1 is reported for Ugaritic to make direct comparison with previous work. A@10 is still used for Gothic experiments.

Table 3: Results for comparing base model with previous work. Bayesian and NeuroCipher are the models proposed by Snyder et al. (2010) and Luo et al. (2019), respectively. Ugaritic results for previous work are taken from their papers. For NeuroCipher, we run the authors' public implementation to obtain the results for Gothic.

IPA	Ω_{loss}	base	partial	full
+	+	0.435	0.544	0.682
–	+	0.307	0.490	0.599
+	–	0.000	0.493	0.695

Table 4: Ablation study on the pair Gothic-ON. Both IPA embeddings and the regularization on sound loss are beneficial, especially when we do not assume much phonological knowledge about the lost language.

Inscription	Matched stem
ammuhsaminhaidau	xaiðˈ
ammuhsaminhaidau	xaiðˈ
ammuhsaminhaidau	raiðˈ
ammuhsaminhaidau	braiðˈ

Table 5: One example of top 3 model predictions for base on Gothic-PG in WR 0% setting. Spans are highlighted in the inscriptions. The first row presents the ground truth and the others are the model predictions. Green color is used for correct predictions and red for incorrect ones.

for base but unnecessary for full which has readily available character mapping.

Comparison with Previous Work To compare with the state-of-the-art decipherment models (Snyder et al., 2010; Luo et al., 2019), we consider the version of our model that operates with 100% whitespace ratio for the lost language. Table 3 demonstrates that our model consistently outperforms the baselines for both Ugaritic and Gothic. For instance, it reaches over 11% gain for Hebrew/Ugaritic pair and over 15% for Gotchic/Old English.

Identifying Close Known Languages Next we evaluate model's ability to identify a close known language to anchor the decipherment process. We expect that for a closer language pair, the predictions of the model will be more confident while matching more characters. We illustrate this idea with a plot that charts *character coverage* (i.e., what percentage of the lost texts are matched regardless of its correctness) as a function of *prediction confidence* value (i.e., probability of generating this span normalized by its length). As Figure 4b and Figure 4c illustrate, the model accurately predicts the closest languages for both Ugaritic and Gothic. Moreover, languages within the same family as the lost language stand out from the rest.

The picture is quite different for Iberian (see Figure 4d). No language seems to have a pronounced advantage over others. This seems to accord with the current scholarly understanding that Iberian is a language isolate, with no established kinship with others. Basque somewhat stands out from the rest, which might be attributed to its similar phonological system with Iberian (Sinner and Velaza, 2019) and very limited vocabulary overlap (numeral names) (Aznar, 2005) which doesn't carry over to the lexical system.[6]

6 Conclusions

We propose a decipherment model to extract cognates from undersegmented texts, without assuming proximity between lost and known languages. Linguistics properties are incorporated into the model design, such as phonetic plausibility of sound change and preservation of sounds. Our results on Gothic, Ugaritic, and Iberian shows that our model can effectively handle undersegmented texts even when source and target languages are not related. Additionally, we introduce a method for identifying close languages that correctly finds related languages for Gothic and Ugaritic. For Iberian, the method does not show strong evidence supporting Basque as a

[6]For true isolates, whether the predicted segmentations are reliable despite the lack of cognates is beyond our current scope of investigation.

related language, concurring with the favored position by current scholarship.

Potential applications of our method are not limited to decipherment. The phonetic values of lost characters can be inferred by mapping them to the known cognates. These values can serve as the starting point for lost sound reconstruction and more investigation is needed to establish their efficacy. Moreover, the effectiveness of incorporating phonological feature embeddings provides a path for future improvement for cognate detection in computational historical linguistics (Rama and List, 2019). Currently our method operates on a pair of languages. To simultaneously process multiple languages as it is common in the cognate detection task, more work is needed to modify our current model and its inference procedure.

Acknowledgments

We sincerely thank Noemí Moncunill Martí for her invaluable guidance on Iberian onomastics, and Eduardo Orduñ Aznar for his tremendous help on the Hesperia database and the Vasco-Iberian theories. Special thanks also go to Ignacio Fuentes and Carme Huertas for the insightful discussions. This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract # FA8650-17-C-9116. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Eduardo Orduña Aznar. 2005. Sobre algunos posibles numerales en textos ibéricos. *Palaeohispanica*, 5:491–506.

Taylor Berg-Kirkpatrick and Dan Klein. 2013. Decipherment with a million random restarts. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 874–878. Association for Computational Linguistics.

Steven Bird. 2006. NLTK: The natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72. Association for Computational Linguistics. **DOI:** https://doi.org/10.3115/1225403.1225421

Lyle Campbell. 2013. *Historical Linguistics*. Edinburgh University Press.

Christos Christodouloupoulos and Mark Steedman. 2015. A massively parallel corpus: The bible in 100 languages. *Language Resources and Evaluation*, 49(2):375–395. **DOI:** https://doi.org/10.1007/s10579-014-9287-y, **PMID:** 26321896, **PMCID:** PMC4551210

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7059–7069.

Bradley Hauer, Ryan Hayward, and Grzegorz Kondrak. 2014. Solving substitution ciphers with combined language models. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2314–2325, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Nishant Kambhatla, Anahita Mansouri Bigvand, and Anoop Sarkar. 2018. Decipherment of substitution ciphers with neural language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 869–874, Brussels, Belgium. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/D18-1102

Kevin Knight, Anish Nair, Nishit Rathod, and Kenji Yamada. 2006. Unsupervised analysis for decipherment problems. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 499–506, Sydney, Australia. Association for Computational Linguistics. **DOI:** https://doi.org/10.3115/1273073.1273138

Kevin Knight and Kenji Yamada. 1999. A computational approach to deciphering unknown scripts. *Unsupervised Learning in Natural Language Processing*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised machine translation using monolingual corpora only.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018b. Word translation without parallel data. In *International Conference on Learning Representations*.

Jiaming Luo, Yuan Cao, and Regina Barzilay. 2019. Neural decipherment via minimum-cost flow: From ugaritic to linear b. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3146–3155, Florence, Italy. Association for Computational Linguistics.

Noemí Moncunill Martí and others. 2017. Indigenous naming practices in the western Mediterranean: The case of Iberian. *Studia Antiqua et Archaeologica*, 23(1):7–20.

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)*, Paris, France. European Language Resources Association (ELRA).

Malte Nuhn, Julian Schamper, and Hermann Ney. 2013. Beam search for solving substitution ciphers. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1568–1576, Sofia, Bulgaria. Association for Computational Linguistics.

Taraka Rama and Johann-Mattis List. 2019. An automated framework for fast cognate detection and bayesian phylogenetic inference in computational historical linguistics. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6225–6235, Florence, Italy. Association for Computational Linguistics. **DOI:** https://doi.org/10.18653/v1/P19-1627

Jesús Rodríguez Ramos. 2014. Nuevo índice crítico de formantes de compuestos de tipo onomástico íberos. *Arqueoweb: Revista sobre Arqueología en Internet*, 15(1):7–158.

Donald A. Ringe. 2017. *From Proto-Indo-European to Proto-Germanic*. Oxford.

Alejandro Garcia Sinner and Javier Velaza. 2019. *Palaeohispanic Languages and Epigraphies*. Oxford University Press.

Benjamin Snyder, Regina Barzilay, and Kevin Knight. 2010. A statistical model for lost language decipherment. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1048–1057, Uppsala, Sweden. Association for Computational Linguistics. **DOI:** https://doi.org/10.1093/oso/9780198792581.001.0001

Kenneth N. Stevens. 2000. *Acoustic phonetics*, volume 30. MIT Press.

Larry Trask. 2008. *Etymological Dictionary of Basque*. http://www.bulgari-istoria-2010.com/Rechnici/baski-rechnik.pdf.

Norbert Wagner. 2006. Zu got. hv, q und ai, au. *Historische Sprachforschung/Historical Linguistics*, pages 286–291.

Julius Zacher. 1855. *Das Gothische Alphabet Vulfilas und das Runen Alphabet: eine Sprachwissenschaftliche Untersuchung*, FA Brockhaus.

A Appendices

A.1 Derivations for Dynamic Programming

We show the derivation for $\Pr(X)$ here—other quantities can be derived in a similar fashion.

Given any X with length n, let $p_i(X)$ be the probability of generating the prefix subsequence $X_{:i}$, and $p_{i,z}(X)$ be the probability of generating $X_{:i}$ using z as the *last* latent variable. By definition, we have

$$\Pr(X) = p_n(X). \tag{11}$$

$$p_i(X) = \sum_z p_{i,z}(X). \tag{12}$$

$p_{i,z}$ can be recursively computed using the following equations:

$$p_{i,\mathbf{O}} = \Pr(\mathbf{O}) \cdot p_0 \cdot p_{i-1}. \tag{13}$$

$$p_{i,\mathbf{E_l}} = \Pr(\mathbf{E_l}) \cdot \Pr(x_{i-l+1:l}|\mathbf{E_l}) \cdot p_{i-l}. \tag{14}$$

A.2 Data Preparation

Stemming Gothic stemmers are developed based on the documentations of `Gomorphv2`.[7] Stemmers for Proto-Germanic, Old Norse and Old English are derived from relevant Wikipedia entries on their grammar and phonology. For all other languages, we use the Snowball stemmer from NLTK (Bird, 2006)

IPA Transcription We use the `CLTK` library[8] for Old Norse and Old English, and a rule-based converter for Proto-Germanic based on (Ringe, 2017, pp. 242–260). Basque transcriber is based on its Wikipedia guide for transcription, and all other languages are transcribed using Epitran (Mortensen et al., 2018). The `ipapy` library[9] is used to obtain their phonetic features. There are 7 feature groups in total.

Known vocabulary For Proto-Germanic, Old Norse, and Old English, we extract the information from the descendant trees in Wiktionary.[10] All matched stems with at least four characters form the known vocabulary. It resulted in 7883, 10,754 and 11,067 matches with Gothic inscriptions, and 613, 529, 627 unique words in the vocabularies for Proto-Germanic, Old Norse, and Old English, respectively. For Ugaritic-Hebrew, we retain stems with at least three characters due to its shorter average stem length. For the Iberian-Latin personal name experiments, we take the list provided by Ramos (2014) and select the elements that have both Latin and Iberian correspondences. We obtain 64 unique Latin stems in total. For Basque, we use a Basque etymological dictionary (Trask, 2008), and extract Basque words of unknown origins to have a better chance to match Iberian tokens.

For all other known languages used for the closeness experiments, we use the Book of Genesis in these languages compiled by Christodouloupoulos and Steedman (2015) and take the most frequent stems. The number of stems is chosen to be roughly the same as the actual close relative, in order to remove any potential impact due to different vocabulary sizes. For instance, for the Gothic experiments in Figure 4b, this number is set to be 600 since the PG vocabulary has 613 words.

A.3 Training Details

Architecture For the majority of our experiments, we use a dimensionality of 100 for each feature embedding, making the character embedding of size 700 (there are 7 feature groups). For ablation study without IPA embeddings, each character is directly represented by a vector of size 700 instead. To compare with previous work, we use the default setting from `Neurocipher` which has a hidden size of 250, and therefore for our model we use a feature embedding size of 35, making it 245 for each character.

Hyperparameters We use SGD with a learning rate of 0.2 for all experiments. Dropout with a rate of 0.5 is applied after the embedding layer. The length for matched spans l in the range $[4, 10]$ for most experiments and $[3, 10]$ for Ugaritic. Other settings include $T = 0.2, \lambda_{\text{cov}} = 10.0, \lambda_{\text{cov}} = 100.0$. We experimented with two annealing schedules for the insertion penalty α: $\ln \alpha$ is annealed from 10.0 to 3.5 or from 0.0 to 3.5. These values are chosen based on our preliminary results, representing an extreme (10.0), a moderate (3.5), or a non-existent (0.0) penalty. Annealing last for 2000 steps, and the model is trained for an additional 1000 step afterwards. Five random runs are conducted for each setting and annealing schedule, and the best result is reported.

[7] `http://www.wulfila.be/gomorph/gothic/html/`.

[8] `http://cltk.org/`.

[9] `https://github.com/pettarin/ipapy`.

[10] `https://www.wiktionary.org/`.