Utilizing Word Embeddings based Features for Phylogenetic Tree Generation of Sanskrit Texts

Diptesh Kanojia[†,♣,⋆], Abhijeet Dubey[†], Malhar Kulkarni[†], Pushpak Bhattacharyya[†], Reza Haffari[⋆]
[†]IIT Bombay
[♣]IITB-Monash Research Academy
[⋆]Monash University
[†]{diptesh,abhijeet,pb}@cse.iitb.ac.in, [†]malhar@iitb.ac.in,
[⋆]reza.haffari@monash.edu

Abstract

Tracing the root of a text i.e., the original version of the text, by inferring phylogenetic trees has been a topic of interest in philological studies. However, existing methods face meaning conflation deficiency due to the usage of lexical similarity based measures which feed the distance matrix to clustering algorithms. In this paper, we utilize word embeddings as features to compute the distances among manuscripts. We conduct this pilot study on using word embeddings to compute inter-manuscript distances and provide an effective distance matrix to infer phylogenetic trees. We conduct experiments on the historical Sanskrit text known as Kāśikāvrtti (KV) and infer phylogenetic trees using this approach. For comparison, we also develop baseline methods using lexical distance-based measures to infer phylogenetic trees for KV. We show that our methodology produces better trees which club closely related manuscripts together compared to the baseline methods.

1 Introduction

Phylogenetics is defined as the task of creating a Phylogenetic Tree which represents a hypothesis about the evolutionary ancestry of a set of genes, species or any other taxa. It is the study of evolutionary history and relationships among various taxa. A Taxon represents a group of one or more manuscripts written in Sanskrit in our case, where we analyze how the manuscripts are related to each other. These relationships are discovered through phylogenetic methods that compute observed heritable traits in a manuscript, such as spelling errors, variations in text, text deletion, the morphology of the text etc. under a model of the evolution of these traits. The result of these analyses is a phylogeny (also known as a phylogenetic tree) – a diagrammatic hypothesis about the history of the evolutionary relationships of a group of manuscripts (usually belonging to the same text).

The computational purview of our problem deals with developing new methodologies for the estimation of the said trees. Computational historical linguistics, which involves the development of methods for estimating evolutionary histories of languages and, of models of language evolution, is another research problem based on phylogenetics. Phylogenetic methods are designed to recover the "true" evolutionary tree as often as possible. They do not guarantee to do so with high probability under reasonable conditions. Some which offer this guarantee vary considerably in their requirements (Warnow et al., 2001). To rigorously establish the validity of such a phylogenetic approach, a fundamental question that must be addressed is whether the models in use are identifiable. Parameters for simple models include the topology of the evolutionary tree, edge lengths on the tree, and rates of various types of substitution, though more complicated models have additional parameters as well. If a model is non-identifiable, one cannot show that performing inference with it will be statistically consistent. Informally, even with large amounts of data produced by an evolutionary process that was accurately described by the model, we might make erroneous inferences if we use a non-identifiable model. Under other models, many methods will be able to recover the tree if given long enough sequences.

The latter techniques are said to be statistically consistent under the model of evolution. Under some models of evolution, no method can be guaranteed to recover the true tree with high probability, due to unidentifiability.

Using the currently available models, finding optimal phylogenetic trees using compatibility criteria is, in its general case, NP-Complete (Warnow, 1993). Also, finding a maximum compatible tree is NP-Hard (Roch, 2006). As a consequence, this will mean that efficient algorithms to solve the problem, probably, can not exist. On the other hand, by restricting the kinds of input to the problem, we may be able to solve it efficiently. Our work restricts the input of data to a distance matrix which consists of distances between various manuscripts. We hypothesize inter-manuscript distance by using two methodologies and are able to construct phylogenetic trees based on both of them. Phylogenetic reconstruction and analysis is based on a data matrix where the rows represent the manuscripts to be studied, and the columns represent a linguistic feature or character (Nichols and Warnow, 2008). Moreover, the methods inspired from glottochronology take a boolean matrix as input, which denotes the change in the state of the 'characters' (the 'characters' can be lexical, morphological or phonological) to infer the phylogenetic trees. In our case, the distance matrix consists of manuscripts to be studied in both rows and columns, but the distances calculated are based on either character-based features (which is our baseline methodology) or word embeddings based distances which is our novel contribution to the area.

Our work is based on an earlier published sample edition of the KV on A 2.2.6 (Kulkarni, 2009). This edition was prepared using seventy manuscripts written in several scripts and collected from various parts of the world. This earlier work did not utilize the computational method to establish inter-relations between manuscripts. Kulkarni and Kahrs (2018) also published a manually drawn tree based on the edition mentioned above. In this work, we apply the computational methods on the same data mentioned above and automatically infer phylogenetic trees that show the inter-relations between manuscripts.

1.1 Motivation

Texts are important sources of intellectual history. In the Indian context, texts have travelled in the course of time both orally and written. Establishing a particular text using extant available resources enables us to have an authenticated base for the reconstruction of intellectual history. In order to create an authenticated base, we need to apply technological tools and methods. These will ensure objectivity and scientific explanation in the establishment of the text. Previous work on creating phylogenetic trees have not explored the usage of word embeddings which foray in the semantic space of linguistics. Word embeddings can provide a highly accurate representation of the context for a given word (Rong, 2014)

Rama and Singh (2009) use corpus-based measures to compute the distance matrix containing inter-language distances and construct phylogenetic trees for a linguistic area[1]. Corpus-based measures can calculate the inter-language distance, but they use feature n-grams and cognate identification methods which loosely take into account the semantics of a word. It is well known that word meaning can be represented with a range of senses/concepts. The methods above do not take into account the 'semantics' in a language and measure the inter-language distance only based on associated words pairs. Recently, an increasing boom on large-scale pre-trained word embedding models e.g., FastText (Bojanowski et al., 2017), ELMo (Peters et al., 2018), BERT (Devlin et al., 2018) have attracted considerable attention in the field of NLP. Inspired by the above works, this paper proposes to use word embeddings (Mikolov et al., 2013) created using fasttext approach (Conneau et al., 2017) to find the inter-manuscript distance based on functional units in a text.

[1]The term linguistic area or Sprachbund (Emeneau, 1956) refers to a group of languages that have become similar in some way as a result of proximity and language contact, even if they belong to different families. The best-known example is the Indian (or South Asian) linguistic area.

The question that we try to answer in this paper is:

"Can word embeddings with sub-word information help build more accurate phylogenetic trees from multiple versions of a manuscript ?"

2 Related Work

Computational phylogenetics has, in recent years, developed various methodologies under the purview of computational biology (Felsenstein and Felenstein, 2004; Huelsenbeck et al., 2001; Saitou and Nei, 1987; Swofford et al., 1996). The growth of phylogenetics as an area with significance to statistical methods is captured by Felsenstein (2001) in an article where he explains the developments of numerical methods for the creation of phylogenies. These methods have been widely adopted in computational linguistics for the construction of phylogenetic trees. A major disadvantage of using these character-based or lexical distance-based methods is the need for manually curated word lists. Csernel and Patte (2007) discuss the LCS algorithm for preparing a critical edition of Sanskrit texts and provide a method for comparison of Sanskrit manuscripts. Among the many available methods (Huelsenbeck, 1995) to construct phylogenetic trees, UPGMA (Gronau and Moran, 2007) is widely used in historical linguistics. It assumes a constant rate of evolution and is not a well-regarded method for inferring relationships unless this assumption has been tested and justified for the data set being used. The UPGMA method constructs phylogenetic trees based on a distance matrix which can be computed in various ways. Saitou and Nei (1987) proposed neighbour joining method to construct phylogenies based on sequence analysis, which uses genetic distance as a clustering metric. Moret et al. (2002) study the sequence lengths required by neighbour-joining, greedy parsimony, and a phylogenetic reconstruction method based on disk-covering and the maximum parsimony criterion and show improvements in large scale phylogenetic reconstruction. Symmetric cross-entropy is one of the methods which is purely a letter n-gram based measure similar to the one used by Singh (2006b) for language and encoding identification. Singh and Surana (2007) used corpus-based measures to show that corpus can be used for a comparative study of languages. They used both character n-gram distances and surface similarity (Singh, 2006a) to identify the potential cognates, which in turn are being used to estimate the inter-language distance. Rama and Singh (2009) also used measures based on cognate identification, and feature n-grams to infer this matrix. Ellison and Kirby (2006) discussed establishing a probability distribution for every language through intra-lexical comparison using confusion probabilities and estimate distances using KL divergence and Rao's distance (Atkinson and Mitchell, 1981). Automatic Cognate Detection (ACD) is an important task which can help phylogenetic reconstruction and complement current research on language phylogenies (Rama et al., 2018). Rama (2016) come up with siamese architectures that jointly learn phoneme level feature representations and language relatedness from raw words for cognate identification. Rama et al. (2017) explore the use of unsupervised methods for detecting cognates in multilingual word lists. They use online EM to train sound segment similarity weights for computing similarity between two words. Kanojia et al. (2019) utilize wordnets and identify cognates among Indian languages for improvement in the construction of the phylogenetic trees. They used lexical similarity based measures to find the similarity among Indian language word lists and induced the cognates in clustering methods to generate phylogenies. Kulkarni (2012) builds a phylogenetic tree for Malayalam manuscripts of the Kāśikāvrtti, and show that M is the archetype source and Ma, Mb and Mc are its hyperarche child nodes. M is decided as a source based on the analysis made on the manual reading of the manuscripts. In this process, manuscripts are grouped together and named as M1, M2, M3 …, M11. Kulkarni (2003) and Kulkarni (2008) build a similar tree for the Sharada manuscripts of the KV.

To the best of our knowledge, no one has utilized word embeddings to construct the distance matrix for inter-manuscript distances. We deploy lexical similarity-based methods as a baseline for inter-manuscript distance and compare the tree with the trees generated via our approach i.e., using word-embeddings to construct the distance matrix for the clustering methods (UPGMA and Neighbour Joining).

We contribute the following through this work:

- We hypothesize inter-manuscript distance and create efficient distance matrices for phylogenetic tree construction.
- We build baseline methodology using lexical similarity based measures for comparison with our approach and generate phylogenetic trees.
- We construct a distance matrix through a word embeddings based approach as a novel contribution and show that the trees generated are better than the baseline method.

3 Dataset and Experiment Setup

3.1 Dataset

We collect the following data for performing our experiments and tree construction.

3.1.1 KV Dataset

For distance matrix generation, we focus on specific portions of the KV. We collect seventy different versions of the KV on AST 2.2.6. We perform cleaning and manual analysis with the help of philologists. These versions were available in different parts of the country from where we accumulated them in a single repository. We observe different kinds of changes in these versions and describe them in Section 6.

3.1.2 Raw Corpus for obtaining Word embeddings

We obtain raw monolingual Sanskrit corpus from various sources. We download the Sanskrit Wikimedia dump and collate all the articles as a single corpus. We, also, add Glosses and Example sentences from the Sanskrit Wordnet to this corpus. We obtain raw corpus from other sources available online[2]. We perform cleaning for this corpus by removing any other ASCII characters apart from the Devanagari script. The final cleaned coprus used for creating embeddings contains 5,38,323 lines. Eventually, We use binarized vectors to compute the distance between two words.

3.2 Experimental Setup

The Neighbor Joining method and the UPGMA method are both distance-based methods as described in Section 4. They require a distance matrix which specifies the distance between the Taxa being used to populate the phylogeny. We also describe the methodologies used to obtain these matrices in Section 4. For our experiments, we divide the KV data into different functional units. The functional unit division in KV depends on the type of sutra. The sutra that we use for our experiments, namely AST 2.2.6, is of the type vidhi.

The functional unit division of this type is as follows:

- vidhi: This type of sutra prescribes either a verbal element or an operation. The KV on this sutra contains the following functional parts (Sutra AST 2.2.6):
    1. The sentence explaining the meaning of the words in the sutra.
    2. Examples

[2]Available on the School of Sanskrit and Indic Studies, J.N.U. and NLP for Sanskrit from GitHub

These functional units help us understand the text in a better manner, and for computational purposes, they create separate divisions in the text so that the versions are compared to each other in an efficient manner. We compare each functional unit only with its counterpart from the versions. For e.g., In AST 2.2.6 dataset, we compare the examples from one version only with the examples of the other version.

For training the word embeddings based model, we use Gensim[3]. We choose FastText (Bojanowski et al., 2017) for training the word embeddings and obtaining vectors as it utilizes subword-level information within the text. Sanskrit is an agglutinative language which is also highly morphological. To capture the morphology and semantics within each word, we also need to take into account the sub-word level information. We train the models with the following hyperparameters. We create these models based on 100 and 50 dimensions due to a limited amount of the corpus collected[4]. The rest of the parameters were the same for both the models. We restrict the context window to 5 and use 0.1 as the learning rate. The maximum length of word n-gram we use is one word. We retain the sampling threshold at a default 0.0001. We use softmax as the loss function and train the models for five epochs[5].

4 Methodology

In this section, we describe the various methodologies used for calculating the inter-manuscript distances and tree construction.

4.1 Computing the Inter-Manuscript Distances

We use two approaches for constructing the inter-manuscript distances. The baseline approach utilizes various lexical similarity based measures and later, we also provide weights to them, using empirical approaches, to increase their efficiency. In our approach, we use word-embedding based models and compute distances using vectors obtained from them. Since angular cosine distance distinguishes nearly parallel vectors better (Cer et al., 2018), we also include this in our approach, apart from cosine distance to generate more trees and discuss the outcome in Section 5.

4.1.1 Lexical Distance based measures: A Baseline Approach

We use the following lexical similarity based measures to compute the distances among manuscripts:

Normalized Edit Distance Method (NED)

The Normalized Edit Distance (also known as Levenshtein Distance) approach computes the edit distance (Nerbonne and Heeringa, 1997) for all word pairs in a functional unit of the text and then provides as output the average distance between all word pairs (we term it as 'Unit Distance'). In each of the operations has unit cost (except that substitution of a character by itself has zero cost), so NED is equal to the minimum number of operations required to transform 'word a' to 'word b'. A more general definition associates non-negative weight functions (insertions, deletions, and substitutions) with the operations.

Cosine Distance (CoD)

The cosine similarity measure (Salton and Buckley, 1988) is another similarity metric that depends on envisioning preferences as points in space. It measures the cosine of the angle between two vectors projected in a multi-dimensional space. The cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1]. The name derives from the term

[3]Gensim Source

[4]The standard number of dimensions for word embeddings, given a big corpus, is 300

[5]More epochs usually lead to a better learned/trained model; we retain the best epoch output with a minimum loss to be utilized for our work

"direction cosine": in this case, unit vectors are maximally "similar" if they're parallel and maximally "dissimilar" if they're orthogonal (perpendicular). This is analogous to the cosine, which is 1 (maximum value) when the segments subtend a zero angle and 0 (uncorrelated) when the segments are perpendicular. In this context, the two vectors are the arrays of character counts of two words. We calculate the cosine distance as (1 - Cosine Similarity).

Jaro-Winkler Distance (JWD)

Jaro-Winkler distance (Winkler, 1990) is a string metric measuring similar to the normalized edit distance deriving itself from Jaro Distance (Jaro, 1989). Here, the edit distance between two sequences is calculated using a prefix scale P which gives more favourable ratings to strings that match from the beginning, for a set prefix length L. The lower the Jaro–Winkler distance for two strings is, the more similar the strings are. The score is normalized such that 1 equates to no similarity and 0 is an exact match.

Distance Matrix Computation

The above similarity metrics use different ways to compute the distance between each word pair and hence, produce varying distance matrices. We compute the distance between a sutra by averaging over each 'Unit Distance' present in a sutra. We computer these distances between all the manuscript pairs. Thus, we generate three inter-manuscript distance matrices based on the methods described above.

Since all the matrices above use different ways to compute distances, we performed another set of experiments for coming up with a more homogenous approach. For computational purposes, we provide all the metrics equal weightages initially, and compute a single the distance matrix using the average score of all three methods. So, for manuscripts p and q, the average inter-manuscript distance is defined as:

$$LD_{pq} = \frac{(NED_{pq} + CoD_{pq} + JWD_{pq})}{3} \tag{1}$$

We, also, experiment over weightages and later provide different weightages to each method. Empirically, we find best results by setting the weight of NED to 50%, CoD to 25%, and JWD to 25%. For masnucripts p and q, the weighted average inter-manuscript distance is defined as:

$$LD_{pq} = (NED_{pq} * 0.5) + (CoD_{pq} * 0.25) + (JWD_{pq} * 0.25) \tag{2}$$

Using the baseline methodology, we create a total of 5 matrices for the text in the AST 2.2.6 dataset.

4.1.2 Word embeddings based distance measures: Our Approach

We calculate the cosine distance between all word pairs belonging to the same functional unit from the embedding space. Thus, the average over the word pair distances gives us 'Unit Distance'. Similar to the baseline method, we average over all unit distances to find out the inter-manuscript distance for each manuscript pair and compute the distance matrix. Since angular cosine distance distinguishes nearly parallel vectors better (Cer et al., 2018), we also use angular cosine distance and calculate the inter-manuscript distance for each manuscript pair, in a similar fashion. We perform this experiment using two different models described in the experimental setup.

Thus, for each dataset, our approach generates four matrices i.e., a matrix which utilizes Cosine Distance from the model with 100 dimensions, another which utilizes Cosine Distance

from the model with 50 dimensions and another pair of matrices with Angular Cosine Distance from the models with 100 and 50 dimensions each. Using this approach, we create a total of four matrices.

Using all of the methodologies described above (both baseline and our approach), we create a total of 9 matrices for the text in AST 2.2.6 dataset.

4.2 Tree generation using distance-based clustering methods

We choose two distance-based methods for our work, namely, the Neighbor Joining method and the UPGMA method. We further describe these methods below, along with the reasons for choosing these methods.

4.2.1 Distance-based Methods

Distance analysis compares two aligned manuscripts at a time and builds a matrix of all possible sequence pairs. During each comparison, the number of changes (base substitutions and insertion/deletion events) is counted and presented as a proportion of the overall sequence length. These final estimates of the difference between all possible pairs of manuscripts are known as pairwise distances. A variety of distance algorithms are available to calculate the pairwise distance (between versions), for example, Proportional (p) distances. We use the baseline approach and our approach to compute these pairwise distances. Once the pairwise distances are calculated, they must be arranged into a tree. There are many ways to "arrange" the Taxa according to their distances. One way to cluster or optimize the distances is to join Taxa together according to their increasing differences, as embodied by their distances. Other ways use various coefficients to measure how well the branch lengths of the tree reflects the original pairwise distances.

Distance-matrix methods of phylogenetic analysis explicitly rely on a measure of "genetic distance" between the manuscripts being classified, and therefore they require an MSA (multiple sequence alignment) as an input. Distance is often defined as the fraction of mismatches at aligned positions, with gaps either ignored or counted as mismatches (David, 2001). The main disadvantage of distance-matrix methods is their inability to efficiently use information about local high-variation regions that appear across multiple subtrees (Felsenstein and Felenstein, 2004). Distance methods attempt to construct an all-to-all matrix from the sequence query set describing the distance between each sequence pair. From this is constructed a phylogenetic tree that places closely related manuscripts under the same interior node and whose branch lengths closely reproduce the observed distances between manuscripts. Distance-matrix methods may produce either rooted or unrooted trees, depending on the algorithm used to calculate them. They are frequently used as the basis for progressive and iterative types of multiple sequence alignment. The distance-based methods which we use are:

UPGMA Method

The Unweighted Pair Group Method with Arithmetic mean (UPGMA) method (Sokal and Rohlf, 1962) produces rooted trees and requires a constant-rate assumption, i.e. they assume an ultrametric tree in which the distances from the root to every branch tip are equal. At each step, the nearest two clusters are combined into a higher-level cluster. The distance between any two clusters A and B, each of size (i.e., cardinality) |A| and |B|, is taken to be the average of all distances D(x,y) between pairs of objects x in A and y in B, that is, the mean distance between elements of each cluster. In other words, at each clustering step, the updated distance between the joined clusters and a new cluster X is given by the proportional averaging of the distance between A given X and the distance between B given X.

We use the UPGMA method to construct phylogenetic trees for all the manuscript pairs. The input to the UPGMA method is the distance matrix created via the methodologies described above. We use the implementation of UPGMA provided by PHYLIP (Felsenstein, 1993) and

generate baseline trees for NED, CoD, JWD, Average, and Weighted Average distance matrices. We also generate trees for distance matrices obtained using our approach of cosine distances and angular cosine distances from word embeddings space.

Neighbor Joining Method

Neighbour-Joining (Saitou and Nei, 1987) is a bottom-up (agglomerative) clustering method for the creation of phylogenetic trees. It applies general data clustering techniques to sequence analysis and uses genetic distance as a clustering metric. The simple version of the neighbour-joining method produces unrooted trees, but it does not assume a constant rate of evolution (i.e., a constant timeline) across lineages. Neighbour-joining may be viewed as a greedy algorithm for optimizing according to the 'balanced minimum evolution' (BME) criterion. For each topology, the tree length (sum of branch lengths) is a particular weighted sum of the distances in the distance matrix, with the weights depending on the topology. The optimal topology (as per BME) is the one which minimizes this length. At each step, it greedily joins the pair of taxa which provides the greatest decrease in the estimated tree length. This procedure is not guaranteed to find the topology which is optimal by the BME criterion, although it often does and is usually quite close.

Similarly, we use the neighbour-joining method to construct phylogenetic trees for all the manuscript pairs. The input to this method is also the distance matrix created via the methodologies described above. We use the implementation of neighbour-joining provided by PHYLIP (Felsenstein, 1993) and generate all the trees from the matrices described above.

5 Results

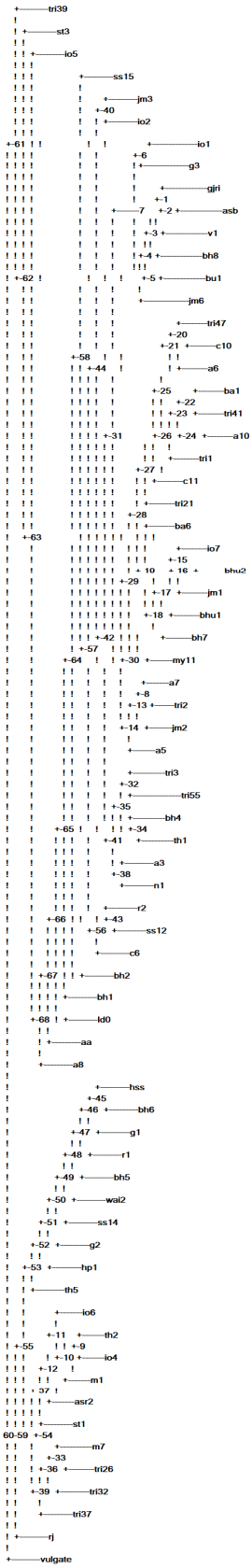
We generate trees using both the neighbour-joining and the UPGMA methods for all the matrices described above and compare them with the trees manually created by our philologists. The basis of this evaluation was the expert knowledge of our philologists who have studied the KV and are aware of the origin, groupings, and a vague timeline of all these manuscript versions. Their findings indicate that the trees generated via our approach of using word embeddings were closest to the manually created trees and required a few corrections among the subgroupings to be accurate. Although, among the baseline approaches, the weighted average methodology also reached the closer to the manually created phylogenetic tree, but it was still a few corrections behind. We can not present the complete set of 18 trees (9 x UPGMA and 9 x Neighbour Joining) here hence show the best tree generated by the baseline method in Figure 1a for the text in 2.2.6 dataset. We obtain this tree using our novel approach of using word-embeddings based model and using Neighbour-joining as the tree generation methodology. In Figure 1b for the text in 2.2.6 dataset, we also show the tree obtained by the weighted average lexical similarity measure, which was also generated using the Neighbour-joining method.

Among the word embeddings based approach, the trees generated via cosine distance are reported to be more accurate than the trees generated via angular cosine distance, as per our philologists.

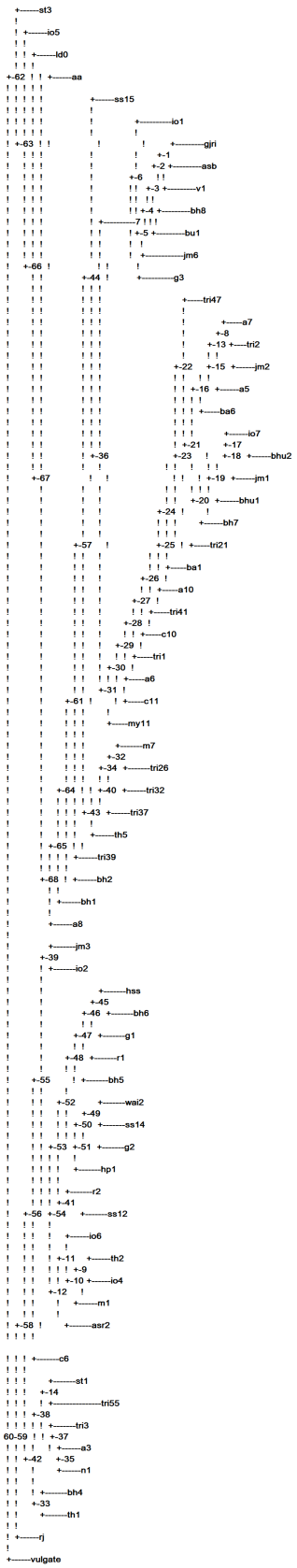
We compared the matrices generated by both cosine distance and angular cosine distance and found out that the distance values did not have a lot of difference. This is probably due to the lack of a large raw monolingual corpus for creation of word embeddings for Sanskrit. Despite being one of the most ancient languages, the availability of the resource for Sanskrit is scarce, which motivates us further to keep exploring this area. We discuss the results of our work and the merits of our methodology in the next section. We also provide justifications of our philologists' view in the forthcoming section.

6 Discussion

We discuss the functional units of the AST 2.2.6 dataset in the section above in brief and generate results based on the comparison of each unit. The division of KV data for the AST 2.2.6 text is



(a) Tree Generated using Neighbour-joining method. Distance matrix computed using the word-embeddings based method



(b) Tree Generated using Neighbour-joining method. Distance matrix computed using the lexical similarity-based method (See Equation 2)

shown in Table 1.

2.2.6.	नञ्
2.2.6.1	नञ् समर्थेन सुबन्तेन सह समस्यते तत्पुरुषश्च समासो भवति।
2.2.6.2	न ब्राह्मणो अब्राह्मणः। अवृषलः।।

Table 1: Example of Functional Unit based Division for sūtra AST 2.2.6

As can be seen in Figure 1a above, the sub-groupings for manuscripts has been done more accurately. Manuscripts io1, g3, gjri, asb, v1, bh8, bu1 and jm6 have been grouped together since they do not contain a common functional unit. The same can be said about the tree in Figure 1b but it does not group bh1 and ld0 in the same sub-group which should not have been the case.

Differences among the manuscript variants in this edition (Kulkarni, 2009) are mainly divided into four categories. The apparatus of this edition contains the mention of the following types of variants:

Omission (Om.): absence of a word.

Addition (Add.): presence of an additional word

Change of word (CW): lexical changes in the word due to morphological inflection, or due to the opinion of the scribe who created the manuscript variant.

Change in the place of a word (CPW): change in the positioning of a word among the functional unit in a text.

We develop both the baseline approach and word embeddings based approach keeping these variants in mind. Our approaches handle these variants in the following manner:

Ommission (Om.)

Ommission reflects the omitted portion of the text derived after comparing the critical edition with the manuscripts of the text. Our approaches calculate the distances between all word pairs of each functional unit, on both sides. When we perform the comparison between an omitted word on one side and do not find its counterpart on the other side, it results in a higher penalty and a greater distance like it should for an omitted word.

Addition (Add.)

Addition refers to the added portion of the text as available in the manuscripts. It can be one or more words depending on the variant. When we average of all the distances between all word pairs, and in the comparisons made, do not find the added words; it results in a high penalty a greater overall distance like it should for an added portion.

Change of word (CW)

CW refers to a change of word, in the manuscript, in comparison with the critical edition i.e., a word may undergo some morphological inflection or takes some other form but retains a semantic notion. In such a case, the baseline approach measures the lexical changes in a word but penalise this change relatively lower in magnitude. In our approach, since the semantic notion is maintained, the embeddings would provide with nearby vectors and thus also penalise relatively lower in magnitude, which is what should be done for such a variance.

Change in the place of a word (CPW)

CPW refers to the change in the place of the word in the manuscript in comparison with the critical edition. CPW implies that the words in question exist in the manuscript but changes its place. This is not the case with the previous three types of changes. Our methodology counters this variance when we average over all the word pairs. Since the word is indeed present in the functional unit of the text, we should be able to find its occurrence on the other side, and thus this would result in a penalty of lower magnitude in terms of distance. We discuss these approaches with our philologists and their views are in accordance with what our methodology does in penalising computing distances.

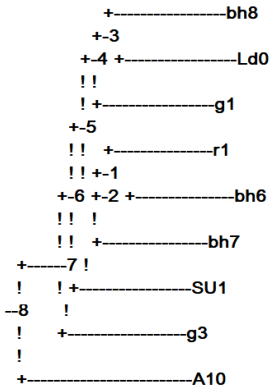


Figure 2: Phylogenetic Tree for the dated manuscripts generated using our method

Availibility of the timeline

Ancient Sanskrit text and its manuscripts are scarcely found dated. The unavailability of a timeline (or a temporal reference of versions) of how these texts evolved is a primary reason phylogenetic methods are needed to derive the root version (or the critical edition). We also note that some manuscripts among all the versions are dated, which do help identify the accuracy of a generated tree. Among the seventy versions of KV, we currently have the temporal references for eleven versions. We also generate phylogenetic trees for these versions using the neighbour-joining method based on the distance matrix computed using the word embeddings based approach they provided us with the best trees for AST 2.2.6. We depict this tree in Figure 2. In this tree, we have not yet implemented a method to refer to the timeline which is available. We plan to refine and generate such sub-trees based on the temporal references available to implement more accurate sub-trees of this type.

7 Conclusion and Future Work

In this paper, we presented a novel word embeddings based approach to create inter-manuscript distances and hypothesize functional units as a part of the text. We devised a baseline approach for drawing a comparison from our approach, which is based on lexical distance-based measures. We collect manuscript versions from different sources and accumulate them in a single repository and compute the inter-manuscript distance between all manuscript pairs, thus formulating a distance matrix for each approach. We collect raw Sanskrit corpus from various sources and create a word embeddings model using the state-of-the-art library. We release this word embeddings model publicly for the use of other researchers looking to explore this area. Also, we compute inter-manuscript distances using this model and generate trees for both using both the baseline and this approach. We compare the trees manually, evaluate them with the help of expert philologists where we go on to show that the trees generated via word embeddings based models were better in subgrouping and required the least number of corrections to reach the state of manually drawn trees. We discuss the merits of our approach

with examples and provide justifications of our results. Our approach clearly outperforms the baseline method and thus should help the researchers in this area to create better, more accurate phylogenetic trees in the near future.

In future, we would like to extend our dataset of the KV text to complete all the containing sutras and perform the same experiments for all such portions of the KV text. We plan to divide each of such portions of text into functional units and perform the same experiment for the text. We aim to include the other material like text commentaries and earlier texts as a part of the experiment in the future, as they provide important references to the text.

References

[Atkinson and Mitchell1981] Colin Atkinson and Ann FS Mitchell. 1981. Rao's distance measure. Sankhyā: The Indian Journal of Statistics, Series A, pages 345–365.

[Bojanowski et al.2017] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5:135–146.

[Cer et al.2018] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. arXiv preprint arXiv:1803.11175.

[Conneau et al.2017] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. arXiv preprint arXiv:1710.04087.

[Csernel and Patte2007] Marc Csernel and François Patte. 2007. Critical edition of sanskrit texts. In Sanskrit Computational Linguistics, pages 358–379. Springer.

[David2001] W Mount David. 2001. Bioinformatics: sequence and genome analysis. Bioinformatics, 28.

[Devlin et al.2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[Ellison and Kirby2006] T Mark Ellison and Simon Kirby. 2006. Measuring language divergence by intra-lexical comparison. In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, pages 273–280. Association for Computational Linguistics.

[Emeneau1956] Murray B Emeneau. 1956. India as a lingustic area. Language, 32(1):3–16.

[Felsenstein and Felenstein2004] Joseph Felsenstein and Joseph Felenstein. 2004. Inferring phylogenies, volume 2. Sinauer associates Sunderland, MA.

[Felsenstein1993] Joseph Felsenstein. 1993. PHYLIP (phylogeny inference package), version 3.5 c. Joseph Felsenstein.

[Felsenstein2001] Joseph Felsenstein. 2001. The troubled growth of statistical phylogenetics. Systematic Biology, pages 465–467.

[Gronau and Moran2007] Ilan Gronau and Shlomo Moran. 2007. Optimal implementations of upgma and other common clustering algorithms. Information Processing Letters, 104(6):205–210.

[Huelsenbeck et al.2001] John P Huelsenbeck, Fredrik Ronquist, Rasmus Nielsen, and Jonathan P Bollback. 2001. Bayesian inference of phylogeny and its impact on evolutionary biology. science, 294(5550):2310–2314.

[Huelsenbeck1995] John P Huelsenbeck. 1995. Performance of phylogenetic methods in simulation. Systematic biology, 44(1):17–48.

[Jaro1989] Matthew A Jaro. 1989. Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. Journal of the American Statistical Association, 84(406):414–420.

[Kanojia et al.2019] Diptesh Kanojia, Malhar Kulkarni, Pushpak Bhattacharyya, and Gholemreza Haffari. 2019. Cognate identification to improve phylogenetic trees for indian languages. In Proceedings of the ACM India Joint International Conference on Data Science and Management of Data, pages 297–300. ACM.

[Kulkarni and Kahrs2018] Malhar Kulkarni and Eivind Kahrs. 2018. Materials for the critical edition of kāśikāvrtti 1.1.

[Kulkarni2003] Malhar Kulkarni. 2003. The sharada manuscripts of the kāśikāvrtti. pages 353–364.

[Kulkarni2008] Malhar Kulkarni. 2008. The sharada manuscripts of the kāśikāvrtti: Part ii. pages 419–428.

[Kulkarni2009] Malhar Kulkarni. 2009. A sample of the new edition of the kāśikāvrtti: 2.2.6. LXV:116–127.

[Kulkarni2012] Malhar Kulkarni. 2012. The malayalam manuscripts of the kāśikāvrtti: A study. 6:103–112.

[Mikolov et al.2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems, pages 3111–3119.

[Moret et al.2002] Bernard ME Moret, Usman Roshan, and Tandy Warnow. 2002. Sequence-length requirements for phylogenetic methods. In International Workshop on Algorithms in Bioinformatics, pages 343–356. Springer.

[Nerbonne and Heeringa1997] John Nerbonne and Wilbert Heeringa. 1997. Measuring dialect distance phonetically. In Computational Phonology: Third Meeting of the ACL Special Interest Group in Computational Phonology.

[Nichols and Warnow2008] Johanna Nichols and Tandy Warnow. 2008. Tutorial on computational linguistic phylogeny. Language and Linguistics Compass, 2(5):760–820.

[Peters et al.2018] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

[Rama and Singh2009] Taraka Rama and Anil Kumar Singh. 2009. From bag of languages to family trees from noisy corpus. In Proceedings of the International Conference RANLP-2009, pages 355–359.

[Rama et al.2017] Taraka Rama, Johannes Wahle, Pavel Sofroniev, and Gerhard Jäger. 2017. Fast and unsupervised methods for multilingual cognate clustering. arXiv preprint arXiv:1702.04938.

[Rama et al.2018] 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.

[Rama2016] Taraka Rama. 2016. Siamese convolutional networks for cognate identification. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1018–1027.

[Roch2006] Sebastien Roch. 2006. A short proof that phylogenetic tree reconstruction by maximum likelihood is hard. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 3(1):92–94.

[Rong2014] Xin Rong. 2014. word2vec parameter learning explained. arXiv preprint arXiv:1411.2738.

[Saitou and Nei1987] Naruya Saitou and Masatoshi Nei. 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Molecular biology and evolution, 4(4):406–425.

[Salton and Buckley1988] Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. Information processing & management, 24(5):513–523.

[Singh and Surana2007] Anil Kumar Singh and Harshit Surana. 2007. Can corpus based measures be used for comparative study of languages? In Proceedings of ninth meeting of the ACL special interest group in computational morphology and phonology, pages 40–47. Association for Computational Linguistics.

[Singh2006a] Anil Kumar Singh. 2006a. A computational phonetic model for indian language scripts. In Constraints on Spelling Changes: Fifth International Workshop on Writing Systems. Nijmegen, The Netherlands.

[Singh2006b] Anil Kumar Singh. 2006b. Study of some distance measures for language and encoding identification. In Proceedings of the Workshop on Linguistic Distances, pages 63–72. Association for Computational Linguistics.

[Sokal and Rohlf1962] Robert R Sokal and F James Rohlf. 1962. The comparison of dendrograms by objective methods. Taxon, pages 33–40.

[Swofford et al.1996] DL Swofford, GJ Olsen, PJ Waddell, and DM Hillis. 1996. Phylogenetic inference, p. 407–514. Molecular Systematics (second edition). Sinauer Associates, Sunderland, Massachusetts.

[Warnow et al.2001] Tandy Warnow, Bernard ME Moret, and Katherine St John. 2001. Absolute convergence: true trees from short sequences. In Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms, pages 186–195. Society for Industrial and Applied Mathematics.

[Warnow1993] Tandy J Warnow. 1993. Constructing phylogenetic trees efficiently using compatibility criteria. New Zealand Journal of Botany, 31(3):239–247.

[Winkler1990] William E Winkler. 1990. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage.