Mapping Hymns and Organizing Concepts in the Rigveda: Quantitatively Connecting the Vedic Suktas

Venkatesh Bollineni, Igor Crk, Eren Gultepe*
Dept. of Computer Science, Southern Illinois University Edwardsville, USA
Correspondence: *egultep@siue.edu

Abstract

Accessing and gaining insight into the Rigveda poses a non-trivial challenge due to its extremely ancient Sanskrit language, poetic structure, and large volume of text. By using NLP techniques, this study identified topics and semantic connections of hymns within the Rigveda that were corroborated by seven well-known groupings of hymns. The 1,028 suktas (hymns) from the modern English translation of the Rigveda by Jamison and Brereton were preprocessed and sukta-level embeddings were obtained using, i) a novel adaptation of LSA, presented herein, ii) SBERT, and iii) Doc2Vec embeddings. Following an UMAP dimension reduction of the vectors, the network of suktas was formed using k-nearest neighbours. Then, community detection of topics in the sukta networks was performed with the Louvain, Leiden, and label propagation methods, whose statistical significance of the formed topics were determined using an appropriate null distribution. Only the novel adaptation of LSA using the Leiden method, had detected sukta topic networks that were significant (z = 2.726, p < .01) with a modularity score of 0.944. Of the seven famous sukta groupings analyzed (e.g., creation, funeral, water, etc.) the LSA derived network was successful in all seven cases, while Doc2Vec was not significant and failed to detect the relevant suktas. SBERT detected four of the famous suktas as separate groups, but mistakenly combined three of them into a single mixed group. Also, the SBERT network was not statistically significant.

1 Background and Significance

The Rigveda is written in Vedic Sanskrit and is the oldest existing sample of Sanskrit literature, written approximately 3000 years ago, in the region of present-day Afghanistan and the Punjab region of India (Jamison and Brereton, 2014). It is a heterogeneous collection of hymns (suktas) written by various poets (Rishis), that praise gods, describe rituals, and provide wisdom (Jamison and Brereton, 2014; Tiwari, 2021). Popular mantras recited by Hindus, such as the Gayatri mantra, is chanted at three different times of the day (Smith, 2019) for the purposes of mental well-being, and the Mahamrityunjaya mantra, which is recited for physical protection and longevity, are both sourced from the Rigveda (Devananda and Devananda, 1999).

Yet despite being a central text in Hinduism, navigating the Rigveda and obtaining insights regarding concepts and topics are not as straightforward as the Bible or Quran, for which there are innumerable resources (such as commentaries) and written for individuals at varying levels of skill and familiarity with the books. This is especially true for individuals who do not speak or understand any of the Indian languages. Although scholarly articles regarding specific topics (such as death) in the Rigveda are available, for the layperson interested in learning about the Rigveda, organizing and collating the information may be unwieldy (Jamison and Brereton, 2014). This is further evidenced in NLP studies, where the quantity of studies focused on the Abrahamic religions vastly outnumbers those focused on Hindu religious texts (Hutchinson, 2024).

2 Related Work

Recent studies have analyzed Hindu religious and literary texts from various aspects. One study extracted and formed social networks among the Pandavas (protagonists) and Kuaravas (antagonists) in the Mahabharata (an epic poem from the Hindu scriptures) using matrix factorization and spectral graph theory techniques (Gultepe and Mathangi, 2023). In another study, using linguistic and lexical features in Sanskrit, the Mahabharata was stratified into clusters (Hellwig, 2017). Another study had determined topics on the English translations of two other important Hindu texts (Chandra and

Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities, pages 514–523
May 3-4, 2025 ©2025 Association for Computational Linguistics

Ranjan, 2022), the Upanishads and the Bhagavad Gita, using pre-trained sentence embeddings obtained from deep learning networks, Sentence Embeddings using Siamese BERT-Networks (SBERT) (Reimers and Gurevych, 2019) and Universal Sentence Encoder (USE) embeddings (Cer et al., 2018). Many hymns in the Rigveda can be attributed to specific devas (deities in Hinduism), such as *Indra* and *Agni* and were predicted using neural network-based word embedding models such as Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014) with linear classifiers (Akavarapu and Bhattacharya, 2023).

Many studies have focused on modelling the syntactics and parsing of the Sanskrit language using various deep learning techniques, such as recurrent neural networks (Aralikatte et al., 2018; Hellwig and Nehrdich, 2018) or transformers (Sandhan et al., 2022; Hellwig et al., 2023; Nehrdich et al., 2024) to generate new Sanskrit text. Another use has been to create sentence and word embeddings using transformers and static models for semantic and analogy tasks (Lugli et al., 2022). Some studies have shown that combining semantic information from the Sanskrit Word Net (Short et al., 2021) with parsed sentences in the Vedic TreeBank (Hellwig et al., 2020) can help to provide better understanding of sentence structure (Biagetti et al., 2023) and may improve Sanskrit language modelling using Sanskrit neural word embeddings (Sandhan et al., 2023)

Only a handful of studies have focused on clustering or stratifying Vedic texts for the purposes of obtaining insights about texts written in Vedic Sanskrit. One such study had performed Bayesian mixture modelling to obtain a chronological ordering of texts written in Vedic Sanksrit, such as the Rigveda, Atharvaveda, and post-Rigvedic Sanskrit, such as the Aitareya Brahmana (Hellwig, 2020). A similar study had analyzed the similarity of passages within the Maitrayani and Kathaka Samhitas using word embeddings (Miyagawa et al., 2024). Another study had performed clustering on the linguistic and textual features of the 10 books in the Rigveda to determine whether the historical order of these books can be obtained in a data-centric way (Hellwig et al., 2021). This study showed that the stratification generally followed the historical divisions.

The Rigveda has been historically divided into ten books of which the oldest parts are Books II to VII (called the "Family Books"), followed by Books I, VIII, and IX which are accepted to be younger than the Family Books, and Book X is the youngest (Jamison and Brereton, 2014). However, no study has directly investigated the possible organization of the suktas in the Rigveda using NLP techniques such as word, sentence, or document embeddings.

3 Aim and Contribution

Thus, the goal of this study was to organize the network of related suktas and uncover the topics contained within the 10 books of the Rigveda in a data-driven manner, without employing prior knowledge about the suktas or topics. Potentially providing a guide for individuals unfamiliar with this complex and varied religious text. This endeavour was mainly facilitated by a novel innovation presented in this study, which we call mean-LSA, where the document vectors obtained using LSA (latent semantic analysis) (Deerwester et al., 1990) were computed from the original length of each sukta (document). This was accomplished by taking the average of all LSA word vectors in a sukta. This is in contrast to obtaining the sukta vectors from suktas that were split into a pre-specified document length, which generally causes a loss of semantic information in normal LSA document vectors.

Another innovation of this study was that the significance of the sukta networks and detected topics were assessed using a null distribution formed by a random permutation of the network adjacency matrices. Although network structure and topics detected may appear well clustered and organized, i.e. visually the documents appear to be clustered with clear structure, the structure may be due to chance occurrence or an inducement of the preprocessing. This test provides an unbiased method of assessing whether real network structure has been found. Using both innovations, this study demonstrated that historically relevant topics in the Rigveda were detected using the mean-LSA embeddings and were more significant and accurate than those obtained by using the deep learning embedding techniques of SBERT and Doc2Vec (Le and Mikolov, 2014), both of which provided non-significant network structure.

4 Methods

The six steps to obtain the network of hymns (suktas) and topics within the Rigveda is summarized in Figure 1. The processing pipeline contained six

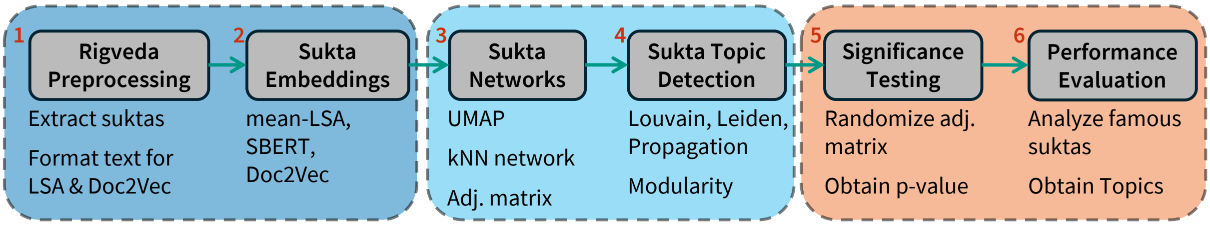

Figure 1: Processing pipeline for obtaining the network of suktas and topics using the three types of embedding techniques (mean-LSA, SBERT, Doc2Vec). Steps (1) and (2) created the embeddings to form the sukta networks. In steps (3) and (4), using the 4-nearest neighbours of each sukta, the network of topics were detected using community detection methods. Finally, in steps (5) and (6), the statistical significance of the detected network structures were determined and the grouped suktas were analyzed.

steps, (1) Rigveda preprocessing to obtain suktas, (2) creation of the sukta embeddings, (3) formation of the sukta similarity network, (4) detection of the topics within the sukta networks, (5) testing of the statistical significance of the sukta networks, and (6) determining the relevance of detected sukta topics.

4.1 Rigveda Prepocessing

To form the network of suktas and detect topics within the Rigveda using word (LSA), sentence (SBERT), or document (Doc2Vec) embeddings, the modern English translation by Jamison and Brereton was used as the source text (Jamison and Brereton, 2014). The Rigveda consists of 10 books (mandalas), 1,028 hymns (suktas), and 10,552 verses (mantras) of varying lengths (Table 1) (Jamison and Brereton, 2014; Tiwari, 2021). Each sukta in the Rigveda is referred to by its mandala and sukta number, e.g., RV 10.129 represents 129th sukta in the 10th mandala, which is the famous Nasadiya sukta in the Rigveda (Jamison and Brereton, 2014).

Book	Hymns	Verses
1	191	2,006
2	43	429
3	62	617
4	58	589
5	87	727
6	75	765
7	104	841
8	103	1,716
9	114	1,108
10	191	1,754

Table 1: Organization of the documents contained in the Rigveda. Each book (mandala) consists of a collection hymns (suktas), and each hymn is composed of a series of verses (mantras) of varying lengths.

The three embeddings (LSA, SBERT, Doc2Vec) require slightly different types of text preprocessing. Common to all methods, the text from the Rigveda was organized at the sukta level, in which all the mantras within a sukta were concatenated together and consider as a single document. For LSA, punctuation, numerals, symbols, and stopwords were removed, followed by a conversion to lowercase letters. For the Doc2Vec, a simple preprocessing of converting all uppercase letters to lowercase and tokenization by space was performed (Le and Mikolov, 2014; Rehurek and Sojka, 2011a). For SBERT, no additional preprocessing was performed (Reimers and Gurevych, 2019).

4.2 Sukta Embeddings

The analysis of the suktas depends on the formation of "sukta embeddings", which are either composed of word, sentence, or document embeddings. In the next subsections, the processing of each method is provided.

4.2.1 mean-LSA

LSA is a classical technique in NLP for obtaining word and document embeddings. Although newer techniques based on deep learning models have been developed, LSA is competitive with methods such as Word2Vec and GloVe on some semantic tasks (Levy et al., 2015). LSA embeddings are computed using singular value decomposition (SVD) (Deerwester et al., 1990) on unigram and TFIDF weighted data of the suktas, which is represented as $\mathbf{X} \in \mathbb{R}^{v \times n}$, where v is the size of the vocabulary, n is the number of suktas, and d is the top singular values (i.e., the dimensionality of the embeddings), giving

$$\mathbf{X}_d = \mathbf{U}_d \mathbf{S}_d \mathbf{V}_d^T. \quad (1)$$

Then, the traditional LSA word embeddings are defined as the rows of

$$\mathbf{W} = \mathbf{U}_d \mathbf{S}_d \tag{2}$$

and document embeddings are defined as the rows of

$$\mathbf{D} = \mathbf{V}_d \mathbf{S}_d. \tag{3}$$

To obtain both type of LSA embeddings, the suktas must be chunked into equal sized documents. This method will provide unique word embeddings, however, the document embeddings will not represent the original suktas, due to the chunking of the texts. To overcome this hurdle, we introduce an innovation of LSA, where for each sukta, the mean of all the word embeddings $\mathbf{w}_i \in \mathbf{W}$ within the sukta is taken to form the sukta embedding $\mathbf{d}_j^{\text{sukta}}$. This method called mean-LSA, creates a unique embedding for each sukta that is representative of the original word length of the sukta. The mean-LSA embeddings have a dimension of 768, to match the pre-trained SBERT embeddings.

4.2.2 SBERT

To obtain the sukta embeddings using SBERT, the pre-trained 768-dimensional sentence embeddings from the all-mpnet-base-v2 sentence transformer model was used (Reimers and Gurevych, 2019). These embeddings have been trained on 1 billion sentence pairs using the self-supervised contrastive learning objective and is ideal for clustering and similarity tasks involving sentences and short paragraphs (Reimers and Gurevych, 2019), similar to the length of suktas. The SBERT model is able to handle variable length documents, without any further processing.

4.2.3 Doc2Vec

Doc2Vec (Le and Mikolov, 2014), creates document embeddings that capture semantic and syntactic properties of variable-length documents. A random document embedding is initialized and fine-tuned by predicting words taken from samples in the document. There are two methods for training Doc2Vec, Distributed Memory (DM) and Distributed Bag of Words (DBOW). The DM method concatenates the document embeddings with the word embeddings, to predict the next word in the document. DBOW uses only the document embedding to predict words within the document. The Gensim implementation of Doc2Vec (Rehurek and Sojka, 2011b) was used to create 768-dimensional sukta embeddings (to match SBERT) with DBOW and trained for 200 epochs.

4.3 Formation of Sukta Networks

The sukta embeddings obtained from mean-LSA, SBERT, and Doc2Vec were reduced in dimensionality using Uniform Manifold Approximation and Projection (UMAP) (McInnes et al., 2018) to improve computational speed and uncover latent structure among the suktas. Then for each embedding method, the 4 k-Nearest Neighbours (kNN) for each sukta embedding was computed and formed into a binarized adjacency matrix. To determine the nearest neighbours, each sukta embedding was normalized to unit norm and the Euclidean distance was computed. The ranking obtained by the Euclidean distance is identical to that obtained by the cosine distance between the embeddings, although the magnitude of the distances may be different. This procedure creates a network of suktas for each of the embedding methods that captures and summarizes the relationships among the suktas.

4.4 Community Detection of Topics

To the detect the community structure within the sukta networks, which may indicate concepts of topics found within the Rigveda, the Louvain (Blondel et al., 2008), Leiden (Traag et al., 2019), and label propagation algorithms (Raghavan et al., 2007) were implemented. The Louvain and Leiden methods attempt to maximize modularity in order to detect communities. Modularity measures the quality of partitioning a network into communities and ranges from [-1,1] (Newman and Girvan, 2004) as $Q = \sum_i (e_{ii} - a_i^2)$, where e_{ii} is the fraction of edges with both nodes in community i, and a_i is the fraction of edges that attach to nodes in community i. The label propagation method attempts to distribute community labels within a detected community in a semi-supervised manner (Raghavan et al., 2007).

4.5 Statistical Significance of Topics

It is necessary to compute the statistical significance of the detected communities within a network to ensure that the observed network structure is not due to chance and the observed groupings represent genuine relationships among the data (Kimes et al., 2017; Lancichinetti et al., 2011; Gultepe et al., 2018; Schrader and Gultepe, 2023). If a high modularity score is obtained, yet with a slight manipulation of the network edges, a similarly high modularity score can be obtained again, then the original modularity score is likely due to a random occurrence of the data. To determine the statistical significance of the network structure, for a predetermined number of iterations, the null distribution was created by randomly permuting the adjacency matrix and performing the relevant network detection method (Gultepe et al., 2018). This procedure was repeated for 5000 iterations and p-value of the original modularity score was obtained by computing the empirical cumulative distribution function (Gultepe et al., 2018). For all tests the significance level was 5%.

4.6 Evaluation of Sukta Topics

For each embedding method, the topic detection method providing the highest modularity score was chosen and then the significance test was performed. After this two-step procedure, to confirm that the relevant groupings of suktas were obtained, the selected suktas by each network was compared to seven famous grouping of suktas. These sukta groupings were: the Creation, Funeral, and Heaven & Earth (Doniger, 1981); Marut (Müller, 1869); Surya and Brihaspati (Chitrav, 2005), and Water (Minter, 1981).

4.7 Experimental Setup

The sukta embeddings from all three methods were row normalized, as it is known to improve representative accuracy (Levy et al., 2015). After grid search, for UMAP it was found that the best parameters for mean-LSA were: number of neighbours = 8, number of dimensions = 10, min distance = 0.0, and metric = Euclidean. For SBERT, the best UMAP parameters were: number of neighbours = 10, number of dimensions = 5, min distance = 0.0, and metric = Euclidean. For Doc2Vec, the best UMAP parameters were: number of neighbours = 10, number of dimensions = 12, min distance = 0.0, and metric = Euclidean. The best network topic detection for mean-LSA, SBERT, and Doc2Vec were obtained Leiden with Dugue modularity, Louvain with Newan modularity, and Louvain with Potts modularity, respectively.

5 Results

Figures 2 (mean-LSA), 3 (SBERT), and 4 (Doc2Vec) show all the detected topic clusters and the grouping of the famous clusters obtained by each of the three sukta embedding methods. The mean-LSA sukta embedding method obtained the best sukta organization, as it was the only significant method ($z = 2.726$, $p < .01$) and was successful in identifying clusters that contained the semantically related suktas for all seven cases. Figures 5, 6, and 7 demonstrate how well the mean-LSA sukta embeddings detected the relevant suktas for each case, as compared to SBERT.

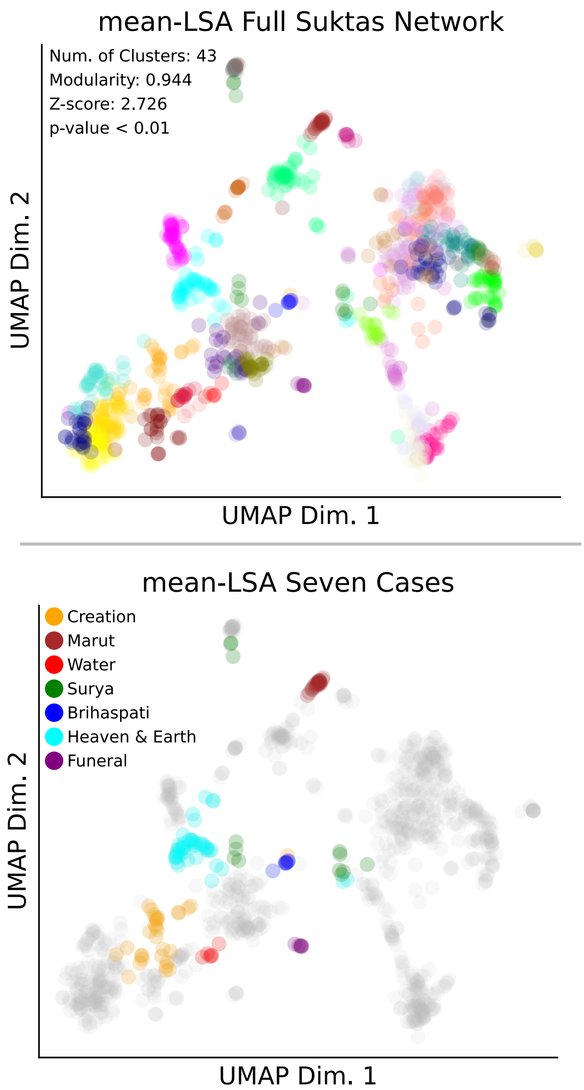

Figure 2: UMAP visualization of the Rigveda sukta network derived from mean-LSA embeddings. Top: The full network representation, shows 43 unique clusters with a modularity of 0.944 that has statistically significance structure ($z = 2.726$, $p < .01$). Bottom: The highlighted clusters represent a subset of seven famous sukta topics - Creation, Marut, Water, Surya, Brihaspati, Heaven & Earth, and Funeral. The mean-LSA embedding network was successful in identifying clusters that contained the semantically related suktas in all seven cases.

Although, the network of suktas found by SBERT embeddings was not statistically significant (z = -0.876, p = .810), we still investigated the individual seven famous cases to determine if there were any relevant groupings of the suktas. Overall, the mean-LSA sukta embeddings selected more of the famous suktas at rate of 71.9% (Table 2) as opposed to the SBERT sukta embeddings which selected the famous suktas at rate of 62.7% (Table 3). We did not investigate the Doc2Vec results any further because not only was the network not significant (z = -0.126, p = .550), there were no meaningful clusters of suktas.

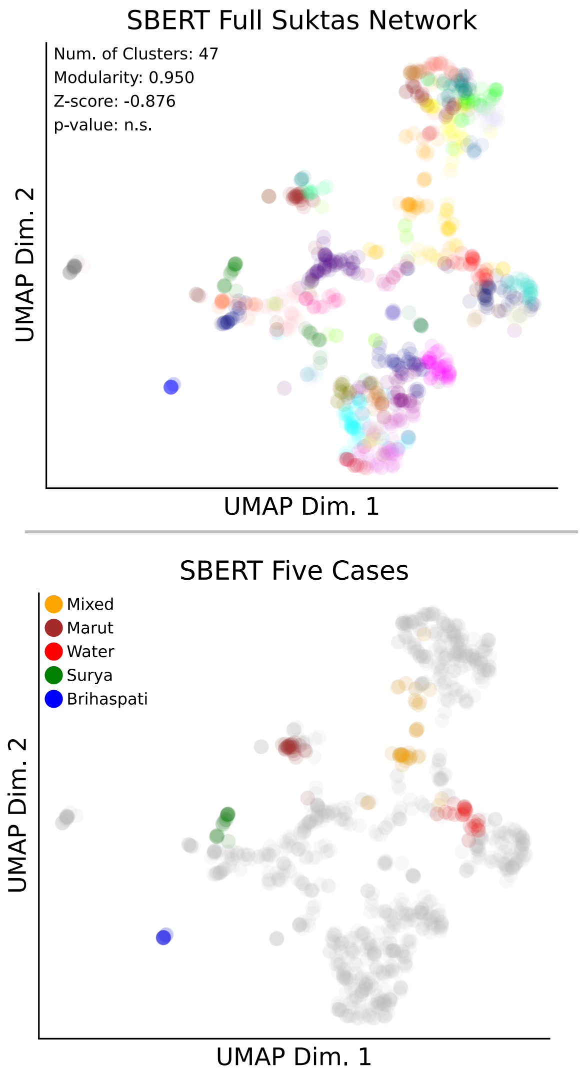

Figure 3: UMAP visualization of the Rigveda sukta network derived from SBERT embeddings. Top: The full network representation, shows 47 distinct clusters with a modularity of 0.950. Although SBERT's modularity is slightly higher than mean-LSA's modularity (0.944), it failed the significance test (z = -0.876, p = .810). Bottom: SBERT failed to separate three different topics of suktas (Creation, Funeral, Heaven & Earth suktas) and clustered them into a single cluster (Mixed).

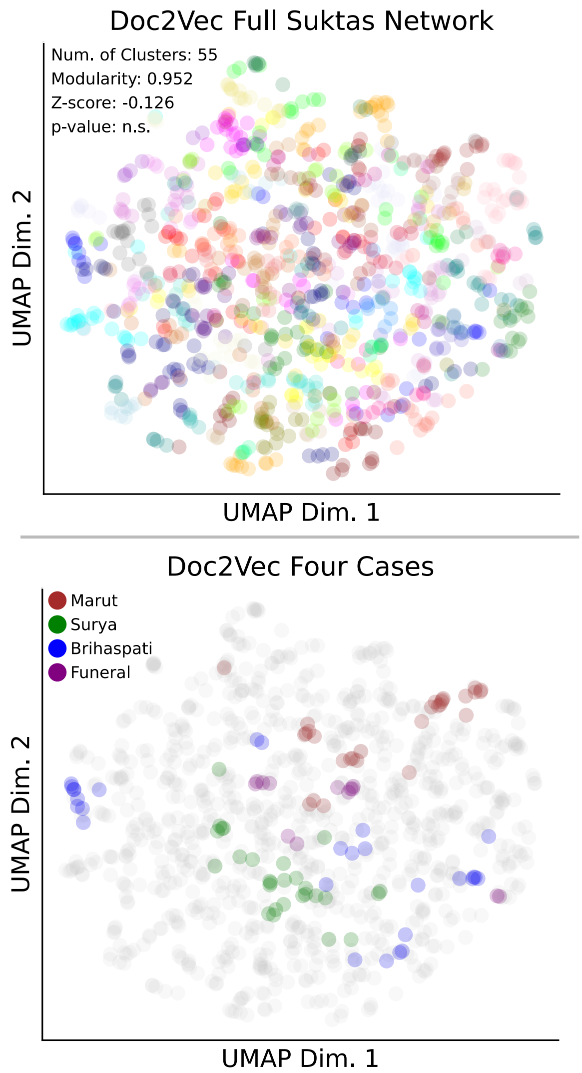

Figure 4: UMAP visualization of the Rigveda sukta network derived from Doc2Vec embeddings. Top: The full network depicts 55 individual clusters with modularity of 0.952, which is the highest among the three sukta embeddings methods. Despite having higher modularity, it was unsuccessful in passing the statistical significance test (z = -0.126, p = .550). Bottom: For three out of the seven famous cases, Doc2Vec failed to group the semantically related suktas into relevant clusters and for the four remaining cases (Marut, Surya, Brihaspati, Funeral) the suktas were irregularly distributed.

6 Discussion and Conclusion

To our knowledge, this is the first study to create a network of suktas contained in the Rigveda. This is accomplished by using the novel method of mean-LSA, which we presented herein. The mean-LSA method creates a document embedding using the word embeddings obtained from LSA by taking the average of the word embeddings for all words contained in a document. Also, we demonstrated

Case	Correct	Missing	Non-famous
Creation	9	0	22
Marut	10	4	14
Water	4	2	2
Surya	6	10	7
Brihaspati	7	2	0
H&E	6	0	41
Funeral	6	6	0

Table 2: Correctly identified famous suktas with mean-LSA. The count of missing famous suktas is also shown along with the selected non-famous suktas. H&E: Heaven and Earth

Case	Correct	Missing	Non-famous
Creation	8	1	30
Marut	12	2	15
Water	4	2	21
Surya	5	11	12
Brihaspati	3	6	7
H&E	6	0	32
Funeral	4	8	34

Table 3: Correctly identified famous suktas with SBERT. The count of missing famous suktas is also shown along with the selected non-famous suktas. H&E: Heaven and Earth

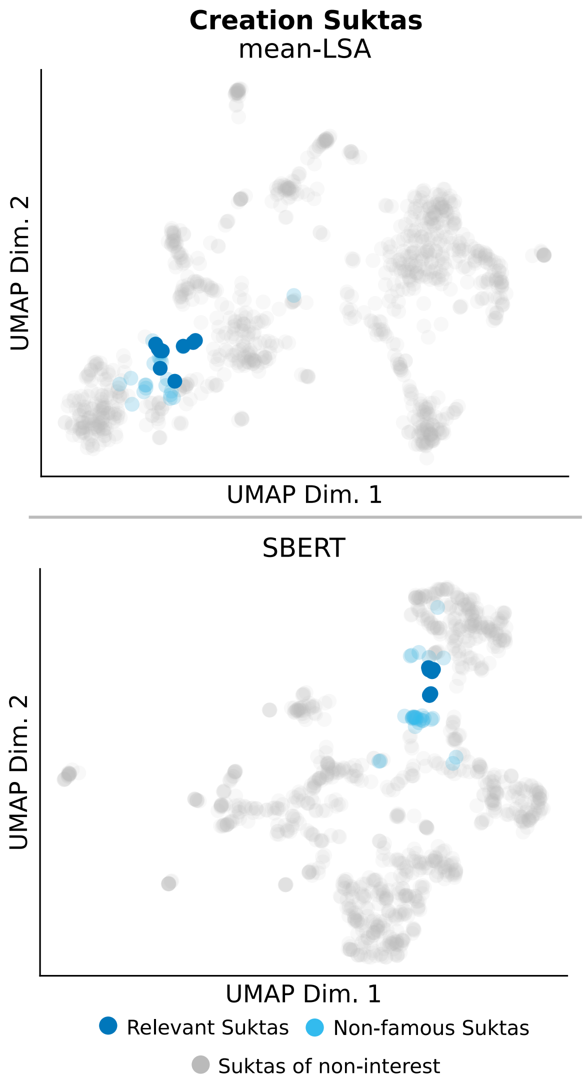

Figure 5: Comparison of the Creation sukta clusters for the mean-LSA and SBERT sukta embeddings. Top: The network of famous Creation suktas using mean-LSA has gathered all the well-known nine suktas (relevant suktas) into a single cluster with 22 other non-famous suktas. Bottom: SBERT has categorized eight of the nine popular creation suktas together. However, this cluster also contains suktas from other two topics (Funeral and Heaven & Earth), indicating that SBERT failed to distinguish suktas belonging to other topics.

that despite having a high modularity score, this may not be indicative of actual topics found by the network structure. This was corroborated by obtaining the significance values of the network structure through randomization of the network adjacency matrices.

This was further demonstrated by the discrepancy of the modularity scores and the signifance values. The Doc2Vec based sukta network, which had the highest modularity score, did not have a statistically significant structure and it failed to detected any meaningful sukta topic communities, especially in the case of the seven famous suktas. The SBERT based network had a similar situation, in which the modularity score was the second highest, yet it was also not statistically significant. When analyzing the seven famous suktas, it mistakenly combined the Funeral suktas with the Creation, and Heaven & Earth suktas.

It may be possible to use the presented statistical significance testing method as a way of determining the cohesiveness and unity of the detected topics. This could be similar to the computation of coherence measures that indicate the relevance of topics against the co-occurrence of words in a topic (Röder et al., 2015). However, the statistical test performed with the random permutation of the adjacency matrix may be considering higher-order concepts, since it is manipulating the connections between documents, rather than only analyzing the collection of words. The underlying premise here is that documents are not simply a collection of words. We plan to investigate this application of statistical significance testing of detected topics in

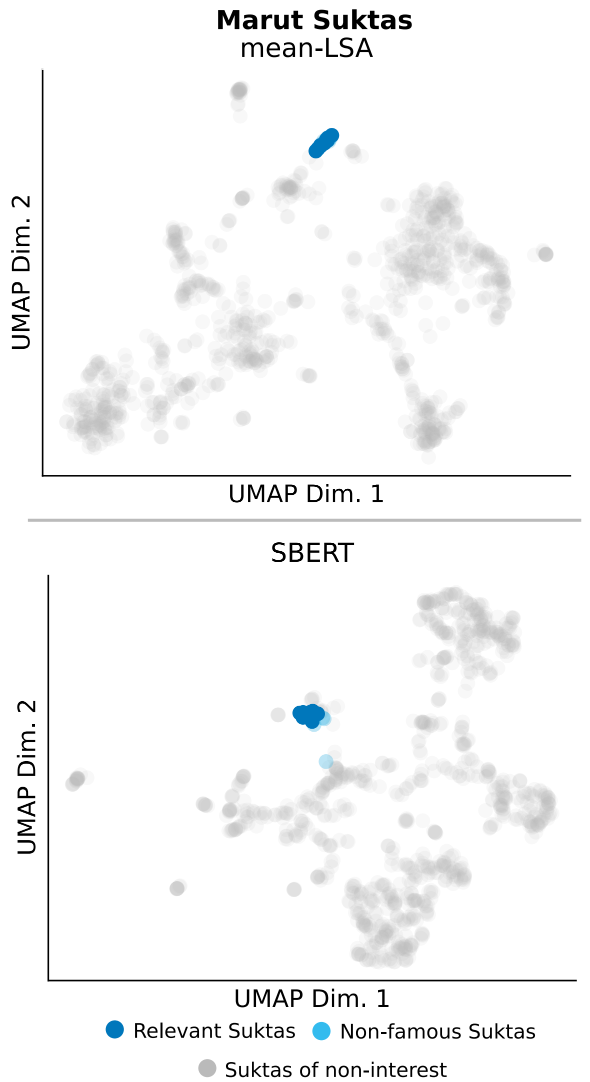

Figure 6: Comparison of the Marut sukta clusters for the mean-LSA and SBERT sukta embeddings. Top: mean-LSA has clustered ten relevant Marut suktas out of the total 14 famous suktas, alongside 14 other non-famous suktas. Bottom: In the case of SBERT, 12 out of the 14 famous Marut suktas, only two were missing and were placed together with 15 non-famous suktas.

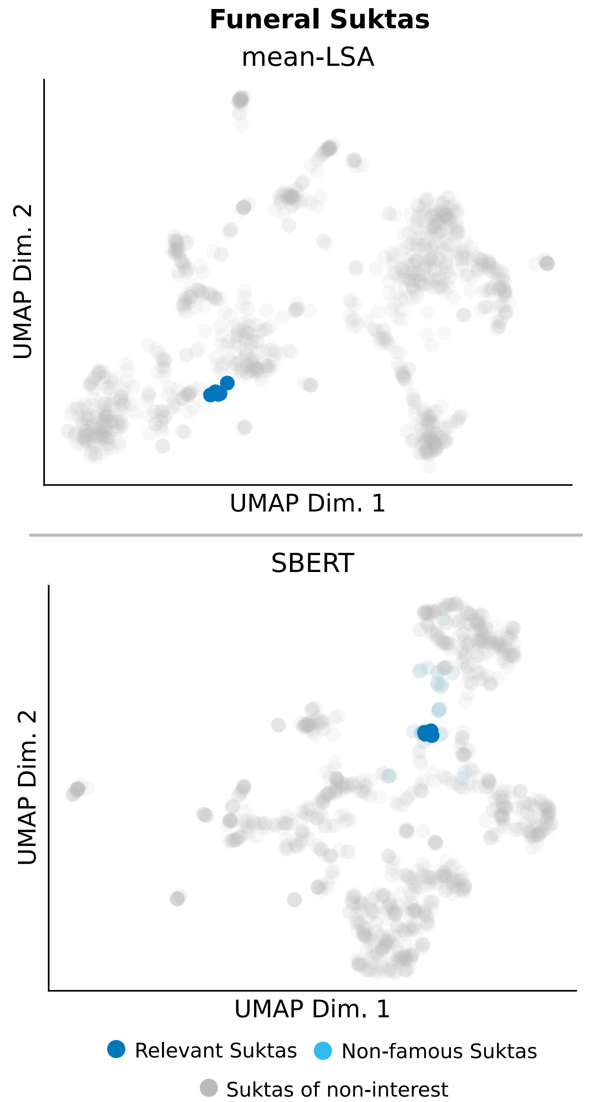

Figure 7: Comparison of the Funeral sukta clusters for the mean-LSA and SBERT sukta embeddings. Top: mean-LSA successfully captured four out of the six famous funeral suktas along with two Yama (God of Death) suktas, which are also related to funerals. With a total cluster size of six suktas, mean-LSA only identified suktas related to funerals and Yama, without including any non-famous suktas. Bottom: SBERT clustered four suktas related to funerals, consisting of one famous funeral sukta along with three Yama suktas. It mistakenly also captured four suktas related to other topics (Creation, and Heaven & Earth), indicating that SBERT struggled to separate the suktas based on their topics.

a future study. We also plan to investigate the training of unsupervised transformer language models.

7 Limitations

Despite its reliability, the main limitation of this work is that the network analyses relied on a single modern English translation. Thus, as with all translations, the original meaning of the Rigveda in the Vedic Sanskrit may have been masked, since the ability to transmit the true meaning will depend on the ability of the translators to translate the text. For future studies, comparison with the Sanskrit version of the Rigveda is planned.

8 Ethics Statement

The Rigveda is a sacred text in Hinduism and we have been careful to present it in the best way possible, by highlighting important suktas that may be of interest to a wide audience of individuals who may want to learn more about the Hindu religion.

References

V.S.D.S.Mahesh Akavarapu and Arnab Bhattacharya. 2023. Creation of a digital rig Vedic index (anukramani) for computational linguistic tasks. In *Proceedings of the Computational Sanskrit & Digital Humanities: Selected papers presented at the 18th World Sanskrit Conference*, pages 89–96, Canberra, Australia (Online mode). Association for Computational Linguistics.

Rahul Aralikatte, Neelamadhav Gantayat, Naveen Panwar, Anush Sankaran, and Senthil Mani. 2018. Sanskrit sandhi splitting using seq2(seq)2. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4909–4914, Brussels, Belgium. Association for Computational Linguistics.

Erica Biagetti, Chiara Zanchi, and Silvia Luraghi. 2023. Linking the Sanskrit WordNet to the Vedic dependency treebank: a pilot study. In *Proceedings of the 12th Global Wordnet Conference*, pages 77–83, University of the Basque Country, Donostia - San Sebastian, Basque Country. Global Wordnet Association.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *J. Stat. Mech. Theory Exp.*, 2008(10):P10008.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Rohitash Chandra and Mukul Ranjan. 2022. Artificial intelligence for topic modelling in hindu philosophy: Mapping themes between the upanishads and the bhagavad gita. *Plos one*, 17(9):e0273476.

Siddhesvarashastri Chitrav. 2005. *Vaidik Suktapath*, volume 1. Bharatiya Charitrakosha Mandal.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Swami Vishnu Devananda and Vishnu Devananda. 1999. *Meditation and mantras*. Motilal Banarsidass Publishing.

Wendy Doniger. 1981. *The Rig Veda: an anthology: one hundred and eight hymns, selected, translated and annotated*, volume 402. Penguin.

Eren Gultepe, Thomas E Conturo, and Masoud Makrehchi. 2018. Predicting and grouping digitized paintings by style using unsupervised feature learning. *Journal of Cultural Heritage*, 31:13–23.

Eren Gultepe and Vivek Mathangi. 2023. A quantitative social network analysis of the character relationships in the mahabharata. *Heritage*, 6(11):7009–7030.

Oliver Hellwig. 2017. Stratifying the mahābhārata: The textual position of the bhagavadgītā. *Indo-Iranian Journal*, 60(2):132 – 169.

Oliver Hellwig. 2020. Dating and stratifying a historical corpus with a Bayesian mixture model. In *Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages*, pages 1–9, Marseille, France. European Language Resources Association (ELRA).

Oliver Hellwig and Sebastian Nehrdich. 2018. Sanskrit word segmentation using character-level recurrent and convolutional neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2754–2763, Brussels, Belgium. Association for Computational Linguistics.

Oliver Hellwig, Sebastian Nehrdich, and Sven Sellmer. 2023. Data-driven dependency parsing of Vedic Sanskrit. *Language Resources and Evaluation*, 57(3):1173–1206.

Oliver Hellwig, Salvatore Scarlata, Elia Ackermann, and Paul Widmer. 2020. The treebank of vedic Sanskrit. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5137–5146, Marseille, France. European Language Resources Association.

Oliver Hellwig, Salvatore Scarlata, and Paul Widmer. 2021. Reassessing rigvedic strata. *Journal of the American Oriental Society*, 141(4):847–865.

Ben Hutchinson. 2024. Modeling the sacred: Considerations when using religious texts in natural language processing. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1029–1043, Mexico City, Mexico. Association for Computational Linguistics.

Stephanie W Jamison and Joel P Brereton. 2014. *The Rigveda: 3-Volume Set*. Oxford University Press.

Patrick K. Kimes, Yufeng Liu, David Neil Hayes, and James Stephen Marron. 2017. Statistical significance for hierarchical clustering. *Biometrics*, 73(3):811–821. Place: England.

Andrea Lancichinetti, Filippo Radicchi, José J. Ramasco, and Santo Fortunato. 2011. Finding statistically significant communities in networks. *PLOS ONE*, 6:1–18.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1188–1196, Bejing, China. PMLR.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Ligeia Lugli, Matej Martinc, Andraž Pelicon, and Senja Pollak. 2022. Embeddings models for buddhist Sanskrit. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3861–3871, Marseille, France. European Language Resources Association.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

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*, volume 26. Curran Associates, Inc.

Michael W. Minter. 1981. *Water Symbolism In The Rgveda*. Ph.D. thesis. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2023-02-19.

So Miyagawa, Yuki Kyogoku, Yuzuki Tsukagoshi, and Kyoko Amano. 2024. Exploring similarity measures and intertextuality in Vedic Sanskrit literature. In *Proceedings of the 4th International Conference on Natural Language Processing for Digital Humanities*, pages 123–131, Miami, USA. Association for Computational Linguistics.

Friedrich Max Müller. 1869. *Rig-Veda-Sanhita: the sacred hymns of the Brahmans*, volume 1. Trübner.

Sebastian Nehrdich, Oliver Hellwig, and Kurt Keutzer. 2024. One model is all you need: ByT5-Sanskrit, a unified model for Sanskrit NLP tasks. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13742–13751, Miami, Florida, USA. Association for Computational Linguistics.

M. E. J. Newman and M. Girvan. 2004. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69:026113.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. 2007. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 76(3):036106.

Radim Rehurek and Petr Sojka. 2011a. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2):2.

Radim Rehurek and Petr Sojka. 2011b. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2):2.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, page 399–408, New York, NY, USA. Association for Computing Machinery.

Jivnesh Sandhan, Om Adideva Paranjay, Komal Digumarthi, Laxmidhar Behra, and Pawan Goyal. 2023. Evaluating neural word embeddings for Sanskrit. In *Proceedings of the Computational Sanskrit & Digital Humanities: Selected papers presented at the 18th World Sanskrit Conference*, pages 21–37, Canberra, Australia (Online mode). Association for Computational Linguistics.

Jivnesh Sandhan, Rathin Singha, Narein Rao, Suvendu Samanta, Laxmidhar Behera, and Pawan Goyal. 2022. TransLIST: A transformer-based linguistically informed Sanskrit tokenizer. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6902–6912, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Samuel R. Schrader and Eren Gultepe. 2023. Analyzing indo-european language similarities using document vectors. *Informatics*, 10(4).

William Michael Short, Silvia Luraghi, and Erica Biagetti. 2021. Sanskrit wordnet. `https://sanskritwordnet.unipv.it/`. Accessed: (Oct. 4, 2024).

Caley Charles Smith. 2019. *The Invisible World of the Rigveda*, pages 1–13. John Wiley & Sons, Ltd.

Shashi Tiwari. 2021. Rigveda. `https://vedicheritage.gov.in/samhitas/rigveda/`. Accessed: (Oct. 11, 2024).

Vincent A Traag, Ludo Waltman, and Nees Jan Van Eck. 2019. From louvain to leiden: guaranteeing well-connected communities. *Scientific reports*, 9(1):1–12.