Evaluating the Amarkosha to Generate Computational Model for Sanskrit Vocabulary and Sanskrit Word Bank

Kaustav Sanyal[1], Pankaj K. Goswami[2], Neelima Pathak[3]

[1]Department of Computer Science & Engineering, Sarala Birla University, Ranchi, India
Email: kaustavsanyal8@gmail.com
[2]Department of Computer Science & Engineering, Sarala Birla University, Ranchi, India
Email: pankaj.goswami@sbu.ac.in
[3]Department of Humanities & Linguistics, Sarala Birla University, Ranchi, India
Email: neelima.pathak@sbu.ac.in

Received: 10.07.2024 Revised: 15.08.2024 Accepted: 24.09.2024

ABSTRACT
Amarkosha is considered to be one of the most complete word banks ever generated for the Sanskrit language. It has the listing of almost 10,000 words along with their morphological construct, a list of paryayavachi words (synonyms), and their gender study or linganushasanam. The scripture is divided into three sections listing 27 clusters of words. The last cluster of the last section is completely dedicated to defining the genders of the words. The scripture itself is so composed that a computational model for the Sanskrit vocabulary can easily be generated from it. As natural language processing (NLP) for any language needs a good word bank along with all its characteristics and behavioral aspects, in this paper we have made an effort to cluster the Sanskrit vocabulary and construct the computational model for the Sanskrit word bank. The clustering of the words is made by two standard methods, k-means clustering and the Louvain community detection method. In a comparative study of both methods, we have observed the Louvain method to be more efficient in clustering the Sanskrit vocabulary as it provides the output that aligns with the original construct and clusters of the Amarkosha itself. Louvain method gives the output of 24 distinct communities for the words, whereas k-means clustering gives 36 clusters as output. This gives 88% accuracy for the Louvain community detection method and 67% accuracy for k-means clustering.

Keywords: Amarkosha, Natural Language Processing, Word Bank, Clustering, K-Means Clustering, Louvain Community Detection, Sanskrit.

1. INTRODUCTION

Word repository formation and its representation as a computational model have been one of the most crucial steps in processing natural languages. Sanskrit, being one of the most ancient languages of the world, provides a very simple approach in this domain. As Maharshi Panini's Ashtadhyayi provides an artificial language model for Sanskrit along with numerous other Vyakarana (Grammar) Shastras (validated scriptures), there exist several word repositories as well [1]. The word repository scriptures for Sanskrit are known as the Kosha Granthas. Before the regular dictionaries and thesaurus came into existence, almost 70 Kosha Granthas were composed for various domains in Sanskrit [2]. The Amarkosha excelled all of them in its composition and unique features. The Amarkosha was composed by Amarsingh Rav during the reign of Chandragupta II around the fourth century [3]. The scriptures list around 10,000 noun clauses. For its unique features, the Amarkosha is considered to be such a scripture that not only should be studied but memorised. Some of its features are,

- Classification of words into various categories,
- Listing of numerous synonyms under the label of Paryaya Vachi words,
- Cross-referencing of the words according to their gender specification,
- Cross-referencing of words according to their context of use with words from different classes [4].

Analysing the characteristics of the categories defined in the Amarkosha, we can identify them as distinct word clusters. Amarsingh Rav manually did not just list around 10,000 words for the Sanskrit language but clustered them into 27 clusters. In the 2nd shloka of the scripture, Amarsingh Rav mentioned his work as "nam-linga-anushasanam"[3], which means the discipline of noun clauses and genders. Over 40 commentaries and explanations have been composed to date on this work only. The construct of the scripture includes three sections, namely SwargadiKandam, BhuvargadiKandam, and SamanyadiKandam.

27 clusters under different titles have been composed in these sections. The last section of the scripture deals with the gender definitions and their associations with the vocabulary. This paper focuses on the word bank that can be extracted from the Amarkosha. Through two different clustering methods, we have tried to match the results of the clustering to the original scripture. With the help of k-means clustering and the Louvain community detection method, we have tried to generate a mathematical model of the Amarkosha that can be used in NLP for Sanskrit language as a full-length word repository.

2. LITERATURE SURVEY

An effort to develop a fully functioning expert system for the Sanskrit language has been made in recent times. But the greatest challenge in the case has always been the domain knowledge. Either the researchers, experts in machine learning, and computational studies lack the traditional knowledge system of Sanskrit or vice versa. Yet some of the greatest works in the domain of computational Sanskrit have been done in recent times. Few of the works mentioned have a direct association with the problem statement of this paper— the generation of a mathematical model for the Sanskrit vocabulary.

Chandran Savithri Anoop and Angarai Ganesan Ramakrishnan [5] in their work have considered syllable-based modeling units for Sanskrit and other Indian languages. Taking into consideration three representations of texts (native script, Sanskrit library phonetics (SLP1) encoding, and syllables), they have tokenised them to evaluate the performances of the tokens for monolingual training and cross-lingual transfer learning. Where they have established the efficiency of SLP1 over a syllable-based approach, Muskaan Singh et.al [6] have implemented a deep neural network on corpus-based machine translation. In their effort to design a Sanskrit-Hindi translation system, they have taken the input from Bhagavat Gita and trained the system with a deep neural network, auto-tuning the input data for better results.

Jivnesh Sandhan et.al [7] in their design of a neural Sanskrit NLP toolkit named 'SanskritShala', have depicted word segmentation, morphological tagging, dependency parsing, and compound type identification. In their work they have used open-access Sanskrit corpora and word similarity, relatedness-based data sets to train their system for their web-based application with 7 different word embedding models. On the other hand, Sujoy Sarkar et.al [8] have done great work in introducing an annotation tool to crowd-source a Sanskrit dataset. In their work they have recognized the named entities from Srimad Bhagavatam with the help of a string-matching heuristic approach and their algorithm identified the named entities with an F-score of 0.80 which we can consider to be reasonable for Sanskrit, being a low-sourced language. J.N.Tripathi [9] in his work did not directly address the Sanskrit language but computed the occurrence of Devanagari alphabets and symbols for the Hindi language. The depiction he made in his work, of Hindi to be less redundant than English for its phonetic nature, serves our purpose as well because the Sanskrit writing system is based on the same Devanagari script.

Moving back in time, Akshar Bharati et. al[10] used Amarkosha directly to prepare a WordNet for the Sanskrit word network. In their work, they have established the remarkable resemblance of Hindi synsets with the ones extracted from the Amarkosha. They have used the resemblance to improve the WordNet for the Hindi language. As a child language of Sanskrit, the WordNet of Hindi inevitably gets richer with its derivation from the Sanskrit vocabulary. On the other side with the Hindi synsets Sanskrit language's synsets got quantitively enhanced. Sivaja S. Nair and Amba Kulkarni [11] in their work considered the inter-relations among the words listed in Amarkosha. From the linear listed structure, they reconstructed the links by dynamically analysing the table to form a network out of the Amarkosha. However, their complete work is recorded in their full-length thesis, which gives more elaborated insight into the domain [12]. Piyush Jha et.al [13] in their work depicted Neural Machine Translation's application to build a Vocabulary-based Translation for Low-resource Language Pair of Sanskrit-Hindi. Sitender and Seema Bawa [14] in their work have explained the mechanism of translating the Sanskrit language into Universal Networking Language. In this work as well, the root guidelines of Sanskrit grammar have not been utilised.

In all the above-mentioned works, which have worked with close association to the Sanskrit vocabulary we have observed a lack of scriptural accuracy and consideration. Sanskrit is a very scripture-oriented language and traditionally rich background; we have pointed out an acute lack of scriptural references in them. The reference work composed by Bhattoji Dikshita on the Ashtadhyayi of Panini [15], commentary of Jayaditya and Vamana on the same [16], and the composition of all the root verbs [17] have to be taken into consideration to analyse the word-bank of the Sanskrit language. The Varga classifications, word clusters, Linganushasanam, and Paryaya Vachi words are never taken into consideration to generate a fully developed Word clustering to design a computational model for the Sanskrit vocabulary in our work we have made an effort to fill the gap with the help of standard NLP based clustering methods that are explained in the following sections.

On the other hand, for the technical and applied aspects of this work, standard clustering methods have been analysed. Malay K. Pakhira [18] in his work has depicted the linear time complexity of the K-Means algorithm with his unique approach where he managed to modify the k-means algorithm with the time complexity of O(n), instead of $O(n^2)$. Jun Zhang et. al [19] in their work have taken a different approach. They have proposed a unique approach for a more efficient Louvain Community Detection Method especially designed for large networks.

3. METHODOLOGY

The structure of the Amarkosha is a clustered composition itself. The clustering is done on several basis following the foundation of the Shastras. In Amarkosha not only the clustering of Sanskrit vocabulary is done, but several attributes of every word are also defined with precision. Among them we have taken into consideration, only the following attributes, Varga, Linga, and Paryaya Vachi words. To cluster the Sanskrit vocabulary, we have taken two standard methods of clustering to make a comparative study, the K-Means algorithm and the Louvain method [20] for community detection on the dataset [21]. Though clustering and community network generation have the same objective for sparse data, they have their domain of application where they differ from each other in certain aspects. But in the case of clustering of Sanskrit vocabulary, it is hard to decide which approach is to consider for the behavioral nature of the Sanskrit words. That is why we have considered the standard approaches for each of the types to clusterize the Sanskrit vocabulary. Along with Python programming, we have taken the assistance of the Gephi tool to visualise the community networks.

A. K-Means Clustering:

Being the most popular approach for clustering, k-means deals with unlabelled data. As an unsupervised learning, k-means clustering generates the clusters based on the degree of similarity between points. We have implemented this approach to the Sanskrit vocabulary to find out the accuracy of the clusters concerning the scriptural reference of the Paryaya Vachi words in Amarkosha. As each of the words includes a list of such Paryaya valid words along with their genders and categories (scriptural reference-based), the k-means clustering is viable to consider for the clustering purpose. The number of optimum clusters for the dataset is defined using the Elbow method and given by,

$$WSS = \sum_{i=1}^{m} (x_i - c_i)^2 \qquad (1)$$

Where WSS [22] is Within the sum of squares, x_i is the i^{th} word in the vocabulary, and c_i is the closest point to the centroid. From the elbow point, we have decided the optimum number of clusters for the dataset.

B. Louvain Community Detection:

Louvain community detection is a popular approach for generating community networks for similar behaving nodes for large networks. For running the Louvain method on the Amarkosha dataset we primarily need to generate a network from it. The modularity ΔQ of the network is computed by,

$$\Delta Q = \left[\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m}\right)^2\right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right] \qquad (2)$$

Where $\sum_{in}$ is the sum of all the in-degree edges in a centroid of a community, $\sum_{tot}$ is the sum of all the out-degree nodes from a centroid C, k_i is the sum of in-degree edges of a word node, $k_{i,in}$ is the sum of out-degree edges of a word node, m is the sum of the weights of all edges in the network [20].

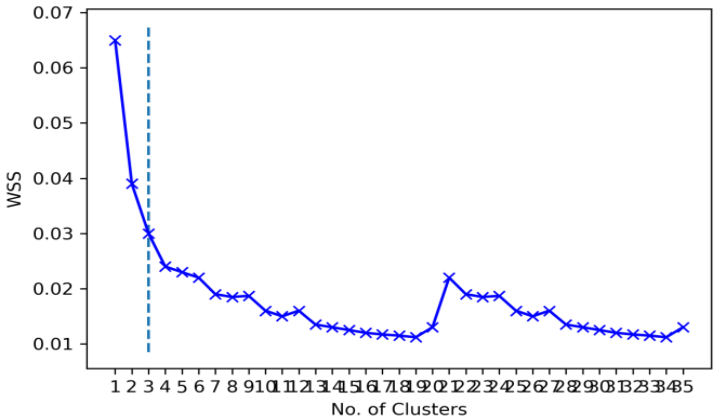
Implementation of this approach to the Sanskrit vocabulary has been taken into consideration for the potential of the dataset to build a network. Each of the words has a gender and is associated with a specific scriptural category, along with a list of Paryaya Vachi words, the nodes (representing a word) will have the possibility to have several in-degree and out-degree nodes that will make the clustering closer to the clusters mentioned in the Amarkosha.

4. RESULT & DISCUSSION

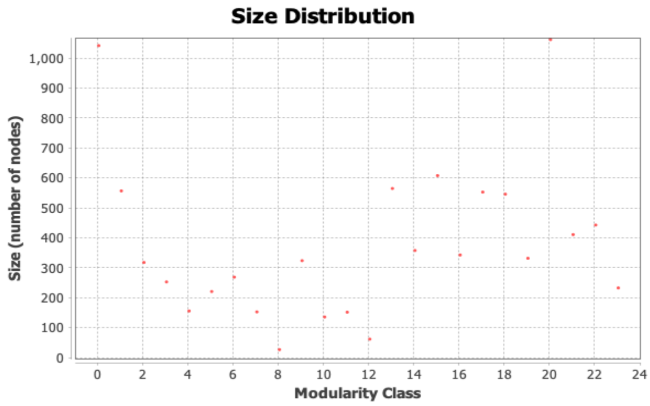
Running both the approaches on the dataset we have results that allow us to evaluate the Sanskrit vocabulary more efficiently for natural language processing.

A. The K-Means Clustering:

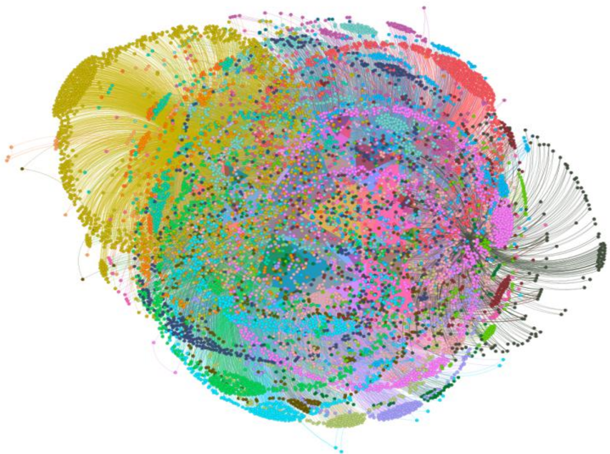
The implementation of k-means clustering gave rise to 36 clusters from the Sanskrit vocabulary dataset. With the elbow method [23] we got the elbow point at the value of 3. Figure 1. shows the elbow point on the value 3 for the given dataset. This denoted the optimal number of clusters for the Sanskrit vocabulary should be 3. With an increasing number of clusters after 3, the behavior of the WSS value gets stabilised and does not show any significant behaviour. The centroids for the first three clusters are labeled as, 'स्वर्गवर्गः', 'पुं', and 'स्त्री'.



B. The Louvain Community Detection:
On the other hand, the Louvain Community Detection on resolution set to 0.5 [24], gave rise to 24 communities as shown in Figure 2. The modularity for the network is computed to be 0.485.
C. Discussion:
The Amarkosha defines 27 clusters for the Sanskrit vocabulary. 11 additional clusters were made that defined the genders of the words. Those additional clusters can be defined



as the principle of genders (Linganushasanam) in the scripture [25]. The result from the k-means clustering method exactly computed 36 clusters from the data. But the clusters it made had errors. The list below provides the clusters defined in the Amarkosha and the clusters computed from the k-means clustering.The clusters made from the k-means method have dual values and do not differentiate the Varga and the Linganushasanam of the Amarkosha [26]. Considering 27 clusters of Amarkosha without the broad classification, the accuracy is 67%.



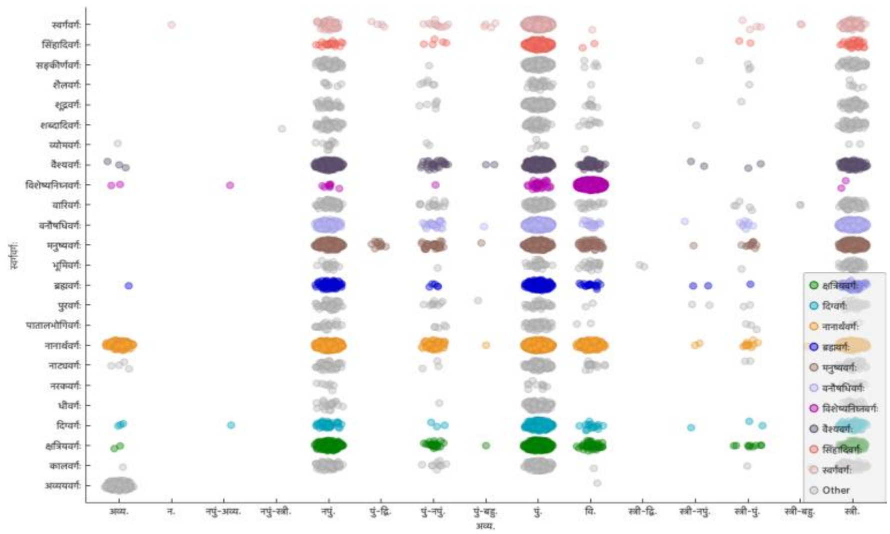
On the other hand, the Louvain community detection method gave 24 communities. Each community is generated around the centroids labeled 24 clusters defined under the Varga classification of Amarkosha. The Linganushasanasm clusters of Amarkosha are not distinctly computed but duly connected to the associative Vargas and they are 10 in number as nodes with only in-degree edges. These nodes denote the association of the genders to the words. Moreover, the community network allowed us to evaluate the Paryaya Vachi words in every community. For example, the word 'पिशाच', a पुंगलिंग word is connected to the Paryaya of 'देवयोनि' under the community' स्वर्गवर्ग'. This gives us an accuracy of 88% for the Sanskrit vocabulary. Figure 3. shows the depiction of the Community Network that has evolved from the Amarkosha using the Louvain Community Detection algorithm. Here we have provided the list of the clusters defined in the Amarkosha, clusters made by the k-means method, and the communities generated from the Louvain community detection method.

I. The Amarkosha Clusters:स्वर्गवर्गः, व्योमवर्गः, दिग्वर्गः, कालवर्गः, धीवर्गः, वाग्वर्गः, शब्दादिवर्गः, नाट्यवर्गः,,पातालभोगिवर्गः, नरकवर्गः, वारिवर्गः , भूमिवर्गः, पुरवर्गः, शैलवर्गः, वनौषधिवर्गः, सिंहादिवर्गः, मनुष्यवर्गः, ब्रह्मवर्गः, क्षत्रियवर्गः,, वैश्यवर्गः, शूद्रवर्गः, विशेष्यनिघ्नवर्गः, संकीर्णवर्गः, नानार्थवर्गः, नानार्थाव्ययवर्गः, अव्ययवर्गः, लिंगादिसंग्रहवर्गः.

II. K-Means Clusters:शूद्रवर्गः, वनौषधिवर्गः, सिंहादिवर्गः, नानार्थवर्गः, नानार्थवर्गः, क्षत्रियवर्गः, शब्दादिवर्गः, नानार्थवर्गः, मनुष्यवर्गः, क्षत्रियवर्गः, नाट्यवर्गः, नानार्थवर्गः, ब्रह्मवर्गः, विशेष्यनिघ्नवर्गः, शब्दादिवर्गः, ब्रह्मवर्गः, वनौषधिवर्गः, नानार्थवर्गः, वनौषधिवर्गः, सिंहादिवर्गः, वैश्यवर्गः, नानार्थवर्गः, विशेष्यनिघ्नवर्गः, वैश्यवर्गः, विशेष्यनिघ्नवर्गः, नाट्यवर्गः, नानार्थवर्गः, ब्रह्मवर्गः, विशेष्यनिघ्नवर्गः, शब्दादिवर्गः, सिंहादिवर्गः, नानार्थवर्गः, नानार्थवर्गः, क्षत्रियवर्गः, शब्दादिवर्गः, विशेष्यनिघ्नवर्गः.

III. Louvain Communities:स्वर्गवर्गः, व्योमवर्गः, दिग्वर्गः, कालवर्गः, धीवर्गः, वाग्वर्गः, शब्दादिवर्गः, नाट्यवर्गः, पातालभोगिवर्गः, नरकवर्गः, वारिवर्गः ,भूमिवर्गः, पुरवर्गः, शैलवर्गः, वनौषधिवर्गः, सिंहादिवर्गः, मनुष्यवर्गः, ब्रह्मवर्गः, क्षत्रियवर्गः, वैश्यवर्गः, शूद्रवर्गः, विशेष्यनिघ्नवर्गः, अव्ययवर्गः, नानार्थाव्ययवर्गः.

The list is provided as the output without any variation in the order. Figure 4. shows the distribution of the various clusters against the gender-based categorization depicting the "Naam-linga-anushasanam" (discipline of names and genders of the words) mathematically.



5. FUTURE SCOPE

The scope of this research can be extended in several directions, providing deeper insights into the clustering of Sanskrit vocabulary and its alignment with classical linguistic frameworks. One promising area for future work is to incorporate the morphological structure of Sanskrit words more comprehensively. This could be achieved by analyzing the clustered word network based on their roots (Dhatu) and suffixes (Pratyay), as described in Maharshi Panini's Dhatupatha. The Dhatupatha is a foundational text that lists all the Dhatus along with their meanings and applications. By leveraging this resource, future studies can refine the word clusters not only by semantic proximity, as is done using the Amarkosha, but also by morphological similarity. This would enable a more granular and linguistically informed understanding of the Sanskrit lexicon. Further research could aim at developing a computational framework that integrates both the Amarkosha and Dhatupatha for more nuanced clustering. The current work has established the Louvain method as an efficient community detection algorithm for Sanskrit vocabulary. By expanding the dataset to include the morphological and semantic insights from Dhatupatha, future research could result in the creation of an expert system capable of handling more advanced tasks in Natural Language Processing (NLP). Specifically, such a system could significantly enhance the development of Natural Lan-

guage Understanding (NLU) and Natural Language Generation (NLG) tools for Sanskrit, which remain relatively underexplored in comparison to other languages.
There is also the potential to incorporate other classical linguistic texts like Panini's Ashtadhyayi, which defines the rules of Sanskrit grammar, and Yaska'sNirukta, which explains the etymology of words. By integrating these texts, future models could ensure a more contextually and syntactically accurate clustering process. This would allow the computational outputs to align not only with the lexical repositories like the Amarkosha, but also with the grammatical rules and etymological structures that govern the Sanskrit language. These advancements could contribute to a wide range of applications, from automated translation and linguistic analysis of ancient texts to the modernization of Sanskrit computational tools. This work would also support the preservation and digitization of Sanskrit knowledge systems, ensuring that traditional linguistic frameworks are seamlessly integrated with contemporary computational methods.

CONCLUSION

Concerning time complexity, though k-means clustering has the time complexity of $O(n^2)$ and the Louvain method has the time complexity of O(nlog2n), keeping the degree of accuracy in mind, the Louvain method of community detection has been proven to be more efficient in clustering the Sanskrit vocabulary. The Louvain method has also been proven a more efficient approach to cluster the Sanskrit vocabulary according to the Amarkosha. Sanskrit being a very scripture-oriented language, the computation results are expected to align with the postulates of the Shastras. The Amarkosha being one of the most celebrated Kosha Granthas of Sanskrit, is considered to be an authority in the case of word embeddings in the Sanskrit language. Thus, the Louvain method is established to be a better and efficient approach to cluster the Sanskrit vocabulary for its alignment to the definition of the Amarkosha.
The Amarkosha is not only a word repository for the Sanskrit vocabulary, but it also provides the morphological structure of each of the words. The morphological structure of the Sanskrit words has two major parts, the root (Dhatu) and the suffix (Pratyay). As a future work, the network can be evaluated in terms of their Dhatus from the Root Repository composed by Maharshi Panini, named the Dhatupatha. As the Dhatupatha lists all the Dhatus along with their meanings and domain of application, the work can be a good step towards developing an expert system that will be capable of Natural Language Understanding and Natural Language Generation for the Sanskrit language.

REFERENCES

[1] Kak, S.C., 1987. The Paninian Approach to Natural Language Processing. International Journal of Approximate Reasoning, 1:117-130.

[2] Staal, J.F., 1972. A Reader on the Sanskrit Grammarians. MIT Press, Cambridge, Massachusetts, and London, England.

[3] Abhimanyu, M. L., 1937. The Amara-Kosha of Shri Amara Sinha. Master Khelarilal& Sons., Sanskrit Book Depot., Kachauri Gali, Benaras City, India.

[4] Arya, M., 2019. Structure of Pāṇiniyan Grammar (Aṣṭādhyāyī). International Multidisciplinary Research Journal - ISSN 2424-7073 Language Studies – Volume 2, IssueNo.1: Pages 15-20.

[5] Anoop, C. S., Ramakrishnan, A. G., 2023. Suitability of syllable-based modeling units for end-to-end speech recognition in Sanskrit and other Indian languages. Expert Systems with Applications, Volume 220, 119722, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2023.119722.

[6] Singh, M., Kumar, R., Chana, I., 2020. Corpus-based Machine Translation System with Deep Neural Network for Sanskrit to Hindi Translation. Procedia Computer Science,Volume 167,Pages 2534-2544, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2020.03.306.

[7] Sandhan, J., Agarwal, A., Behera, L., Sandhan, T., Goyal, P., 2023. SanskritShala: A Neural Sanskrit NLP Toolkit with Web-based Interface for Pedagogical and Annotation Purposes. arXiv:2302.09527v1.

[8] Sarkar, S., Krishna, A., Goyal, P., 2023. Pre-annotation Based Approach for Development of a Sanskrit Named Entity Recognition Dataset. Proceedings of the Computational Sanskrit and Digital Humanities, 18th World Sanskrit Conference, pages 59-70.

[9] Tripathi, J. N., 2015. Statistical Studies of Printed Devnagri (Hindi) Text. IETE Journal of Research, 17:1, 25-27, DOI: 10.1080/03772063.1971.11486712.

[10] Bharati, A., Kulkarni, A., Nair, S. S., 2008. Use of Amarako´sa and Hindi WordNet in Building a Network of Sanskrit Words. Proceedings of ICON-2008: 6th International Conference on Natural Language Processing, Macmillan Publishers, India.

[11] Nair, S.S., Kulkarni, A., 2010. The Knowledge Structure in Amarakośa. Jha, G.N. (eds) Sanskrit Computational Linguistics. ISCLS 2010. Lecture Notes in Computer Science, vol 6465. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-17528-2_13.

[12] Nair, S.S., Kulkarni, A., 2011. The Knowledge Structure in Amarakośa . Department of Sanskrit Studies, School of Humanities, University of Hyderabad Hyderabad 500046.

[13] Jha, P., Kumar, R., Sahula, V., 2023. Filtering and Extended Vocabulary based Translation for Low-resource Language pair of Sanskrit-Hindi. ACM Trans. Asian Low-Resour. Lang. Inf. Process. https://doi.org/10.1145/3580495.

[14] Sitender., Bawa, S., 2018. SANSUNL: A Sanskrit to UNL Enconverter System. IETE Journal of Research, 67:1, 117-128, DOI: 10.1080/03772063.2018.1528187.

[15] Dikshita, B., 1969, Vaiyakarana-Siddhanta-Kaumudi, ChowkhambaSanskrit Series Office, Varanasi, India.

[16] Tripathi, J. S. L., Malaviya, S., 1986, Kasika: A Commentary on Pāṇini's Grammar of Vamana and Jayaditya, Volume 1, Tara Printing Works, Varanasi, India.

[17] Palsule, G. B., 1955. A Concordance of Sanskrit Dhatupathas (with Index of meanings). The Bulletin of the Deccan College Research Institute, XV Nos 1-2, Poona, India.

[18] Pakhira, M. K., 2014. A Linear Time-Complexity K-means Algorithm Using Cluster Shifting. International Conference on Computational Intelligence and Communication Networks, Bhopal, India, pp. 1047-1051, doi: 10.1109/CICN.2014.220.

[19] Zhang, J., Fei, J., Song, X., Feng, J., 2021. An Improved Louvain Algorithm for Community Detection, Hindawi, Mathematical Problems in Engineering Volume 2021, Article ID 1485592, 14 pages https://doi.org/10.1155/2021/1485592.

[20] Blondel, V. D., Guillaume, J. L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. arXiv:0803.0476v2.

[21] Ajit Krishnan, Amara-quiz. January, 2022. [dataset], https://raw.githubusercontent.com/aupasana/amara-quiz/master/database/amara_pada.csv

[22] Ingrassia, S., Punzo, A., 2020. Cluster Validation for Mixtures of Regressions via the Total Sum of Squares Decomposition. J Classif37, 526–547 (2020). https://doi.org/10.1007/s00357-019-09326-4.

[23] Baptista, M. L., Henriques, E. M. P., Goebel, K. 2021. More effective prognostics with elbow point detection and deep learning. Mechanical Systems and Signal Processing, Volume 146, 106987, ISSN 0888-3270, https://doi.org/10.1016/j.ymssp.2020.106987.

[24] Lambiotte, R., Delvenne, J. C., Barahona, M. 2009. Laplacian Dynamics and Multiscale Modular Structure in Networks, arXiv:0812.1770.

[25] Hellwig, O., Nehrdich, S., Sellmer, S. 2023. Data-driven dependency parsing of Vedic Sanskrit. Language Resources & Evaluation, Springer,https://doi.org/10.1007/s10579-023-09636-5.

[26] Mahesh, A. V. S. D. S., Bhattacharya, A., 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.