Web-Based Sanskrit Verb Thesaurus

A Project Report Submitted

in Partial Fulfilment of the Requirements

for the Degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Aditya Srinivas Menon - 2020BCS0008

Aditya Garg - 2020BCS0086

Aditya Tyagi - 2020BCS0126

Tarun Boya - 2020BCS0180

to

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY KOTTAYAM

KERALA - 686635, INDIA

April 2024

DECLARATION

We, **Aditya Srinivas Menon** (**Roll. no. 2020BCS0008**), **Aditya Garg** (**Roll. no. 2020BCS0086**), **Aditya Tyagi** (**Roll. no. 2020BCS0126**), **Tarun Boya** (**Roll. no. 2020BCS0180**), hereby declare that, this report entitled **'Web-Based Sanskrit Verb Thesaurus'** submitted to Indian Institute of Information Technology Kottayam towards partial requirement of **Bachelor of Technology** in **Computer Science and Engineering** is an original work carried out by us under the supervision of **Dr. Nandini J Warrier** and has not formed the basis for the award of any degree or diploma, in this or any other institution or university. We have sincerely tried to uphold academic ethics and honesty. Whenever an external information or statement or the result is used then, that have been duly acknowledged and cited.

Kottayam-686635
April 2024

Aditya Srinivas Menon - 2020BCS0008
Aditya Garg - 2020BCS0086
Aditya Tyagi - 2020BCS0126
Tarun Boya - 2020BCS0180

CERTIFICATE

This is to certify that the work contained in this project report entitled **"Web-Based Sanskrit Verb Thesaurus"** submitted by **Aditya Srinivas Menon** (**Roll. no. 2020BCS0008**), **Aditya Garg** (**Roll. no. 2020BCS0086**), **Aditya Tyagi** (**Roll. no. 2020BCS0126**), **Tarun Boya** (**Roll. no. 2020BCS0180**) to the Indian Institute of Information Technology Kottayam towards partial requirement of **Bachelor of Technology** in **Computer Science and Engineering** has been carried out by them under my supervision and that it has not been submitted elsewhere for the award of any degree.

Kottayam-686635
April 2024

Dr. Nandini J Warrier
Project Supervisor

ACKNOWLEDGMENT

We would like to extend our sincerest thanks to our guide Dr. Nandini J Warrier for her mentorship and continued support throughout the entire research process. We would like to thank the members of my panel, Dr. Vengadeswaran S, Dr. Manu Madhavan, and Dr. Selvi C for their time, expertise, and constructive feedback during the project review. We are grateful to Indian Institute of Information Technology Kottayam for providing the facilities and resources needed to conduct this research. We also want to acknowledge our classmates who shared their knowledge and experiences, fostering a collaborative research environment that enhanced the quality of this work.

Aditya Srinivas Menon (Roll No. 2020BCS0008)
Aditya Garg (Roll No. 2020BCS0086)
Aditya Tyagi (Roll No. 2020BCS0126)
Tarun Boya (Roll No. 2020BCS0180)

ABSTRACT

Sanskrit, an Indo-Aryan language with deep roots in the linguistic heritage of the Indian subcontinent, stands as a culturally significant and intricate language. Recognizing the vital role it plays, this work aims to use the Sanskrit Verb Thesaurus, "Kriyanighantu," compiled by Manoj VR to build a web-based Sanskrit Verbnet. Through the development of a comprehensive online resource, this project aims to enhance the accessibility and understanding of Sanskrit, bridging the gap between tradition and technology. We also compare the Verbnet with the existing Sanskrit wordnet and provide quantitative and qualitative advantages of our approach over theirs. This project aims to make a substantial contribution to Sanskrit linguistics and language comprehension in modern digital situations.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Sanskrit, an Indo-Aryan language, is one of the oldest languages of the Indian subcontinent. The Sanskrit language is the mother tongue of all Indo European languages spoken in India. A lot of Indian manuscripts, for over 2,000 years, were written in Sanskrit making it a *heritage language* [1]. Thus, being able to understand Sanskrit and build morphological tools around it is important. The task of understanding and translating the ancient manuscripts from Sanskrit to English and other Indian Languages is an ongoing task. Throughout its two millennia of development, Sanskrit grammar has continuously changed as new meanings have been added to its base vocabulary. Sanskrit scholars have had to spend a great deal of time reading many texts in their original context due to the language's complexity. To tackle this intricacy, the development of online resources for understanding Sanskrit grammar would be useful.

1.2 Project Background

The vast system of Sanskrit verbology encompasses a huge library of root words, allowing for the creation of a myriad of combinations. The Sanskrit Verb Thesaurus, "Kriyanighantu" [3], compiled and published by Manoj VR documents over 1,993 of these root words and their meanings. This serves as a valuable resource for Sanskrit enthusiasts seeking appropriate verbs and their meanings. Given the evolving nature of Sanskrit, where many words accrue new meanings over time, having a comprehensive Verb Thesaurus becomes crucial for grammarians aiming to grasp the original intent of texts. Spanning 664 pages, the thesaurus provides an extensive collection of verbs, complete with forward and backward links for enhanced accessibility. Manoj VR's dedication to Sanskrit research is widely praised. An example from the book is shown in Figure 1.1, two different words have multiple meanings of their own, while one of the meanings match with each other.

Having a web-based platform for querying the verbs and its relations mentioned in the book offers distinct advantages. Firstly, it enhances accessibility and convenience by eliminating the necessity for physical books. The web interface also enables Sanskrit linguists to search, enabling swift access to verbs mentioned in the book. It not only promotes efficiency but also contributes to cost savings, eliminating the need for expenses associated with buying physical books.

Moreover, the web interface will act as a valuable tool for Sanskrit grammarians, aiding in the interpretation and analysis of Sanskrit grammar. Beyond its linguistic utility, this digital medium plays a crucial role in the preservation of cultural heritage. It becomes an invaluable resource in pro-

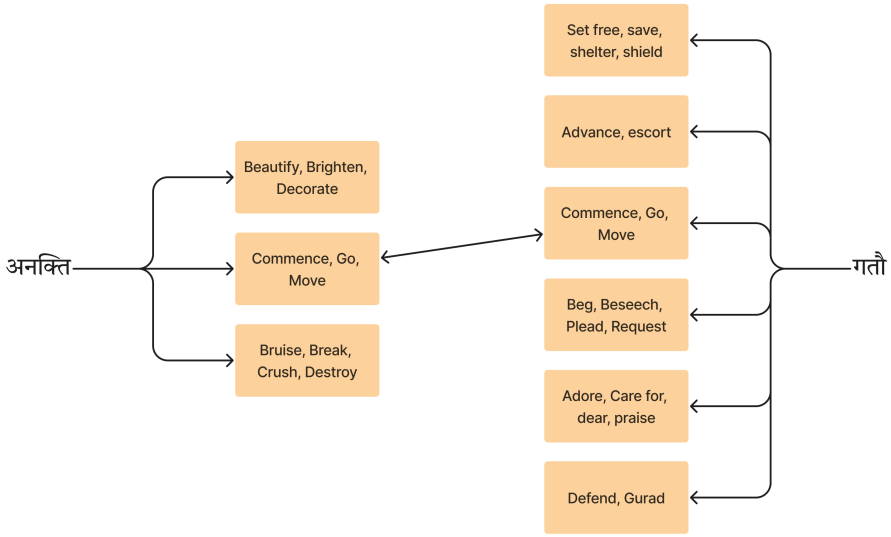


Figure 1.1: Example of two words that share a common meaning from the book Kriyanighantu

tecting and conveying the rich cultural legacy in Sanskrit literature by better understanding of ancient writings.

1.3 Problem Statement

This project aims to make a web-based Verbnet (Wordnet [5] with verbs) using the book. Although there exists a Sanskrit wordnet [1], it contains Nouns, Adverbs and Adjectives and give less importance to the rich Verbology of Sanskrit. Moreover, existing Sanskrit wordnet has limited number of words. To solve this problem we leverage the book to build a online, web-based version Sanskrit verb thesaurus. We hope to make Sanskrit more

accessible by building the Verbnet.

The rest of the report is organized as follows. A survey on existing approaches and the proposed approach is mentioned in Chapter 2. An extensive study on different database types and data schema is mentioned in Chapter 3. More information about the Verbnet and the website is mentioned in Chapter 4. The future course of the project and conclusion is mentioned in Chapter 5, followed by bibliography and references.

Chapter 2

Existing Approaches

2.1 Overview of WordNet

A wordnet is a database of lexicons that are oranized in terms of their meaning and semantic relationships [5]. It is a lexical database that establishes relations between words by connecting them via synonyms, hyponyms, and meronyms. A synonym is a pair of words that mean exactly or nearly the same. A hyponym is a relation where one word is a part of a set that is defined by the other word. A meronym defines a part-of reltionship between two words. The synonyms alone are grouped into a set called synset, with meaning and example usage.

A wordnet is usually made accessible via a web browser with an user interface. It is mainly used in text analysis and building AI applications.

2.1.1 Advantages of a Wordnet

The advantages of WordNet are mainly associated with various aspects of natural language processing (NLP) and linguistic research. One key benefit lies in Word Sense Disambiguation (WSD), where WordNet provides a comprehensive inventory of words and their senses. For example, the word "bark" in English can mean a tree bark or a animal sound, which is a dog bark.

Moreover, WordNet excels in encoding relationships between words, synonyms, antonyms, hyponyms (more specific terms), and hypernyms (more general terms).

In the realm of information retrieval, WordNet serves as a potent tool by expanding query terms to include synonyms and related words. This expansion significantly improves the effectiveness of information retrieval systems, providing a elevated search experience.

Overall, WordNet stands as a general resource for NLP research, offering a robust foundation for linguistic analysis and comprehension. Its multifunctional capabilities make it a cornerstone in the field, supporting advancements in linguistics.

2.2 Wordnet for English

The first attempt to make a wordnet for the English language was successfully done by Princeton University [5]. It was released under a *BSD Lisence*.

The Open English Wordnet (OEWN) [4] is an open-source alternative to the Princeton Wordnet which contained modern words and phrases. The data

for OEWN is stored locally in the users browser memory when its loaded, making it extremely fast. The backend is written in the Rust programming language which also helps in the inference speed as Rust is a high-performance programming language that emphasises on speed and efficiency.

2.3 Wordnet for Sanskrit

A few efforts have been taken to build a wordnet in Sanskrit. These efforts have been studied and discussed below:

2.3.1 Sanskrit Wordnet

The development of the Sanskrit Wordnet [1] can be traced back to the expansion of the Hindi Wordnet [6]. Initially derived from the English Wordnet, the Hindi Wordnet served as the foundational source for its Sanskrit counterpart. The hierarchy preservation principle (HPP) is followed in the Sanskrit Wordnet, which means that the hierarchy of the Hindi Wordnet is used as a reference to establish semantic relations for the synsets of Sanskrit Wordnet.

Since Sanskrit is a gendered language, all the nuances with respect to gender are included in the synsets.

The Sanskrit Wordnet, despite its merits, presents several notable disadvantages. The distribution of POS tags have been mentioned in Table 2.1. It is to be noted that there are a large number of Nouns and very few Verbs. Sanskrit is a language that is known for its rich Verbology and this wordnet does not inlcude all the Verbs in Sanskrit, making it difficult to use.

Additionally, the User Interface (UI) of the Sanskrit Wordnet requires

enhancements and the use of an outdated database (text files as backend) and aging technology poses challenges in terms of scalability, efficiency, and compatibility with current standards. The paper fails to go into the details of how the wordnet was built. Additional information on the wordnet itself would have been helpful in reproducability.

Lastly, the paper overlooks the practical applications of the Sanskrit Wordnet, particularly in the domain of natural language processing tasks. Insights into how the resource can be leveraged in practical scenarios, such as machine translation or information retrieval, would significantly enhance its value and promote broader adoption within the field of computational linguistics.

Noun	Verb	Adverb	Adjective
17413	1246	263	3990

Table 2.1: Distribution of POS tags in the Sanskrit Wordnet

2.3.2 JNU Tanalyser

The JNU Tanalyser[1] is another approach that aims to build a Sanskrit wordnet. Instead of collecting data, it uses morphological analysis to group words into synsets. The backend of this Sanskrit verb analyzer system is supported by lexical resources stored in the form of data files. Following a multi-tier architecture, the system involves user interaction, Apache-tomcat, Java servlet, and data files, ensuring a structured and layered approach to its functionality.

[1]http://sanskrit.jnu.ac.in/tanalyzer/tanalyze.jsp

However, the JNU Tanalyser is not without its limitations. Firstly, its architecture relies on older tools and frameworks, potentially affecting its overall efficiency. Moreover, a notable drawback is its inconsistent functionality, as the website does not work reliably most of the time.

Furthermore, the morphological analysis aspect of the system appears to have limitations, leading to the identification of a relatively small number of verbs. This stands in contrast to the extensive coverage of verbs available in a "Kriyanighanthu".

2.3.3 Other approaches

Many of the approaches from the years 2000-2010 that try to build Sanskrit wordnet have been discontinued or ended in failure. Some examples like the Inria Sanskrit Wordnet[2] and the University of Hyderabad Wordnet[3] have been discontinued and the websites do not function anymore. This is a huge issue as Sanskrit grammarians cannot find a working online thesaurus for verbs that is accessible and encompasses a wide variety of verbs from the language.

2.4 Proposed approach

The project to enhance Sanskrit language understanding involves a systematic five-step process. The overall framework of the proposed approach has been shown in Figure 2.1.

[2]https://sanskrit.inria.fr/DICO/grammar.html
[3]http://sanskrit.uohyd.ernet.in/

Step 1: Digitalization: The initial phase focuses on converting the existing Sanskrit verb thesaurus, authored by Manoj VR, into a digital format. This step involves transforming from traditional book based approaches to a more modern digital format.

Step 2: Database Schema: Following digitalization, a database schema is creaated based on the structure of the data collected. This framework is essential for organizing and categorizing the verbs and understanding relations between them and how to show it using data structures.

Step 3: Data Conversion: With the database schema in place, the next step involves populating the database with the content from the Sanskrit verb thesaurus.

Step 4: User-Friendly Interface: To enhance accessibility, a user-friendly interface is developed in the fourth step. This frontend facilitates intuitive verb retrieval, offering users a seamless experience in navigating and exploring the online Sanskrit verb thesaurus.

Step 5: Document Search: The final step introduces a powerful feature – document search. Users can input a Sanskrit document into the application, and the system will annotate all relevant verbs. This advanced functionality aligns with the project's goal of transforming complex Sanskrit into an accessible, digital resource, allowing for efficient analysis of verbs within the context of diverse texts.

In summary, the project aims to revolutionize Sanskrit language understanding by seamlessly transitioning from traditional to digital resources, bridging the gap between age-old linguistic traditions and modern technology. Through these strategic steps, the project aspires to make Sanskrit

verbology more accessible, preserving and enhancing the understanding of this culturally rich language.

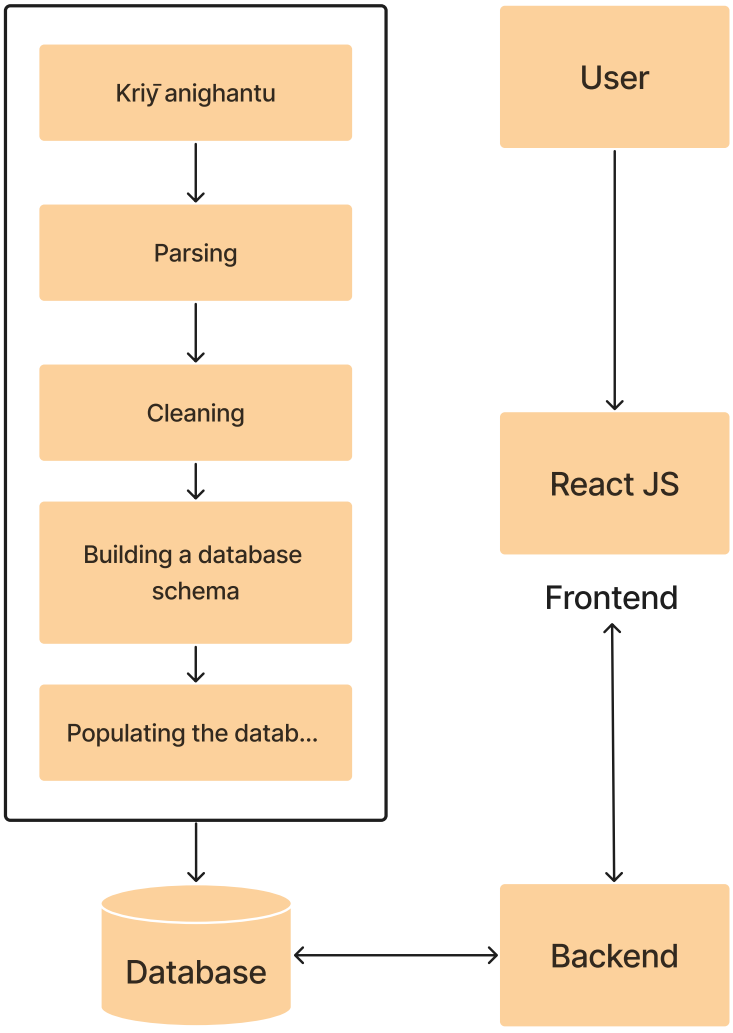


Figure 2.1: Framework of the proposed approach

Chapter 3

Database Evaluation

The next stages of the project included creating a database schema and choosing the right type of database for the Verbnet. The goal was to compare between SQL (e,g,. MySql), NoSQL (e,g,. MongoDb), Graph databases (e,g,. Neo4J) and Distributed filesystems (e,g,. Hadoop, Pig and Hive). This section goes more into detail of how the database types were chosen and the metrics tracked.

3.1 Literature Survey

To understand more about the metrics used to compare databases we performed literature survey.

Performance Comparison of SQL and NoSQL Databases [2]

The paper talks about the performance difference in SQL and NoSQL databases. The study focuses on several key metrics to evaluate their efficiency. These

metrics include the time required to instantiate a database bucket, read values corresponding to given keys from the bucket, write key-value pairs to the bucket, and fetch all the keys in the bucket. The objective is to provide a understanding of how these databases perform in real-world scenarios.

One notable finding is the absence of an observable correlation between the chosen data schema and overall performance. This will help in achieving a suitable schema for a given data.

The evaluation results reveal that the read performance of SQL surpasses that of some NoSQL databases, although not consistently across the entire NoSQL spectrum. This comparison sheds light on the specific strengths and weaknesses of each type of database in handling key-value stores, offering valuable insights for selecting a database for various applications.

A comparison between several NoSQL databases [8]

Similar to the previous paper this work conducts a comprehensive analysis by comparing various NoSQL databases, delving into both qualitative and quantitative aspects. The primary emphasis lies on the comparison process itself and an exploration of the features unique to each database.

When it comes to qualitative aspects, the paper focusses on persistence, replication, high availability, transactions, rack-locality awareness, implementation language, influences/sponsors, and license type of the databases. These give a high-level view of the database which might affect the decision in choosing it over the other possibilities

Quantitative evaluations further perform comparative study, focusing on metrics such as the size of data, throughput, and latency. These quantitative

benchmarks aim to quantify the performance and efficiency of each NoSQL database.

It is noteworthy, however, that the paper refrains from explicitly stating the results of these evaluations.

3.2 Choosing a database

The utilization of the data provided in the book encountered licensing issues, posing challenges in incorporating it for the current review. While the data was being processed, we decided to use the English wordnet to test the databases. The data was downloaded in the form of *xml* files from the Open English Wordnet (OEWN) [4]. The data is then converted to *json* format as given in Listing 3.2.

Each type of database will be loaded with the data as mentioned in Listing 3.2 and compared with each other based on these metrics:

- No. of nodes
- No. of relationships between nodes
- Time taken to add nodes
- Time taken to add relationships
- Time taken to query a single node, and
- Time to query children nodes of a node for up to depth N
- Throughput for each operation

- Latency for each operation

The data ended up having 10,000 nodes and 7,000 relationships between them, which is similar to the number of nodes and relationships from the book.

Listing 3.1: Data format of the OEWN data

```
[
    {"id": "oewn-theta wave-n",
    "lemma": "theta wave"
    pos": "n" ,
    "synsets ": ["oewn-theta rhythm-n",
                 "oewn-theta wave-n"]
    },
    {
    "id": "oewn-thunderbolt-n" ,
    "lemma": "thunderbolt",
    "pos": "n"
    "synsets" : ["oewn-bolt-n",
                 "oewn-bombshell-n",
                 "oewn-thunderclap-n"]
    }
]
```

Database	Add data	N Count	R Count	Throughput	Latency
Unit	(s)	(s)	(s)	(ops/s)	(ms)
SQLite3	0.102867	0.739	1.955	3890.82	0.26
Neo4J	124.22	0.043	0.047	0.00805	124220
PIG on HDFS	233	38.739	39.957	0.009524	105000

Table 3.1: Metrics for loading and querying data

Query Depth (N)	**SQL Time (s)**	**Neo4J Time (s)**
1	0.131	0.026
2	0.195	0.029
3	0.483	0.03
4	2.667	0.04
5	14.041	0.135
6	86.190	0.153
7	452.405	0.165
8	1285.007	0.169
50	-	0.417
100	-	0.47
200	-	0.45
2000	-	0.466

Table 3.2: Time to query children nodes of a node for up to depth N for SQL and Neo4J

3.2.1 Methodology

The work focuses on SQL (Python sqlite3 version 2.6.0), graph databases (neo4j) and distributed file systems (HDFS using PIG) as the main database types. For all experiments, we use a virtual machine with 128 GB storage and 8 GB RAM.

3.2.2 Results

Table 3.1 shows the time taken to add nodes and relationships, get node count and get relationships count from the different databases. It is noticed that SQL and PIG on HDFS performs well with respect to loading data and relationships. Whereas, Neo4J struggles to load just 10k entires. SQL again performs well while querying node count and relationship count, but Neo4J is able to do it in a fraction of a millisecond. PIG takes time as it has to query the nodes from a distributed datastore.

Table 3.2 presents the time taken to query the top N children of the nodes. It is observed that for small values of N, the SQL query performance is comparable to that of Neo4J. However, as N increases, the time taken to query the children in Neo4J stabilizes at approximately 0.450 seconds, whereas for SQL, it exhibits exponential growth. Graphs illustrating these metrics are depicted in Figure 3.1 and Figure 3.2, corresponding to the data presented in Table 3.2.

PIG, on the other hand, demonstrates performance similar to SQL. However, it incurs longer processing times due to the distributed nature of the data storage, requiring querying across multiple distributed datanodes.

3.3 Conclusion

From the results, it is evident that Graph databases, such as Neo4j, demonstrate efficient performance in querying nodes within a graph structure. Once populated, these graph databases serve as shared resources accessible to multiple users. Service providers like Neo4j offer comprehensive tools for deploy-

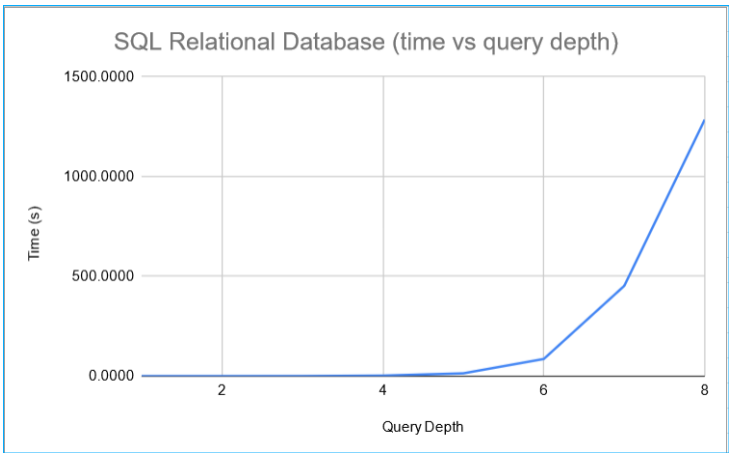


Figure 3.1: Time to query children nodes of a node for up to depth N for SQL for different values of N

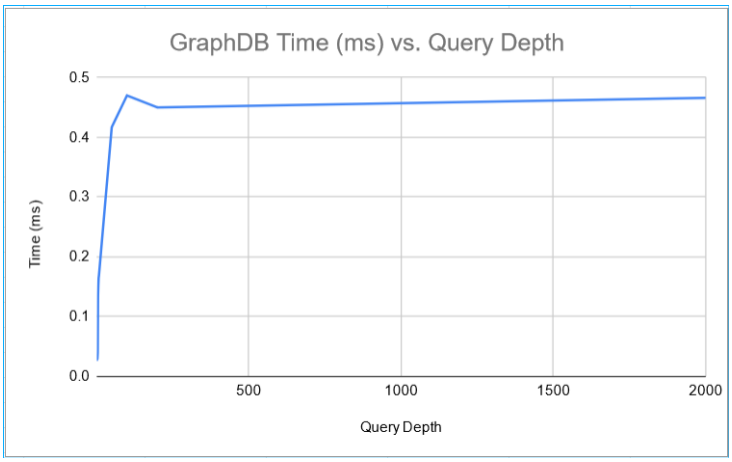


Figure 3.2: Time to query children nodes of a node for up to depth N for Neo4J for different values of N

ing graph databases and monitoring their operations. Additionally, these services facilitate analytical insights aimed at enhancing search capabilities based on user-generated data.

Chapter 4

Verbnet

This chapter gives more information on the Verbnet, it's schema, the website, and the system diagram for the service.

4.1 Schema

The book comprises a total of 2000 verbs and their respective meanings. Synonyms are words that mean the same thing. Therefore, if two verbs point to the same meaning in the book, they are synonyms. The meanings of each word are mentioned separately in alphabetical order and are linked to its verb using the *"look_up"* attribute. We define difference schema for the verbs and the meanings. The schema for verbs is mentioned in schema 4.1. Each verb will have a *"unique_id"* which will act as an identifier. The *"lemma"* will have the verb itself and *"adverb"* contains the English adverb. All the English meanings are listed in the *"english_meaning"* attribute. The meaning of the verb in Sanskrit is given in the *"look_up"* attribute and a list

of other linked verbs are given in the *"see_also"* section.

Listing 4.1: Data schema for verbs from Kriyanighanthu

```
[
  {
    "id": "unique_id",
    "lemma": "word_in_sanskrit",
    "adverb": "adverb",
    "english_meaning": [
      "meaning_1",
      "meaning_2"
    ],
    "look_up": [
      "look_up_1",
      "look_up_2"
    ],
    "see_also": [
      "see_also_1",
      "see_also_2"
    ]
  }
]
```

Each *look_up* is a Sanskrit meaning that is defined by the schema 4.2. Each meaning is given a *"unique_id"* and the Sanskrit meaning is given in *"lemma"*. All the English meanings are listed in the *"english_meaning"*

attribute and all the verbs that share the same meaning are mentioned in *"members"* attribute.

Listing 4.2: Data schema for verb meanings from Kriyanighanthu

```
[
  {
    "id": "unique_id",
    "lemma": "sanskrit_meaning",
    "english_meaning": [
      "meaning_1",
      "meaning_2"
    ],
    "members": [
      "unique_id_1",
      "unique_id_2"
    ],
  }
]
```

4.2 Database

As extrapolated from the findings delineated in Section 3, we use a graph database for our Verbnet. We use neo4j graph database to load, store, host, and share the Sanskrit verbs. Due to copyright issues on using the data from Kriyanighantu, we use the publicly available Sanskrit WordNet data from PyIWN [7] for the initial version of the Sanskrit Verbnet. The words are

first extracted and stored in CSV files and later loaded into neo4j using the cypher commands mentioned in Listing 4.3.

The *nodes.csv* CSV file for the nodes contain the following columns: *"id"*, *"lemma"* for the Sanskrit word and *"pos"* for the part of speech tag. The *relationship.csv* CSV file for the relationships between nodes contain a *"start_id"* and *"end_id"* columns that link two nodes with each other.

Listing 4.3: Cypher command for loading nodes

```
1 LOAD CSV WITH HEADERS FROM 'nodes.csv' AS row
2 CREATE (:Node {id: row.id, lemma: row.lemma, pos: row.
    pos})
```

The cypher command mentioned in Listing 4.4 was used to load the relationship data.

Listing 4.4: Cypher command for loading realtionships

```
1 LOAD CSV WITH HEADERS FROM 'relationships.csv' AS row
2 MATCH (start:Node {id: row.start_id})
3 MATCH (end:Node {id: row.end_id})
4 CREATE (start)-[:RELATED_TO]->(end)
```

4.3 System diagram for Verbnet

Figure 4.1 gives the system diagram for the Sanskrit Verbnet. The user interface is made using HTML, CSS and Javascript. The frontend is brought together by the React framework in the frontend. In the backend, the Sanskrit Verbnet uses a NodeJS server that talks to the Neo4J database server

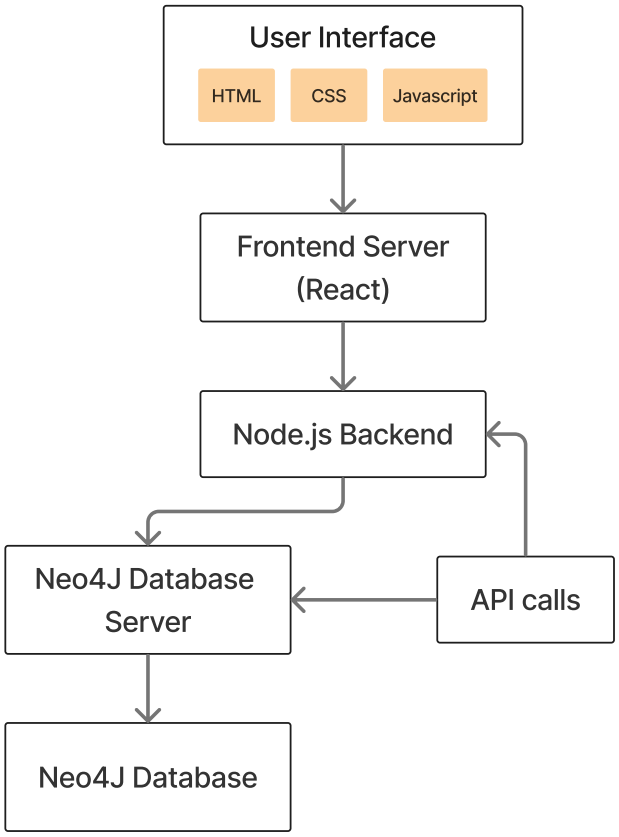


Figure 4.1: System diagram for the Sanskrit Verbnet

using API calls. All the data is stored in the Neo4J database as nodes (Verbs) and relations (Synonyms).

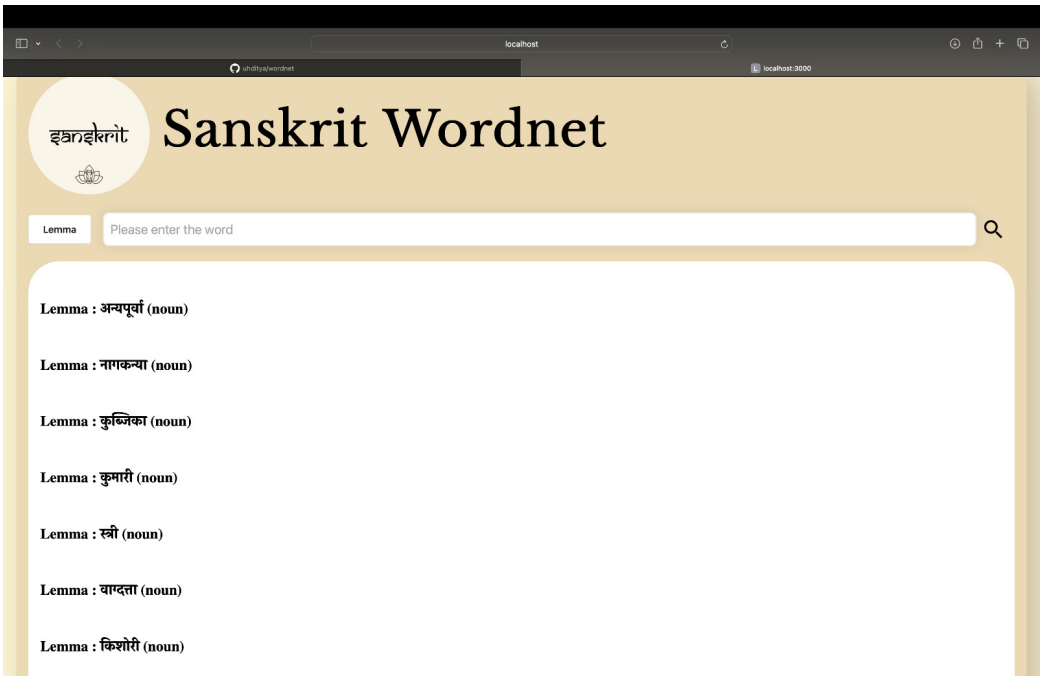


Figure 4.2: User Interface for the Sanskrit Verbnet

4.4 Website and UI

The User Interface was made using HTML, CSS and Javascript. Figure 4.2 shows the landing page of the website. Figure 4.2 shows the results when the word was searched.

4.5 Functionalities

The Sanskrit Verbnet provides the following functionalities:

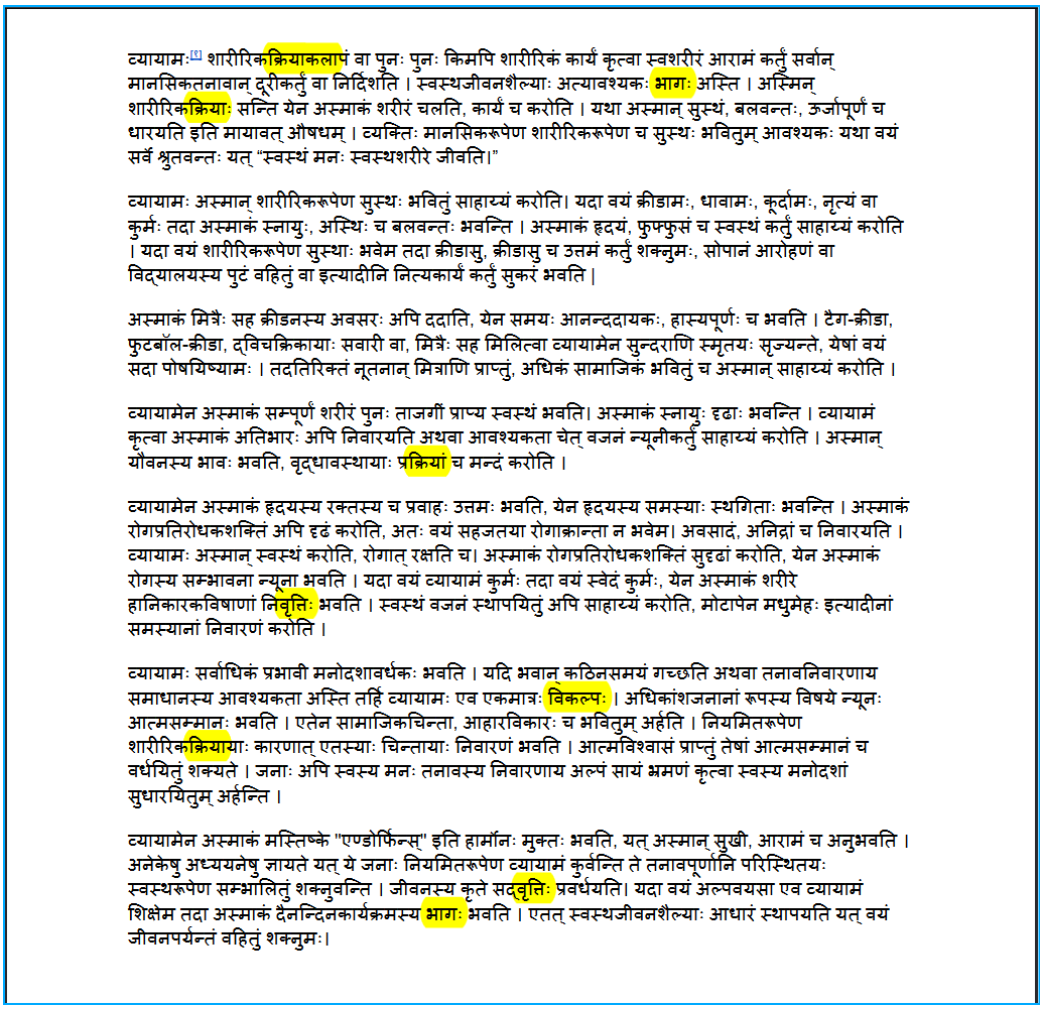
- Upon receiving a word input from the user, the system will employ the **Lemma Search** functionality to retrieve synonyms and present them accordingly.

- In response to a given search query word, this Verbnet uses an algorithmic approach to highlight synonyms within a provided PDF document containing Sanskrit words using the **PDF Search** functionality. The resulting output will be a modified PDF file, where all identified synonyms corresponding to the search query are visually distinguished.

This method facilitates efficient retrieval and comprehension of synonymous terms within Sanskrit texts, aiding scholars and researchers in their linguistic analyses and studies.Figure 4.3 illustrates the functionality of the **PDF Search** feature. Upon querying for the term in कर्म, which translates to "deed" or "work" in English, and providing an input PDF containing the Wikipedia article on कर्म [1], the output PDF highlights the following words:

[1]Wikipedia article on कर्म

कला, विकल्पः, भागः, वृत्तिः, and क्रिया. These terms convey a similar meaning to the searched word, thereby facilitating efficient information retrieval.



व्यायामः[1] शारीरिकक्रियाकलापं वा पुनः पुनः किमपि शारीरिकं कार्यं कृत्वा स्वशरीरं आरामं कर्तुं सर्वान् मानसिकतनावान् दूरीकर्तुं वा निर्दिशति । स्वस्थजीवनशैल्याः अत्यावश्यकः भागः अस्ति । अस्मिन् शारीरिकक्रियाः सन्ति येन अस्माकं शरीरं चलति, कार्यं च करोति । यथा अस्मान् सुस्थं, बलवन्तः, ऊर्जापूर्णं च धारयति इति मायावत् औषधम् । व्यक्तिः मानसिकरूपेण शारीरिकरूपेण च सुस्थः भवितुम् आवश्यकः यथा वयं सर्वे श्रुतवन्तः यत् "स्वस्थं मनः स्वस्थशरीरे जीवति।"

व्यायामः अस्मान् शारीरिकरूपेण सुस्थः भवितुं साहाय्यं करोति। यदा वयं क्रीडामः, धावामः, कूर्दामः, नृत्यं वा कुर्मः तदा अस्माकं स्नायुः, अस्थिः च बलवन्तः भवन्ति । अस्माकं हृदयं, फुफ्फुसं च स्वस्थं कर्तुं साहाय्यं करोति । यदा वयं शारीरिकरूपेण सुस्थाः भवेम तदा क्रीडासु, क्रीडासु च उत्तमं कर्तुं शक्नुमः, सोपानं आरोहणं वा विद्यालयस्य पुटं वहितुं वा इत्यादीनि नित्यकार्यं कर्तुं सुकरं भवति ।

अस्माकं मित्रैः सह क्रीडनस्य अवसरः अपि ददाति, येन समयः आनन्ददायकः, हास्यपूर्णः च भवति । टैग-क्रीडा, फुटबॉल-क्रीडा, द्विचक्रिकायाः सवारी वा, मित्रैः सह मिलित्वा व्यायामेन सुन्दराणि स्मृतयः सृज्यन्ते, येषां वयं सदा पोषयिष्यामः । तदतिरिक्तं नूतनान् मित्राणि प्राप्तुं, अधिकं सामाजिकं भवितुं च अस्मान् साहाय्यं करोति ।

व्यायामेन अस्माकं सम्पूर्णं शरीरं पुनः ताजगीं प्राप्य स्वस्थं भवति। अस्माकं स्नायुः दृढाः भवन्ति । व्यायामं कृत्वा अस्माकं अतिभारः अपि निवारयति अथवा आवश्यकता चेत् वजनं न्यूनीकर्तुं साहाय्यं करोति । अस्मान् यौवनस्य भावः भवति, वृद्धावस्थायाः प्रक्रियां च मन्दं करोति ।

व्यायामेन अस्माकं हृदयस्य रक्तस्य च प्रवाहः उत्तमः भवति, येन हृदयस्य समस्याः स्थगिताः भवन्ति । अस्माकं रोगप्रतिरोधकशक्तिं अपि दृढं करोति, अतः वयं सहजतया रोगाक्रान्ता न भवेम। अवसादं, अनिद्रां च निवारयति । व्यायामः अस्मान् स्वस्थं करोति, रोगात् रक्षति च। अस्माकं रोगप्रतिरोधकशक्तिं सुदृढां करोति, येन अस्माकं रोगस्य सम्भावना न्यूना भवति । यदा वयं व्यायामं कुर्मः तदा वयं स्वेदं कुर्मः, येन अस्माकं शरीरे हानिकारकविषाणां निवृत्तिः भवति । स्वस्थं वजनं स्थापयितुं अपि साहाय्यं करोति, मोटापेन मधुमेहः इत्यादीनां समस्यानां निवारणं करोति ।

व्यायामः सर्वाधिकं प्रभावी मनोदशावर्धकः भवति । यदि भवान् कठिनसमयं गच्छति अथवा तनावनिवारणाय समाधानस्य आवश्यकता अस्ति तर्हि व्यायामः एव एकमात्रः विकल्पः । अधिकांशजनानां रूपस्य विषये न्यूनः आत्मसम्मानः भवति । एतेन सामाजिकचिन्ता, आहारविकारः च भवितुम् अर्हति । नियमितरूपेण शारीरिकक्रियायाः कारणात् एतस्याः चिन्तायाः निवारणं भवति । आत्मविश्वासं प्राप्तुं तेषां आत्मसम्मानं च वर्धयितुं शक्यते । जनाः अपि स्वस्य मनः तनावस्य निवारणाय अल्पं सायं भ्रमणं कृत्वा स्वस्य मनोदशां सुधारयितुम् अर्हन्ति ।

व्यायामेन अस्माकं मस्तिष्के "एण्डोर्फिन्स्" इति हार्मोनः मुक्तः भवति, यत् अस्मान् सुखी, आरामं च अनुभवति । अनेकेषु अध्ययनेषु ज्ञायते यत् ये जनाः नियमितरूपेण व्यायामं कुर्वन्ति ते तनावपूर्णानि परिस्थितयः स्वस्थरूपेण सम्भालितुं शक्नुवन्ति । जीवनस्य कृते सद्वृत्तिः प्रवर्धयति। यदा वयं अल्पवयसा एव व्यायामं शिक्षेम तदा अस्माकं दैनन्दिनकार्यक्रमस्य भागः भवति । एतत् स्वस्थजीवनशैल्याः आधारं स्थापयति यत् वयं जीवनपर्यन्तं वहितुं शक्नुमः।

Figure 4.3: Annotated PDF with the synonyms of a given query highlighted within a provided PDF document

Chapter 5

Conclusion and Future Work

The optimization of web interface, with a primary focus on reducing retrieval time, stands as a pivotal task in our project. We want to ensure that the retrieval is efficient and querying words and their synsets happen in real-time.

The development of the Sanskrit Verbnet, as outlined in this report, represents a significant advancement in linguistic resources for Sanskrit scholars and researchers. By compiling a comprehensive database of Sanskrit verbs and their meanings, the Verbnet offers a valuable tool for studying Sanskrit language and literature.

Utilizing a graph database, specifically Neo4J, enhances the efficiency of storing, querying, and retrieving verb data. Additionally, the integration of publicly available resources, such as the data from Kriyanighantu, underscores the collaborative nature of linguistic research and resource development.

The system architecture, depicted in the system diagram, showcases a well-structured framework comprising frontend and backend components.

The user interface, developed using HTML, CSS, and JavaScript, ensures an intuitive and interactive experience for users.

Furthermore, the functionalities provided by the Sanskrit Verbnet, including Lemma Search and PDF Search, demonstrate its versatility and utility in aiding linguistic analysis and comprehension. The ability to identify synonyms within Sanskrit texts, as illustrated in the PDF Search feature, facilitates efficient information retrieval and enhances scholarly research in Sanskrit studies.

Overall, the Sanskrit Verbnet represents a valuable resource for scholars, educators, and enthusiasts of Sanskrit language and literature, providing comprehensive verb data and innovative functionalities to support linguistic analysis and research endeavors.

Bibliography

[1] Malhar Kulkarni, Chaitali Dangarikar, Irawati Kulkarni, Abhishek Nanda, and Pushpak Bhattacharyya. Introducing sanskrit wordnet.

[2] Yishan Li and Sathiamoorthy Manoharan. A performance comparison of sql and nosql databases. In *2013 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, pages 15–19, 2013.

[3] VR Manoj. - *Thesaurus of Synonymous Sanskrit Verbs*. Chinmaya International Foundation Shodha Sansthan, 2021.

[4] John P. McCrae, Alexandre Rademaker, Francis Bond, Ewa Rudnicka, and Christiane Fellbaum. English WordNet 2019 – an open-source WordNet for English. In Piek Vossen and Christiane Fellbaum, editors, *Proceedings of the 10th Global Wordnet Conference*, pages 245–252, Wroclaw, Poland, July 2019. Global Wordnet Association.

[5] George A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38:39–41, 1995.

[6] D Narayan, D Chakrabarti, P Pande, and P Bhattacharyya. An experience in building the indo wordnet-a wordnet for hindi. In *First international conference on global WordNet*, volume 24. Mysore, India, 2002.

[7] Ritesh Panjwani, Diptesh Kanojia, and Pushpak Bhattacharyya. pyiwn: A python based API to access Indian language WordNets. In *Proceedings of the 9th Global Wordnet Conference*, pages 378–383, Nanyang Technological University (NTU), Singapore, January 2018. Global Wordnet Association.

[8] Bogdan George Tudorica and Cristian Bucur. A comparison between several nosql databases with comments and notes. In *2011 RoEduNet International Conference 10th Edition: Networking in Education and Research*, pages 1–5, 2011.