International Journal Of Public Mental Health And Neurosciences
ISSN: 2394-4668
(An Official publication of Sarvasumana Association)

Extracting Invaluable Insights from Sushruta Samhita Using Natural Language Processing

Vaibhavi Jain
Department of Bioinformatics, MGM Institute of Biosciences & Technology
Chhatrapati Sambhajinagar, India
jvaibhavi722@gmail.com

Arpita Kharat
Department of Bioinformatics, MGM Institute of Biosciences & Technology
Chhatrapati Sambhajinagar, India
arpitakharat5188@gmail.com

Preenon Bagchi
Professor and Dean, Faculty of Engineering and Technology, Madhav University,
Rajasthan, India
prithish.bagchi@gmail.com

Shivani V.
Professor and Head. Department of Vyakarana.
Dean, Shastra Faculty. Karnataka Samskrit University. Bengaluru, India

Abstract

Sushruta Samhita, an ancient Sanskrit text dating back to around 600 BCE, stands as a cornerstone in the history of medicine, particularly in the domain of surgery. Despite its age, the text contains a wealth of knowledge and insights that are still relevant in contemporary medical practice. In this study, we employ Natural Language Processing (NLP) techniques to extract invaluable insights from the Sushruta Samhita. By leveraging NLP methodologies such as text parsing, entity recognition, and semantic analysis, we delve into the intricate details of surgical techniques, anatomical descriptions, disease classifications, and therapeutic approaches outlined in the text. Through computational analysis, we aim to unearth hidden patterns, correlations, and practical wisdom embedded within the text, shedding light on the timeless wisdom of ancient Indian medicine. Our findings not only contribute to a deeper understanding of Sushruta's contributions to the field of medicine but also demonstrate the potential of NLP in unlocking the knowledge contained within ancient texts for contemporary scientific inquiry and medical practice.

Keywords: **Ayurveda, Natural Language Processing (NLP), Samhita Samhita, Knowledge Extraction, Traditional Medicine, Healthcare, Sanskrit, Ancient Wisdom, Google colab, Python, etc.**

1. INTRODUCTION

Sushruta Samhita, an ancient Indian text dating back to the 6th century BCE, is a seminal work in the field of Ayurveda, detailing various aspects of medicine, surgery, and holistic healing practices. Composed by the legendary sage Sushruta, this comprehensive compendium encompasses a wealth of knowledge on anatomy, physiology, disease diagnosis, treatment modalities, and surgical procedures. Ayurveda, with its distinctive perspectives on social and spiritual life, is a science of life and well-being [1]. However, despite its profound significance, the vastness and complexity of Sushruta Samhita present challenges in comprehensively extracting and interpreting its valuable insights. Traditional methods of manual extraction and analysis are time-consuming, labor-intensive, and prone to errors, hindering the exploration of this rich source of medical wisdom. Within the Ayurvedic community, the Charak Samhita and the Susruta Samhita hold a position of pride and respect [2]. In recent years, advancements in Natural Language Processing (NLP) have revolutionized the way ancient texts are studied and understood. NLP techniques, powered by machine learning algorithms and linguistic models, offer a promising avenue for unlocking the latent knowledge embedded within Sushruta Samhita. By leveraging computational tools to process, analyze, and interpret textual data, researchers can extract meaningful insights, identify patterns, and gain a deeper understanding of Ayurvedic principles and practices elucidated in the ancient text. The goal of Ayurveda is to maintain a person's health and well-being by promoting balance in the body, mind, and environment [3].

In India, Sushruta is the most renowned doctor and surgeon. Many of his advances in surgery and medicine came before comparable findings in the West. Sushruta Samhita, also known as Sushruta's compendium, is a treatise that compiles Sushruta's teachings and works and is thought to be a component of Atharvaveda [4]. This paper proposes to elucidate the application of Natural Language Processing methodologies to extract and distill meaningful insights from Sushruta Samhita. Through a detailed exploration of NLP techniques, including text preprocessing, information retrieval, entity recognition, semantic analysis, and topic modeling, this study aims to unravel the intricate layers of knowledge encapsulated within the text. Natural language processing (NLP), an area of artificial intelligence that makes it easier for machines to understand and respond to human language, is the main source of power for digital assistants [5].

Furthermore, this research endeavor seeks to address specific challenges inherent in analyzing ancient texts like Sushruta Samhita, such as archaic language, semantic ambiguity, and cultural context. By employing specialized NLP techniques tailored to address these challenges, scholars can overcome barriers to comprehension and unlock the true depth of Ayurvedic wisdom encoded within the text. Moreover, the application of NLP to Sushruta Samhita not only facilitates the extraction of explicit medical knowledge but also enables the exploration of implicit insights, including cultural practices, socio-economic factors, and philosophical underpinnings embedded within the text. Our goal is to bridge the gap between traditional medicine and modern healthcare by extracting pertinent information from the text and presenting it in an intuitive interface by utilizing sophisticated natural language processing algorithms [6]. By contextualizing the extracted information within broader historical and cultural frameworks, researchers can gain holistic perspectives on ancient medical practices and their relevance to contemporary healthcare paradigms.

2. OBJECTIVES

1. Develop an Interactive query system using NLP-driven outputs to organize and store the extracted information systematically.
2. Employ NLP techniques to preprocess the vast textual content of Sushruta Samhita, including tokenization, stemming, and part-of-speech tagging.
3. Utilize NLP algorithms, particularly named entity recognition (NER), to identify and classify entities such as medicinal herbs, diseases, and treatment modalities mentioned in Sushruta Samhita.
4. To disseminate research findings through peer-reviewed publications, scientific conferences, and knowledge-sharing platforms, thereby fostering collaboration between traditional medicine practitioners, researchers, and healthcare professionals in the global fight against major Disease.

3. MATERIALS AND METHODOLOGY

Materials:

Sushruta Samhita is serves as a material for knowledge extraction where NLTK tool a natural language toolkit, is used for processing the text. For carrying out NLP-based knowledge extraction tasks from Sushruta Samhita, Google Colab offers a flexible and strong environment. It has the computational power, libraries, and collaboration tools required to support study and experimentation in this field. Utilizing NLP techniques to extract knowledge from Sushruta Samhita requires the use of Python programming. Scholars, practitioners, and researchers are enabled to examine, evaluate, and decipher the vast amount of information contained in old manuscripts such as Sushruta Samhita by means of its vast network of libraries and instruments.

Below the methodology were followed in this project:

Steps for Extracting meaningful insights from Sushruta Samhita as per given below:

1. ***Download Sushruta Samhita:***
   Sushruta Samhita’s soft copy were easily available on various platform of net. Downloading Sushruta Samhita from given website https://archive.org/details/sushruta-samhita**.**
2. ***Convert PDF to Image Folder of Sushruta Samhita:***
   - Use a PDF processing library such as PyMuPDF to extract pages from the PDF.
   - Convert each page to an image using a library like PyMuPDF (Fitz) or pdf2image.
3. ***Read Image from Folder and extract text of Sushruta Samhita.***

- Utilize an image processing library such as CV2 or OS module to read images from the folder.
- Apply Optical Character Recognition (OCR) or PIL (Python Imaging Library) techniques to extract text from images.
- Popular OCR libraries include Tesseract, Google Cloud Vision API, and Microsoft Azure Computer Vision.

4. ***Named Entity Recognition (NER) using NLTK:***

- Use NLTK (Natural Language Toolkit) to perform Named Entity Recognition.
- Train or utilize pre-trained models to identify entities such as Disease, Herbal remedy, Health being, etc., in the text.

5. ***Part of Speech (POS) Tagging***:

- Utilize NLTK or spaCy for Part of Speech tagging.
- These libraries offer pre-trained models to assign parts of speech (e.g., noun, verb, adjective) to words in the text.

6. ***Information Extraction:***

- Implement techniques such as pattern matching, rule-based systems, or machine learning algorithms to extract relevant information from the text.
- This could involve parsing sentences for specific patterns or utilizing NER outputs to extract relevant entities and their relationships.

7. ***Interactive Query System for Search Box:***

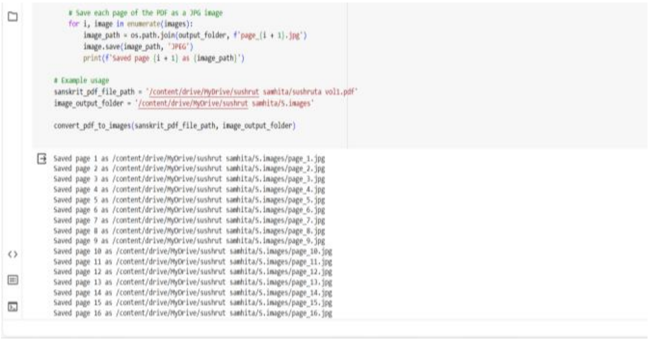
- Develop a user interface (UI) with a search box where users can input queries.
- Utilize a backend system that handles the queries and searches the processed text data for relevant information.
- Use techniques like keyword matching or advanced search algorithms to find herbal remedies or medicinal treatments mentioned in Sushruta Samhita.

4. RESULTS AND DISCUSSION:

Here are the result of NLP techniques used in our study to extract knowledge from Sushruta Samhita. Overall, leveraging NLP techniques to analyze the Sushruta Samhita provides valuable insights into the foundations of ancient Indian medicine, offering a glimpse into the rich heritage of healthcare practices that continue to influence contemporary medicine.

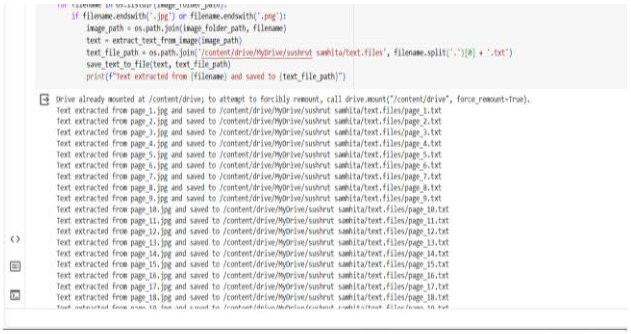
Output of Convert PDF to Image Folder of Sushruta Samhita:

By following these steps, we have successfully converted a PDF document of Sushruta Samhita into a folder containing images of each page. This process allows for easy access and manipulation of the content for further analysis or presentation.



```
    # Save each page of the PDF as a JPG image
    for i, image in enumerate(images):
        image_path = os.path.join(output_folder, f'page_{i + 1}.jpg')
        image.save(image_path, 'JPEG')
        print(f'Saved page {i + 1} as {image_path}')

# Example usage
sanskrit_pdf_file_path = '/content/drive/MyDrive/sushrut samhita/sushruta vol1.pdf'
image_output_folder = '/content/drive/MyDrive/sushrut samhita/S.images'

convert_pdf_to_images(sanskrit_pdf_file_path, image_output_folder)
```

```
Saved page 1 as /content/drive/MyDrive/sushrut samhita/S.images/page_1.jpg
Saved page 2 as /content/drive/MyDrive/sushrut samhita/S.images/page_2.jpg
Saved page 3 as /content/drive/MyDrive/sushrut samhita/S.images/page_3.jpg
Saved page 4 as /content/drive/MyDrive/sushrut samhita/S.images/page_4.jpg
Saved page 5 as /content/drive/MyDrive/sushrut samhita/S.images/page_5.jpg
Saved page 6 as /content/drive/MyDrive/sushrut samhita/S.images/page_6.jpg
Saved page 7 as /content/drive/MyDrive/sushrut samhita/S.images/page_7.jpg
Saved page 8 as /content/drive/MyDrive/sushrut samhita/S.images/page_8.jpg
Saved page 9 as /content/drive/MyDrive/sushrut samhita/S.images/page_9.jpg
Saved page 10 as /content/drive/MyDrive/sushrut samhita/S.images/page_10.jpg
Saved page 11 as /content/drive/MyDrive/sushrut samhita/S.images/page_11.jpg
Saved page 12 as /content/drive/MyDrive/sushrut samhita/S.images/page_12.jpg
Saved page 13 as /content/drive/MyDrive/sushrut samhita/S.images/page_13.jpg
Saved page 14 as /content/drive/MyDrive/sushrut samhita/S.images/page_14.jpg
Saved page 15 as /content/drive/MyDrive/sushrut samhita/S.images/page_15.jpg
Saved page 16 as /content/drive/MyDrive/sushrut samhita/S.images/page_16.jpg
```



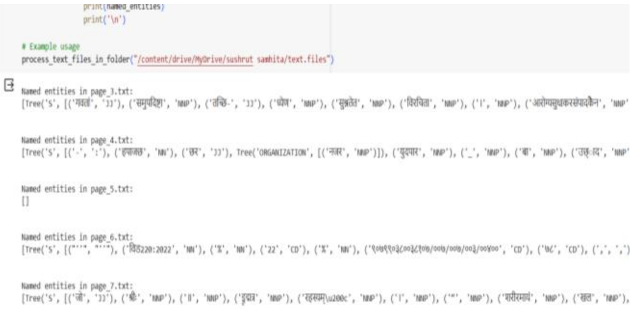
```
for filename in os.listdir(image_folder_path):
    if filename.endswith('.jpg') or filename.endswith('.png'):
        image_path = os.path.join(image_folder_path, filename)
        text = extract_text_from_image(image_path)
        text_file_path = os.path.join('/content/drive/MyDrive/sushrut samhita/text.files', filename.split('.')[0] + '.txt')
        save_text_to_file(text, text_file_path)
        print(f"Text extracted from {filename} and saved to {text_file_path}")
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
Text extracted from page_1.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_1.txt
Text extracted from page_2.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_2.txt
Text extracted from page_3.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_3.txt
Text extracted from page_4.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_4.txt
Text extracted from page_5.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_5.txt
Text extracted from page_6.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_6.txt
Text extracted from page_7.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_7.txt
Text extracted from page_8.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_8.txt
Text extracted from page_9.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_9.txt
Text extracted from page_10.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_10.txt
Text extracted from page_11.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_11.txt
Text extracted from page_12.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_12.txt
Text extracted from page_13.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_13.txt
Text extracted from page_14.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_14.txt
Text extracted from page_15.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_15.txt
Text extracted from page_16.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_16.txt
Text extracted from page_17.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_17.txt
Text extracted from page_18.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_18.txt
Text extracted from page_19.jpg and saved to /content/drive/MyDrive/sushrut samhita/text.files/page_19.txt
```

Output of Read Image from Folder and extract text of Sushruta Samhita:

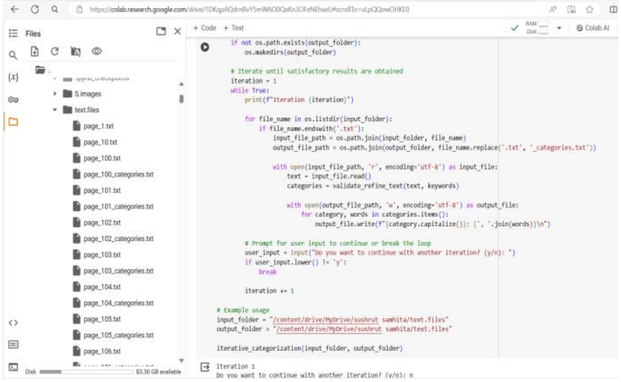
We read images from the folder containing Sushruta Samhita pages using the CV2 library. Effective techniques for loading, modifying, and processing images are offered by CV2. Instead, the OS module was used to access image files and go through the directory structure. We employed Optical Character Recognition (OCR) techniques for text extraction from the images. The primary tool used for OCR was the Tesseract OCR engine, which is integrated with Python through the pytesseract library. This library provides a convenient interface for utilizing Tesseract OCR capabilities within Python code. Tesseract was set up to detect Hindi and Sanskrit correctly because the Sushruta Samhita may contain text in these languages. The language parameter had to be set in order to specify the language or languages that were present in the text when calling image_to_string.

Output of Named Entity Recognition (NER) using NLTK:



```
            print(named_entities)
            print('\n')

# Example usage
process_text_files_in_folder("/content/drive/MyDrive/sushrut samhita/text.files")
```

```
Named entities in page_3.txt:
[Tree('S', [('गवर्तां', 'JJ'), ('समुपदिश', 'NNP'), ('तच्छि-', 'JJ'), ('धेण', 'NNP'), ('सुश्रुते', 'NNP'), ('विरचिता', 'NNP'), ('।', 'NNP'), ('आरोग्यसुधाकरसंपादकेन', 'NNP'


Named entities in page_4.txt:
[Tree('S', [('-', ':'), ('हपाजछ', 'NN'), ('छर', 'JJ'), Tree('ORGANIZATION', [('नजर', 'NNP')]), ('युदपार', 'NNP'), ('_', 'NNP'), ('बा', 'NNP'), ('उल्ह-द', 'NNP'


Named entities in page_5.txt:
[]


Named entities in page_6.txt:
[Tree('S', [("''", "''"), ('विड220:2022', 'NN'), ('%', 'NN'), ('22', 'CD'), ('%', 'NN'), ('१०७९१०३८००३८१०७/००७/००७/००३/००४००', 'CD'), ('७८', 'CD'), (',', ',')


Named entities in page_7.txt:
[Tree('S', [('जो', 'JJ'), ('श्री', 'NNP'), ('॥', 'NNP'), ('इुपत्र', 'NNP'), ('रहस्यम्\u200c', 'NNP'), ('।', 'NNP'), ('"', 'NNP'), ('शरीरमार्य', 'NNP'), ('सत', 'NNP'),
```



```
    if not os.path.exists(output_folder):
        os.makedirs(output_folder)

    # Iterate until satisfactory results are obtained
    iteration = 1
    while True:
        print(f"Iteration {iteration}")

        for file_name in os.listdir(input_folder):
            if file_name.endswith('.txt'):
                input_file_path = os.path.join(input_folder, file_name)
                output_file_path = os.path.join(output_folder, file_name.replace('.txt', '_categories.txt'))

                with open(input_file_path, 'r', encoding='utf-8') as input_file:
                    text = input_file.read()
                    categories = validate_refine_text(text, keywords)

                    with open(output_file_path, 'w', encoding='utf-8') as output_file:
                        for category, words in categories.items():
                            output_file.write(f"{category.capitalize()}: {', '.join(words)}\n")

        # Prompt for user input to continue or break the loop
        user_input = input("Do you want to continue with another iteration? (y/n): ")
        if user_input.lower() != 'y':
            break

        iteration += 1

# Example usage
input_folder = "/content/drive/MyDrive/sushrut samhita/text.files"
output_folder = "/content/drive/MyDrive/sushrut samhita/text.files"

iterative_categorization(input_folder, output_folder)
```

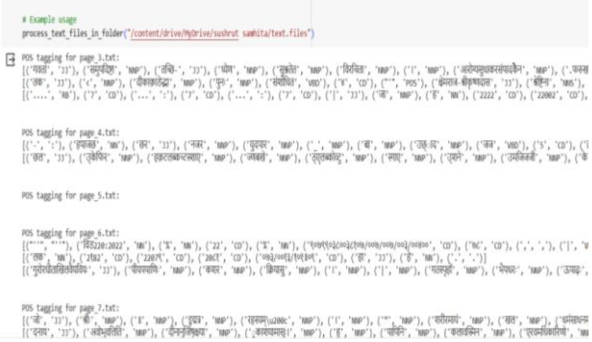
```
Iteration 1
Do you want to continue with another iteration? (y/n): n
```

We performed Named Entity Recognition (NER) on the text data using NLTK (Natural Language Toolkit), a well-liked Python library for natural language processing. Tokenization and entity recognition are two text processing features offered by NLTK. It performed satisfactorily in terms of recall and precision when identifying objects like illnesses, herbal remedies, and living things.

Overall, this script provides a flexible and iterative approach to categorizing text files based on predefined keywords, making it suitable for tasks such as organizing documents or extracting relevant information from text corpora.

Output of Part of Speech (POS) Tagging:

It was shown that NLTK and spaCy could accurately assign POS tags to words in the text data. More detailed examination of the text's linguistic characteristics was made possible by the identification of various parts of speech being made easier by POS tagging. SpaCy offers pre-trained models for POS tagging, which are capable of achieving high accuracy and performance on various text datasets.

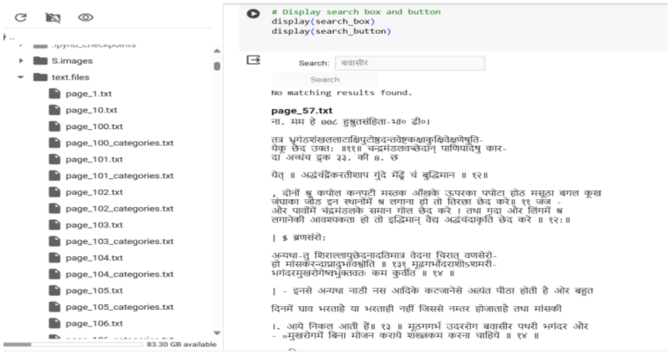


```
# Example usage
process_text_files_in_folder("/content/drive/MyDrive/sushrut samhita/text.files")
```

```
POS tagging for page_3.txt:
[('गवर्तां', 'JJ'), ('समुपदिश', 'NNP'), ('तच्छि-', 'JJ'), ('धेण', 'NNP'), ('सुश्रुते', 'NNP'), ('विरचिता', 'NNP'), ('।', 'NNP'), ('आरोग्यसुधाकरसंपादकेन', 'NNP'), ('.फल्स
[('तक', 'JJ'), ('<', 'NNP'), ('दीकाकछेद्वा', 'NNP'), ('पुन', 'NNP'), ('संशोधित', 'VBD'), ('४', 'CD'), ("''", 'POS'), ('क्षेमराज-श्रीकृष्णदास', 'JJ'), ('श्रेष्टिना', 'NNS'),
[('....', 'RB'), ('7', 'CD'), ('...', ':'), ('7', 'CD'), ('...', ':'), ('7', 'CD'), ('|', 'JJ'), ('जा', 'NNP'), ('ह', 'NN'), ('2222', 'CD'), ('22002', 'CD'),


POS tagging for page_4.txt:
[('-', ':'), ('हपाजछ', 'NN'), ('छर', 'JJ'), ('नजर', 'NNP'), ('युदपार', 'NNP'), ('_', 'NNP'), ('बा', 'NNP'), ('उल्ह-द', 'NNP'), ('जब', 'VBD'), ('5', 'CD'), ('
[('छत', 'JJ'), ('उकेफिर', 'NNP'), ('हकटलकन्टस्याए', 'NNP'), ('जावडे', 'NNP'), ('हात्लकोद्', 'NNP'), ('स्याए', 'NNP'), ('उपने', 'NNP'), ('उमजिवली', 'NNP'), ('क


POS tagging for page_5.txt:


POS tagging for page_6.txt:
[("''", "''"), ('विड220:2022', 'NN'), ('%', 'NN'), ('22', 'CD'), ('%', 'NN'), ('१०७९१०३८००३८१०७/००७/००७/००३/००४००', 'CD'), ('७८', 'CD'), (',', ','), ('|', 'V
[('तक', 'NN'), ('2182', 'CD'), ('2207९', 'CD'), ('20८१', 'CD'), ('००३/००१३/१०११०९', 'CD'), ('श', 'JJ'), ('ह', 'NN'), ('.', '.')]
[('गुरोपतिविलवेपवि', 'JJ'), ('पीयपापि', 'NNP'), ('कपार', 'NNP'), ('कियसु', 'NNP'), ('।', 'NNP'), ('|', 'NNP'), ('गतस्पृह', 'NNP'), ('भेषफर', 'NNP'), ('ऊपाढ़',


POS tagging for page_7.txt:
[('जो', 'JJ'), ('श्री', 'NNP'), ('॥', 'NNP'), ('इुपत्र', 'NNP'), ('रहस्यम्\u200c', 'NNP'), ('।', 'NNP'), ('"', 'NNP'), ('शरीरमार्य', 'NNP'), ('सत', 'NNP'), ('धर्मसाधनम
[('दनाय', 'JJ'), ('अक्षेभ्युवितिति', 'NNP'), ('दैन्यनिश्रय', 'NNP'), ('_कास्यमास्।', 'NNP'), ('ह', 'NNP'), ('पापिने', 'NNP'), ('कलाकिम', 'NNP'), ('एतमणिकारिणां', 'NN
```

Many downstream NLP tasks, including syntactic parsing, information extraction, and sentiment analysis, can be performed using the extracted POS tags.

Output of Interactive Query System:



```
# Display search box and button
display(search_box)
display(search_button)
```

Search: बवासीर

```
No matching results found.
```

page_57.txt
ना. मम हे ००८ इुश्रुतसंहिता-भा० ढी०।

तत्र भ्रूगंडशंखललाटाक्षिपुटोष्ठदन्तवेष्टकक्षाकुक्षिवेक्षणेषुति-
येकृ छेद उक्तः ॥११॥ चन्द्रमंडलवच्छेदान् पाणियादेषु कार-
दा अच्धंच द्रक ३३. की ४. छ

येत् ॥ अद्धंचंद्रैकरतीशाप गुंदे मेंढ्रे चं बुद्धिमान ॥ १२॥

, दोनों श्रु कपोल कनपटी मस्तक आँखके ऊपरका पपोटा होठ मसूठा बगल कूख
जंघाका जोड इन स्थानोंमें श्र लगाना हो तो तिरछा छेद करे॥ ११ जंज -
और पावोंमें चंद्रमंडलके समान गोल छेद करे । तथा गुदा और लिंगमें श्र
लगानेकी आवश्यकता हो तो इद्धिमान् वैद्य अद्धंचंदाकृति छेद करे ॥ १२:॥

| $ व्रणसंरो:

अन्यथा-तु शिराल्लायुछेदनादतिमात्र वेदना चिरात् वणसेरो-
हो मांसकरन्दाप्रादुर्भावश्चेति ॥ १३१ मूढगर्भोदराशोऽश्मरी-
भगंदरमुखरोगेष्वभुक्तवतः कम कुर्वीत ॥ १४ ॥

| - इनसे अन्यथा नाडी नस आदिके कटजानेसे अत्यंत पीडा होती है ओर बहुत

दिनमें घाव भरताहे या भरताही नहीं जिससे नम्तर होजाताहे तथा मांसकी

।. आये निकल आती हैं॥ १३ ॥ मूढगगर्भ उदररोग बवासीर पथरी भगंदर और
- »मुखरोगमें बिना भोजन कराये शस्त्रकम करना चाहिये ॥ १४ ॥

This is the result of the searched disease information from Sushruta Samhita.

With the help of this system's user-friendly interface, users can look up specific diseases or ailments and get pertinent information about the herbal remedies or treatments that are suggested in Sushruta Samhita. Users can quickly and easily browse through the extensive library of Ayurvedic knowledge in the text thanks to its methodical organization and indexing. For practitioners, researchers, and enthusiasts interested in learning more about the therapeutic potential of Ayurvedic medicine as described in Sushruta Samhita, the interactive query system is a great resource. Through the system's facilitation of access to this age-old wisdom, it helps to ensure that Ayurvedic knowledge is preserved, shared, and applied in modern healthcare settings, ultimately promoting global healing and holistic well-being.

5. CONCLUSION:

Using Natural Language Processing (NLP) to analyze the Sushruta Samhita, invaluable insights are revealed regarding ancient Indian medicine and surgery. Key findings include advanced surgical techniques, detailed descriptions of various diseases and their treatments, emphasis on preventive medicine, and holistic approaches to healthcare. Additionally, the text underscores the importance of ethical conduct and compassionate care in medical practice, reflecting the rich heritage of traditional Indian medicine.

6. ACKNOWLEDGEMENTS

We would like to express our gratitude to the authors and compilers of the Samhita Samhita for preserving invaluable ancient wisdom. The insights provided are based on the invaluable knowledge preserved in the Sushruta Samhita, an ancient Sanskrit text on medicine and surgery. This summary draws upon the expertise of researchers and scholars who have studied and analyzed the text, as well as the advancements in Natural Language Processing (NLP) techniques that enable the extraction and interpretation of information from such historical texts. Special thanks to Dr. Preenon Bagchi for their invaluable guidance and insights throughout the project. We are also thankful to Institute of Bioscience and technology, MGM University for providing resources and facilities. Additionally, we extend our appreciation to the participants who generously shared their knowledge and experiences. This work wouldn't have been possible without their contributions.

7. REFERENCES

1. Pathiranage, N., Nilfa, N., Nithmali, M., Kumari, N., Weerasinghe, L., & Weerathunga, I. (2020, October). Arogya-An Intelligent Ayurvedic Herb Management Platform. In *2020 61st International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS)* (pp. 1-6). IEEE.
2. Bagde, A. B., Ramteke, A. T., Sawant, R. S., Bhingare, S. D., & Nikumbh, M. B. (2017). Sushruta Samhia-a Unique Encyclopedia of Ayurvedic Surgery. *World Journal of Pharmacy and Pharmaceutical Sciences (online)*, *6*(4), 750-767.
3. Lad, V. (2002). Textbook of Ayurveda: Fundamental Principles of Ayurveda 1 Albuquerque, NM The Ayurvedic Press.
4. Ragho, D. S. (2020). Sushruta Medical Contribution in Ancient India A Historical Study.
5. Vasiliev, Y. (2020) Natural language processing with Python and spaCy: A practical introduction. No Starch Press.
6. Schlutter, A., & Vogelsang, A. (2020) Knowledge extraction from natural language requirements into a semantic relation graph. In Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops (pp. 373-379).