International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT2390242

Search Engine on Vedas

Krishna Kumar Yadav, Samaranjan Manjhi, Sonu Sharma, Premchand Gupta

Department of Computer Engineering, Shree L.R. Tiwari College of Engineering, Mumbai, Maharashtra, India

ARTICLEINFO

Article History:

Accepted: 05 April 2023
Published: 17 April 2023

Publication Issue
Volume 10, Issue 2
March-April-2023

Page Number
369-376

ABSTRACT

The Search Engine on Vedas Project is an effort to create a comprehensive and user-friendly search engine for the vast corpus of Vedic literature. The project involves digitizing and indexing a diverse range of Vedic texts, including hymns, rituals, commentaries, and translations, and developing a user-friendly interface that allows efficient searching and browsing. The search engine employs advanced search algorithms, such as natural language processing and machine learning, to enable users to search for specific keywords, phrases, or concepts within the Vedic texts. The interface is designed to be intuitive and customizable, with options for filtering and sorting search results based on various criteria, such as author, type of text, language, and publication date. The project also includes interactive features, such as multimedia resources, glossaries, and forums, to facilitate a more engaging and immersive user experience. Additionally, the search engine aims to provide a platform for collaborative research and knowledge-sharing among scholars and enthusiasts of Vedic literature.

Keywords : Indexing, Memory, Application Program Interface (API), Database

I. INTRODUCTION

The Vedas are ancient texts that are considered the foundation of Hinduism, one of the world's oldest religions. These texts contain a wealth of knowledge on topics ranging from philosophy, spirituality, and mythology to science, mathematics, and astronomy. However, accessing and searching through these texts can be challenging due to their vast size, complexity, and diversity.

To address this challenge, a search engine on Vedas project can be developed to help users find relevant information from these ancient texts quickly and efficiently. The search engine on Vedas project involves several key steps, including data acquisition, indexing, search algorithm development, and user interface design. [1]

The first step in the search engine on Vedas project is data acquisition. The Vedas consist of four primary texts: Rigveda, Yajurveda, Samaveda, and Atharvaveda. These texts have been passed down through generations in the form of manuscripts, translations, and commentaries. The data acquisition process involves collecting and digitizing these texts and making them available in a searchable format.

Once the data has been acquired, the next step is indexing. Indexing involves breaking down the text into smaller units, such as words or phrases, and creating an index that maps each unit to its location in the text. The index serves as a roadmap that enables the search engine to quickly locate relevant information in response to user queries.

The search algorithm development is the most critical step in the search engine on Vedas project. The search algorithm determines how the search engine ranks the search results based on the user's query. Various algorithms can be used, such as the TF-IDF (Term Frequency-Inverse Document Frequency) or BM25 (Best Matching 25) algorithm. These algorithms use statistical methods to determine the relevance of each document to the user's query.

In response to these limitations of a search engine on Vedas project that should be considered when developing and using the search engine. The Vedas are a vast collection of ancient texts, and it may not be possible to have a complete or accurate representation of all texts in digital form. This can lead to limitations in the scope of search results and may require manual curation of search results. The Vedas are written in ancient languages, such as Sanskrit and Vedic Sanskrit, which may require specialized language processing techniques to enable effective search capabilities. There may be limitations in the ability of the search engine to accurately process and interpret these languages. The user interface design is also a crucial aspect of the search engine on Vedas project. The user interface should be intuitive and user-friendly, allowing users to enter their search queries and filter the results based on various criteria. The user interface should also provide users with additional information, such as the context and relevance of each search result.

Once the search engine on Vedas project has been developed, it can be deployed on a web server or cloud platform. Users can access the search engine through a web browser or a mobile app, allowing them to search for information from the Vedas anytime and anywhere. The search engine can also be integrated with other tools and platforms, such as digital libraries or educational resources, to provide users with a more comprehensive research experience. [2]

In conclusion, the search engine on Vedas project is an important initiative that can help unlock the knowledge and wisdom contained within these ancient texts. By making the Vedas more accessible and searchable, this project can benefit scholars, students, and anyone interested in exploring the rich history and culture of India. With continued development and innovation, the search engine on Vedas project can become an essential tool for researchers and enthusiasts worldwide.

The Vedas Project has already made significant progress in digitizing and translating the Vedas. The project has digitized over 15,000 Vedic manuscripts and translated them into various languages. The project has also created several interactive tools for learning and exploring the Vedic texts, including a mobile app called "Vedas 360," which allows users to explore the Vedas in 3D.

II. LITERATURE REVIEW

1) Apte, V. S. (1884). The Student's English-Sanskrit Dictionary. Arya Bhushana, Poona. Apte. V. S. (1890). The Practical Sanskrit-English Dictionary. Shiralkar, Poona[1] examines at first sight RDF a seemed to tackle our requirements bet- ter than creating new files containing RDF triples when transforming TEL source files into Ontolex - Lemon. But a brief modelling exercise showed that employing RDF a required adding new elements into the TEI source files These structural modifications to the TEI files would un- necessarily complicate the maintenance of these files over time. Therefore, the method to follow will be to create com pletely new files modelled in Ontolex Lemon.

2) Chiarcos, C. and Ionov, M. (2021). Linking the TEI. Approaches, Limitations, Use Cases. Digital Humanities 2019, July [2] explores there are built an English to Hindi translator using RNN. We experimented with long short-term memory (LSTM) and attention mechanism. Using the attention mechanism and LSTM the correct translation to a target language is made possible. In this project, we have added a feature that we can directly upload a document that is to be translated so eventually it reduces the typing time. To make the translation process more efficient, new rules can be added to the system.

3) Lugli, L. (2020). Drifting in Timeless Polysemy: Chronology in Sanskrit Lexicography. Dictionaries: Journal of the Dictionary Society of North America, 39(1):105-129, August. [4] Many of the ancient and modern documents were written in Sanskrit, since we use an Machine translation system, it will translate Sanskrit to English language. The proposed system can be used for educational purposes, communication etc. The translation o Sanskrit to English is performed very well.

4) Charles Taliaferro (2021). A Dictionary of Philosophy of Religion. Bloomsbury Publishing. pp. 245–246. ISBN 978-1-4411-8504-4. [5] proposes machine Translation is area of research since six decades. It is gaining popularity since last decade due to better computational facilities available at personal computer systems. This paper presents different Machine Translation system where Sanskrit is involved as source, target or key support language.

III. METHODOLOGY

The methodology of the search engine on Vedas project involves several steps, including data collection, data storage, data indexing, search algorithms, user interface design, testing, deployment, and maintenance. Overall, the methodology of the search engine on Vedas project requires a combination of technical expertise in areas such as database management, search algorithms, and web development, as well as knowledge of Vedic texts and related resources. The process involves a systematic and iterative approach to ensure that the search engine is efficient, effective, and user-friendly.

IV. TECHNICAL OVERVIEW

Coming to the technical aspects of our search engine on Vedas project involves several key components and technologies, including data storage, indexing, search algorithms, user interface design, and web development.

The Vedic texts and related resources are stored in a database or other storage medium, along with the necessary metadata such as author, date, type of text, and language. The database may be structured or unstructured, depending on the nature of the data.

A search index is created for the digitized texts, which involves breaking them down into smaller units such as words or phrases and recording their locations in the database. The index enables efficient searching and retrieval of relevant information.

Advanced search algorithms are developed to process user queries and return relevant search results. These algorithms may use techniques such as keyword matching, natural language processing, and machine learning to enhance the accuracy and relevance of search results.

The search engine is developed using web technologies such as HTML, CSS, JavaScript, and server-side scripting languages such as PHP or

Python. The web application may be hosted on a web server or cloud platform, and may include features such as caching, load balancing, and security measures to ensure optimal performance and user experience.

Selection of key factors for evaluation:

Evaluation Factor [6]	Description
Indexing and data storage	The search engine should have a robust indexing system that enables it to efficiently store and retrieve data. The data should be organized in a logical and coherent manner, with metadata that helps users find what they are looking for.
Relevance of search results	The search engine should be able to deliver relevant and useful results that meet the needs and interests of users. The search engine should be able to understand user intent and provide results that are tailored to their needs. factors will be considered as separate factors for evaluation. Therefore, it is only necessary to evaluate the operational cost.
Accessibility	The search engine should be accessible and easy to use for all users, regardless of their technical expertise or background. It should be designed with usability, social media, electronic articles, and the like to identify the size of the community. In addition, interest groups, companies and branches, list of partners, and the number of clients were considered
Ease of learning	Ease of learning depends on two factors. The first factor, simplicity, was discussed above. The second factor is identifying the programming languages that are used in the blockchain platform. The programming language is a key factor for platform selection because if the blockchain platform offers commonly known and presently available programming languages, the development process will be expeditious and convenient for system developers
Usability	accessibility in mind, with features such as text-to-speech and translation capabilities to enhance accessibility for users with disabilities or those who speak different languages.
Speed and efficiency	The speed and efficiency required for a search engine on Vedas project would depend on the specific requirements and usage patterns of the target users. Generally, a search engine on Vedas should be able to return results quickly and efficiently, without delays or timeouts, to provide a satisfactory user experience. The larger the dataset, the more time and resources it will require to process search

	queries and return results. A search engine on Vedas project that needs to process a large amount of data may require more processing power and storage capacity to maintain speed and efficiency. The number of concurrent users accessing the search engine at any given time can impact its speed and efficiency. A search engine on Vedas project that needs to serve a large number of users simultaneously may require more processing power and bandwidth to maintain optimal performance.
User interface design	The user interface should be easy to use and intuitive, with features such as search bars, filters, and sorting options to help users refine their search results.
Security	The search engine should be designed with security and privacy in mind, with appropriate measures in place to protect user data and prevent unauthorized access or use.
Indexing and data storage	The search engine should have a robust indexing system that enables it to efficiently store and retrieve data. The data should be organized in a logical and coherent manner, with metadata that helps users find what they are looking for.
Support for web-application and mobile app development with GUIs	GUI is a common feature of any form of present software application. However, most of the platforms are not capable enough to develop GUI-enabled software apps. Commonly, blockchain platforms provide Software Development Kits (SDKs) to develop software applications to access blockchain data. These SDKs support some of the presently available general-purpose high-level programming languages such as Java, Go, Python, Ruby, Perl, and others. The SDK allows the development of GUI-enabled standard, web, or mobile applications natively or through APIs.
Type of software licenses	There are different types of open source licenses, such as GPL, MIT, and Apache. The software is owned by a company or individual and may require a license fee or subscription to use. Freeware is a type of software that is available for free and can be used without any license restrictions. However, the user may not be able to modify or distribute the software. Shareware is a type of software that is available for free initially but requires a fee for continued use or access to additional features. Public domain is not protected by copyright and

	can be used, modified, and distributed freely. The specific license for a search engine on Vedas would depend on the software used to develop it, and the licensing terms and conditions set by the software provider. It is important to review the license terms and conditions before using or modifying any software to ensure compliance with legal requirements.

V. IMPLEMENTATION

The implementation of a search engine on Vedas project is a complex process that involves multiple steps, including those that involve social media. The first step in implementing a search engine on Vedas project is to collect the relevant data. This may involve digitizing the Vedas texts or accessing existing digital archives. This process can be time-consuming and requires specialized equipment and software to ensure that the digital copies are accurate and of high quality. It may also require expert knowledge in the field of Vedas texts to identify and locate the relevant texts.

The collected data may require pre-processing to ensure that it is in a format that can be used by the search engine. This may involve cleaning the data, removing duplicates, and standardizing the format. The pre-processing step is critical to ensure that the data is accurate, consistent, and easily searchable. This process can be automated using specialized software or may require manual effort, depending on the complexity of the data. [3] [7]

The search engine requires an index of the data to enable fast and efficient search capabilities. Indexing involves creating a searchable database of the content of the Vedas texts, including words and phrases, and their locations within the text. The indexing process can be resource-intensive and time-consuming, depending on the size and complexity of the data. The index should be optimized for speed and efficiency to enable quick and accurate search results.

The search engine requires algorithms that can process user search queries and retrieve relevant search results from the indexed data. This may involve developing advanced search algorithms that can handle complex queries and language processing techniques that can interpret ancient languages like Sanskrit. The search algorithm should be optimized for accuracy, relevance, and speed to provide a user-friendly search experience.

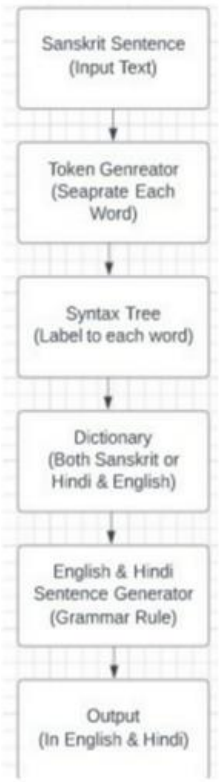
The user interface of the search engine should be intuitive and easy to use. It should allow users to enter search queries and view search results in a user-friendly format. The user interface should be designed with the target audience in mind, and should take into account factors such as user demographics, cultural background, and technical expertise. The design should be optimized for ease of use, accessibility, and compatibility with different devices and browsers.

Once the search engine is developed, it should be thoroughly tested to ensure that it is functioning as intended. Performance testing can identify any issues related to speed and efficiency, and user testing can identify any usability issues that need to be addressed. The search engine should be optimized based on the results of testing to ensure that it meets the desired requirements and user needs. This process may involve making changes to

the search algorithm, user interface design, or data indexing to improve performance and usability.
Once the search engine is tested and optimized, it can be deployed for use by the target users. Ongoing maintenance and updates may be required to ensure that the search engine continues to function optimally over time. This may involve monitoring search engine usage, identifying and addressing technical issues, and making updates to the search algorithm, data indexing, or user interface design as needed.

In summary, implementing a search engine on Vedas project requires expertise in data management, search algorithms, and user interface design. It is a complex process that involves multiple steps, including data collection, pre-processing, indexing, search algorithm development, user interface design, testing and optimization, and deployment. Careful planning and execution are essential to ensure that the search engine meets the desired requirements and provides an effective and user-friendly search experience for users.

VI. DESIGN DETAIL



VII. CONCLUSION & FUTURE WORK

In conclusion, building a search engine on Vedas can be a challenging but rewarding project that involves data indexing, search algorithm development, and user interface design. building a search engine on Vedas can be a challenging and exciting project that can provide valuable insights into the history, culture, and spirituality of ancient India. With continued innovation and development, this search engine can become an essential tool for researchers, students, and enthusiasts of Vedas worldwide. Based on the implementation of a search engine on Vedas, there are several directions for future work that can be pursued. Some of these include:

- Improving search relevance: You can experiment with different ranking algorithms, query expansion techniques, and machine learning models to improve the relevance of your search results.

- Enhancing user experience: You can add new features to your search engine, such as faceted search, personalized recommendations, or multimedia content, to provide users with a more engaging and personalized experience.

- Integrating with other tools and platforms: You can integrate your search engine with other tools and platforms, such as social media, digital libraries, or educational resources, to provide users with a more comprehensive and integrated research experience.

- Scaling and optimizing performance: As your user base grow, you will need to optimize the performance and scalability of your search engine to ensure that it can handle increasing traffic and data volume.

- Integration with other platforms: The Vedas application can serve as a model for implementing Vedic information in other social media platforms. Future work could focus on exploring the integration of this approach with other popular platforms, such as Facebook, WhatsApp, and LinkedIn.

VIII. REFERENCES

[1]. Apte, V. S. (1884). The Student's English-Sanskrit Dictionary. Arya Bhushana, Poona. Apte. V. S. (1890). The Practical Sanskrit-English Dictionary. Shiralkar, Poona.

[2]. Chiarcos, C. and Ionov, M. (2021). Linking the TEI. Approaches, Limitations, Use Cases. Digital Humanities 2019, July

[3]. Tittel, S. and Chiarcos, C. (2020). Linked open data for the historical lexicography of old french. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Paris, France, may. European Language Resources Association (ELRA).

[4]. Lugli, L. (2020). Drifting in Timeless Polysemy: Chronology in Sanskrit Lexicography. Dictionaries: Journal of the Dictionary Society of North America, 39(1):105-129, August.

[5]. Charles Taliaferro (2021). A Dictionary of Philosophy of Religion. Bloomsbury Publishing. pp. 245–246. ISBN 978-1-4411-8504-4.

[6]. Promila Bahadur, AK Jain, and DS Chauhan. 2012. Etrans-a complete framework for english to sanskrit machine translation.

[7]. Hiram Calvo, Arturo P Rocha-Ramirez, Marco A Moreno-Armendariz, and Carlos A Duchanoy. 2019. Toward universal word sense disambiguation using deep neural networks. IEEE Access, 7:60264– 60275.

[8]. Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale.

[9]. Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation,

[10]. Nimrita Koul and Sunilkumar S Manvi. 2019. A proposed model for neural machine translation of sanskrit into english.

Cite this article as :

Krishna Kumar Yadav, Samaranjan Manjhi, Sonu Sharma, Premchand Gupta , "Search Engine on Vedas", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 9, Issue 2, pp.369-376, March-April-2023. Available at doi : https://doi.org/10.32628/CSEIT2390242
Journal URL : https://ijsrcseit.com/CSEIT2390242