JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

TRANSLITERATION KEYBOARD FOR DEVANAGARI ALPHABET

[1]Kaustav Sanyal, [2]P. Shivani Rao, [3]Jasmeet Kour, [4]Priyanka Kumari, [5]Sumi Kumari

[1]Head of Department (CSE), [2]Student, [3]Student, [4]Student, [5]Student
Computer Science and Engineering
Durgapur Institute of Advanced Technology and Management, Durgapur, India

Abstract : This Synopsis explores the development of a Python-based tool for converting English language text into Devanagari script. Leveraging the capabilities of the Google Translate API through the google trans library, the proposed system aims to bridge the gap between English and Devanagari, facilitating cross-language communication and understanding. The use of transliteration keyboard for Devanagari alphabet. In order to do this work we will use python programming language. This paper describes how to use a transliteration strategy in small scale systems to provide configurable localization support. Translation memory is used to provide customisable Localization support for Indian languages like Hindi, which are based on the Devanagari script. The goal of this work is to enable the user to input and retrieve data in Hindi on the fly. We'll be transliterating from English to Devanagari on the spot. No storage is being done with the transliterated data. There are two methods for creating a transliteration tool that aids in translating English text into its equivalent Devanagari language. The first way illustrates typing using a computer keyboard, and the second method uses a virtual keyboard to explain typing. Our goal is to create a reliable and user-friendly program that enables users to convert English text into Devanagari without making any mistakes.

Keywords: leveraging, Python-based tool, Devanagari, Google library.

I. INTRODUCTION

Language translation has become an essential aspect of our interconnected world, facilitating communication across linguistic boundaries. In the context of English and Devanagari, the need for an efficient and accurate translation tool is evident, considering the cultural and historical significance of Devanagari in the Indian subcontinent. This paper introduces a Python-based solution for the automatic conversion of English text into Devanagari script, enabling a seamless exchange of information and ideas between English and Devanagari-speaking communities. [1] The majority language of India, Hindi, uses the same character, Devana gari, although research and applications on natural language processing are still in their infancy. While a lot of work is being done on English title generation these days, less work has been done on Hindi due to a lack of familiarity with Hindi or Sanskrit keyboards. There are just two alphabetic forms in the English language: the 21 consonants (such as B, C, and D) and the 5 vowels (a, e, i, o, and u). In contrast, the Hindi language has 40 consonants and 10 vowels, which are made up of a variety of symbols (the ten vowels include half letters, alphabets, and signs, among other things). Thus, Hindi cannot be directly taught using the English language's current means. Numerous books, records, newspapers, official notice, Since Hindi is used in the writing of books, magazines, and other publications, Hindi title creation tools. Only binary numbers are understood by computers; humans are unable to work with binary numbers. Therefore, in order for computers to comprehend human-understandable language, we must be developed. The interaction between computers and human languages is the focus of the scientific study of language and computer science subject known as natural language processing, or NLP. The connection between computers and humans is fraught with difficulties need a sizable database kept in our system. Natural Language Processing (NLP) is a method by which people can communicate with computers. [1]

II. MOTIVATION

The rise of globalization has led to an increase in the demand for efficient communication between individuals who speak different languages. In this case, language transliteration becomes an essential tool for bridging linguistic gaps. While English is the most widely spoken language in the world, internet communication can be challenging for many regional languages, especially those written in the Devanagari script. The awareness that a seamless and user-friendly transliteration tool is needed to make it possible to convert English text to Devanagari script is what inspired this work. This project aims to develop a transliteration keyboard that addresses the problem of linguistic variety and allows users to rapidly transition between English and other languages using the power of Python programming. [6]

III. PROJECT OBJECTIVE

Design and Implementation: Develop a user-friendly transliteration keyboard system that seam less converts English input to the Devanagari script. This in volves the must be developed. The interaction between computers and human languages is the focus of the scientific study of language and computer science subject known as natural language processing, or NLP. The connection between computers and humans is fraught with difficulties need a sizable database kept in our system. Natural Language Processing (NLP) is a method by which people can communicate with computers. [1] design and implementation of an intuitive interface that ensures ease of use for a diverse range of users.

Language Processing Algorithms: Employ advanced language processing algorithms, leveraging Python's capabilities, to accurately transliterate English text into Devanagari.

Real-time Transliteration Feedback: Implement real-time feedback mechanisms to provide users with instant transliteration results as they type. This feature enhances user confidence and allows for corrections before finalizing the output.

Customization Options: Incorporate customization features, allowing users to tailor the transliteration keyboard to their preferences. This may include options for modifying transliteration rules, selecting font styles, and adjusting input sensitivity.

Error Handling and Correction: Implement mechanisms for detecting and correcting transliteration errors. This includes the Development of algorithms to identify common mistakes and suggest corrections, contributing to the overall accuracy of the transliteration process. [8]

Educational Resources: Develop user guides and tutorials to assist users in understanding and maximizing the features of the transliteration keyboard. Include tooltips and help features within the keyboard interface.

Accessibility: Ensure accessibility features for users with disabilities, including keyboard navigation and screen reader compatibility.

Security and Privacy: Implement secure data handling practices to protect user input and maintain privacy. Comply with relevant data protection regulations and standards.

Documentation: Create comprehensive documentation for both end-users and developers, covering installation, configuration, and customization. By achieving these objectives, the project aims to deliver an efficient and user friendly Python-based transliteration keyboard that effectively converts English text to Devanagari, contributing to improved digital communication across linguistic boundaries. [7]

Evaluation: The accuracy and performance of the conversion tool were evaluated using standard metrics, including precision, recall, and F1 score. Additionally, user feedback and subjective assessments were collected to gauge the tool's practical utility and user satisfaction. [8]

Ethical Considerations: The use of the Google Translate API is subject to ethical considerations, and adherence to the API's terms of service is paramount. Privacy concerns and data security were addressed in the implementation, ensuring the responsible use of translation services. [11]

Limitation: It's critical to be aware of the proposed system's limitations, which include the possibility of translation errors, reliance on other services, and requirement for an internet connection for real-time translation. [10]

IV. RESEARCH METHODOLOGY

The methodology of this research involves the following steps:

Data Collection: A diverse dataset comprising English text samples was collected to train and evaluate the translation model. The dataset covers a range of topics and linguistic nuances to ensure the robustness of the conversion tool. [9]
Implementation of Translation Algorithm: The core of the conversion tool is built on the google library, a Python wrapper for the Google Translate API. This library facilitates the translation of English text to Devanagari script by sending requests to the Google Translate service. [2]

Evaluation: The accuracy and performance of the conversion tool were evaluated using standard metrics, including precision, recall, and F1 score. Additionally, user feedback and subjective assessments were collected to gauge the tool's practical utility and user satisfaction. [8]

Ethical Considerations: The use of the Google Translate API is subject to ethical considerations, and adherence to the API's terms of service is paramount. Privacy concerns and data security were addressed in the implementation, ensuring the responsible use of translation services. [11]

Limitation: It's critical to be aware of the proposed system's limitations, which include the possibility of translation errors, reliance on other services, and requirement for an internet connection for real-time translation. [10]

V. CONCLUSION

The development of a transliteration keyboard using Python for converting English into Devanagari has proven to be a significant and innovative contribution to the field of language processing and digital communication. This research aimed to address the challenges associated with linguistic diversity, providing a user-friendly tool that seamlessly bridges the gap between English and Devanagari, enhancing cross-cultural in teraction and digital accessibility. The successful completion of this synopsis is expected to result in the development of a highly efficient and user-friendly transliteration keyboard for the Devanagari alphabet. The transliteration scheme, keyboard layout, and predictive text algorithms employed will collectively contribute to an enhanced user experience, making it easier to input Devanagari text seamlessly. The synopsis outcomes could have applications in language education, communication technology, and cross-cultural interactions.

.

VI. FUTURE WORK

The field of language translation has never been more exciting. The opportunities for translation management systems to generate accurate, real-time translations are numerous, thanks to the growing and evolving development of artificial intelligence, machine learning, and natural language processing.

Enhanced Accuracy Models: Investigate advanced machine learning models to improve transliteration accuracy, considering deep learning or transformer architectures.

User-Centric Customization: Explore user preferences and behaviour to implement personalized features, allowing users to tailor the keyboard to their specific transliteration patterns.

Cross-Platform Integration: Focus on expanding compatibility across different operating systems and devices, ensuring a seamless user experience on various platforms.

Natural Language Processing (NLP) Integration: Integrate NLP techniques to enhance contextual understanding and improve the system's ability to predict accurate Devanagari transliterations based on English input.

Multilingual Support: Extend the transliteration capabilities to cover multiple languages, facilitating a more versatile and inclusive user experience.

Accessibility Features: Implement features to enhance accessibility, such as voice input, gesture typing, or other innovative methods to cater to a broader user demographic.

Continuous User Feedback: Establish mechanisms for collecting user feedback to iteratively refine and enhance the transliteration model based on real-world usage and evolving language patterns.

Security and Privacy Measures: Address potential concerns related to data privacy and security, ensuring that user information is handled responsibly and securely.

VII. REFERENCES

[1] PJ Antony and KP Soman. Machine transliteration for indian languages: A literature survey. International Journal of Scientific & Engineering Research, IJSER, 2:1–8, 2011.

[2] Bushra Baig, M Kumar, and Sujoy Das. Rule based hindii to urdu transliteration system. Journal of Emerging Trends in Computing and Information Sciences, 3(8), 2012.

[3] Neha Bhadwal, Prateek Agrawal, and Vishu Madaan. A machine translation system from hindi to sanskrit language using rule based approach. Scalable Computing: Practice and Experience, 21(3):543–554, 2020.

[4] Rupal Bhargava, Yashvardhan Sharma, and Shubham Sharma. Sentiment analysis for mixed script indic sentences. In 2016 International conference on advances in computing, communications and informatics (ICACCI), pages 524–529. IEEE, 2016.

[5] Manik rao Dhore and Kumar Shantanu. Development of bilingual application using machine transliteration: A practical case study. CiiT International Journal of Artificial Intelligent Systems and Machine Learning, 3:859–864, 12 2011.

6] Harry Gavali. Text sentiment analysis of Marathi language in English And Devanagari script. PhD thesis, Dublin Business School, 2020.

[7] Trishita Ghosh, Shiba prasad Sen, S k Md Obaidullah, KC Santosh, Kaushik Roy, and Umapada Pal. Advances in online handwritten recognition in the last decades. Computer Science Review, 46:100515, 2022.

[8] Leena Jain and Prateek Agrawal. English to sanskrit transliteration: an effective approach to design natural language translation tool. International Journal of Advanced Research in Computer Science, 8(1):1–10, 2017.

[9] Kamaljeet Kaur and Parminder Singh. Review of machine transliteration techniques. International Journal of Computer Applications, 107(20), 2014. 19

[10] Jayashree Nair, Riyaz Ahammed, and Anakha Shaji. A study on transliteration techniques and conventional transliteration schemes for indian languages. In Sustain able Communication Networks and Application: Proceedings of ICSCN 2021, pages 103–117. Springer, 2022.

[11] Jayashree Nair and Anand Sadasivan. A roman to devanagari back-transliteration algorithm based on harvard-kyoto convention. In 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), pages 1–6. IEEE, 2019.

[12] Nandini Sethi, Amita Dev, and Poonam Bansal. A bilingual machine transliteration system for sanskrit-english using rule-based approach. In 2022 4th International Conference on Artificial Intelligence and Speech Technology (AIST), pages 1–5. IEEE, 2022.