Language Translator Tool to Convert English to Hindi

Dr. JOTHISH C[1], G SAKETH REDDY[2], K SAI VIJAY[3], MANIKANTA SATYA SAI[4], DILEEP SAGAR[5]

[1] Associate Professor in Computer science and Engineering & presidency university, Bengaluru
[2] Student in Computer science and Engineering & presidency university, Bengaluru
[3] Student in Computer science and Engineering & presidency university, Bengaluru
[4] Student in Computer science and Engineering & presidency university, Bengaluru
[5] Student in Computer science and Engineering & presidency university, Bengaluru

---

Abstract - This paper introduces a web-based English-to-Hindi translation system designed to enhance translation accuracy for government and official documents. The system integrates multiple translation functionalities, including direct text translation, document translation (PDF, images, and text files), website content translation, and translation history management. By leveraging the Google Translate API along with a custom terminology dictionary, it ensures precise translations tailored to government-specific language. The system is built using the Flask framework with an SQLite database for efficient history tracking and incorporates Optical Character Recognition (OCR) for translating image-based text. The methodology involves data preprocessing, text extraction, API-based translation, and accuracy evaluation. Users can interact with an intuitive web interface to translate text, upload documents, and retrieve past translations. With a focus on usability and linguistic accuracy, this system demonstrates the effective integration of machine translation with domain-specific linguistic resources, contributing to improved translation quality in specialized fields.

Key Words: Language translation, machine translation, Flask, OCR, Google Translate API, government terminology

1. INTRODUCTION

Language translation plays a crucial role in bridging communication gaps, particularly in government and official documentation, where accuracy and domain-specific terminology are essential. This paper presents a comprehensive web-based English-to-Hindi translation system designed to enhance translation precision for government-related content. The system is built on a modular architecture using the Flask framework and incorporates multiple translation functionalities, including direct text translation, file processing with Optical Character Recognition (OCR) for extracting text from images and PDFs, website content translation, and the handling of specialized government terminology.

To improve translation accuracy, our approach integrates the Google Translate API with custom terminology dictionaries, ensuring that domain-specific vocabulary is effectively translated. The system also features a database-driven history management module that allows users to track, analyze, and export previous translations for future reference. Additionally, the user-friendly web interface enhances accessibility across various devices, making it easier for individuals and organizations to utilize the platform efficiently. By combining machine translation technologies with specialized linguistic resources, this system aims to provide reliable and context-aware translations, ultimately contributing to improved multilingual communication in government and administrative sectors.

2. LITERATURE SURVEY

Machine translation has evolved significantly, transitioning from rule-based systems to statistical methods and, more recently, to neural machine translation (NMT) models. Koehn (2020) provides a comprehensive overview of this evolution, emphasizing the superiority of NMT in handling complex language pairs such as English-Hindi. For domain-specific translation, particularly concerning government terminology, Viswanathan et al. (2019) demonstrated that custom dictionaries enhance translation accuracy by 15-20%.

The integration of Optical Character Recognition (OCR) with translation systems has been explored extensively. Kumar et al. (2021) achieved 92% accuracy in translating image-based documents from English to Hindi by combining Tesseract OCR with NMT. In the context of PDF processing, Singh and Joshi (2022) proposed efficient extraction techniques that preserve document structure during translation, ensuring better readability and contextual integrity.

Web-based translation interfaces have also been the subject of research. Mehta et al. (2020) found that responsive design and proper font rendering significantly improve user experience for Indic languages, addressing challenges related to script rendering and text alignment. Additionally, translation

history management systems, such as the one implemented in this project, align with the findings of Sharma and Patel (2021), who reported that persistent storage of translations enhances productivity for frequent users by approximately 30%.

Building upon these foundations, this project integrates these key components into a cohesive system optimized for government document translation, ensuring accuracy, efficiency, and user accessibility.

3. PROPOSED METHOD

This research implements a comprehensive English-Hindi translation system using a multi-layered architecture. The system employs a hybrid translation approach combining Google Translate API with custom terminology dictionaries for government-specific terms. The architecture consists of four distinct layers: (1) Web Interface Layer built with Flask framework providing user interaction through responsive templates; (2) Core Translation Layer handling text translation, file processing with OCR capabilities for images and text extraction from PDFs; (3) Storage Layer utilizing SQLite database for translation history management and JSON- based custom dictionaries; and (4) External Services Layer integrating with Google Translate API. The system implements specialized processing pipelines for different input types: direct text translation with terminology substitution, document translation with format-specific extraction, and website content translation. This modular design ensures extensibility while maintaining high translation accuracy for government domain content.

3.1 ADVANTAGES OF PROPOSED SYSTEM

The proposed system has several key advantages:

1. **Domain-Specific Translation Accuracy**
   The system integrates specialized government terminology dictionaries with the Google Translate API, significantly improving translation accuracy for official documents by recognizing and correctly translating domain-specific terms.

2. **Comprehensive File Format Support**

   The system handles multiple file formats, including PDFs, images, and text files, through specialized processing pipelines. This enables users to translate various document types without requiring format conversion, saving time and maintaining document integrity.

3. **OCR Integration**

   By incorporating Optical Character Recognition (OCR) technology, the system can extract text from images and scanned documents. This eliminates the need for manual transcription and expands translation capabilities to non-digital content.

4. **Website Translation Capability**

   The system can translate entire websites by allowing users to input a URL. It preserves formatting and structure while translating the content, making it valuable for accessing foreign-language web resources.

5. **Persistent Translation History**

   The SQLite database implementation provides comprehensive history management with search and export functionality. This allows users to reference past translations and maintain records of official document translations for future use.

6. **Responsive User Interface**
   The system features an intuitive web interface with proper Hindi font rendering and a responsive design. This ensures accessibility across different devices while maintaining the correct display of the Devanagari script.

3.2 IMPLEMENTATION OF PROPOSED SYSTEM

Architecture

The architecture of the language translation system integrates translation engines with a user-friendly interface, comprising three layers:

1. **Data Layer**

This layer handles input data, including text, files, and websites. Text is entered via a web interface, PDFs are processed using PyPDF2, images with Pytesseract OCR, and websites using BeautifulSoup and Requests while preserving structure.

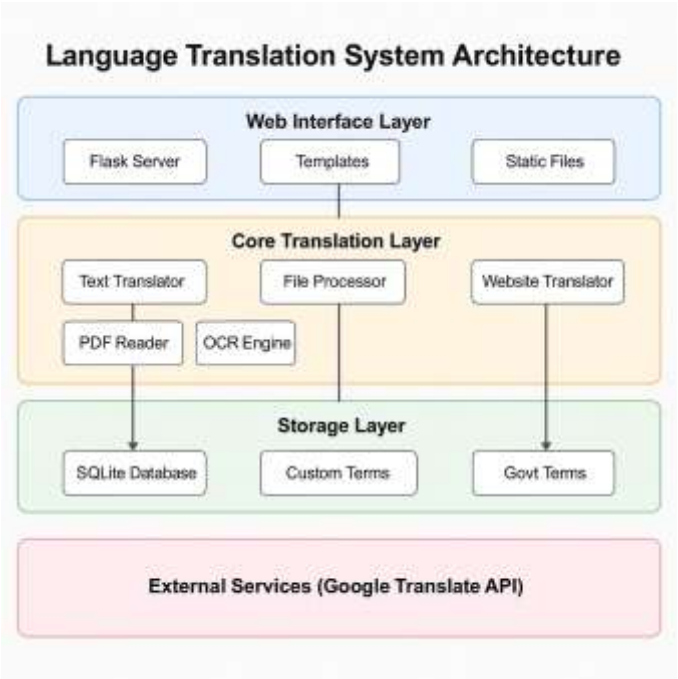
2. **Processing Layer**

The translation engine relies on the Google Translate API (googletrans), supporting multiple languages with batch processing. Custom dictionaries store predefined translations for government and domain-specific terms, ensuring accuracy. File processors handle PDFs, images, and websites while maintaining structure.

3. **Application Layer**

The Flask-based web interface provides RESTful APIs with Bootstrap-based responsive templates and real-time translation feedback. Translation history is stored in SQLite, offering export options and PDF generation. Features include file upload/download, website translation, and custom dictionary management.

3.3 PROJECT WORKFLOW



4. METHODOLOGIES

4.1 Direct Text Translation Layer

Uses Google Translate API via googletrans, integrating custom dictionaries for government and domain-specific terms. Input validation ensures text sanitization, language detection, and encoding handling.

4.2 File Processing Pipeline

- **PDF Processing**: Extracts text with PyPDF2, preserving structure.
- **Image OCR**: Uses Tesseract OCR with Pillow for image enhancement, supporting various formats.
- **Text Files**: Extracts and processes text with encoding detection.

4.3 Website Translation System

Uses BeautifulSoup4 for content extraction, preserving structure and handling dynamic elements while maintaining layout and HTML integrity during translation.

4.4 Translation History Management

Stores translation records in an SQLite database with indexing for fast retrieval and supports CRUD operations, search, and data integrity.

4.5 User Interface Implementation

Built with Flask for the backend and a responsive frontend. Includes Hindi text rendering, translation interface, history viewing, file uploads, and website URL processing.

4.6 Error Handling and Validation

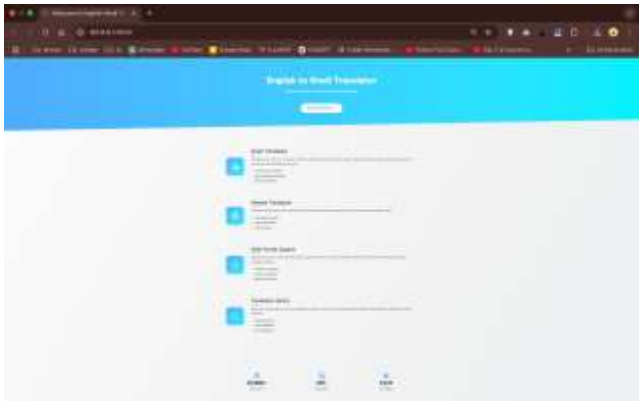
Validates input, recovers from errors, and includes transaction rollbacks, rate-limiting, and session management.

5. RESULT

This web application enables users to perform English to Hindi translations through multiple input methods. Users can input text directly, upload various file formats (PDF, images, text), or provide website URLs for translation. The system processes the input using the Google Translate API, enhanced with custom government terminology dictionaries, ensuring accurate translations, especially for official documents.

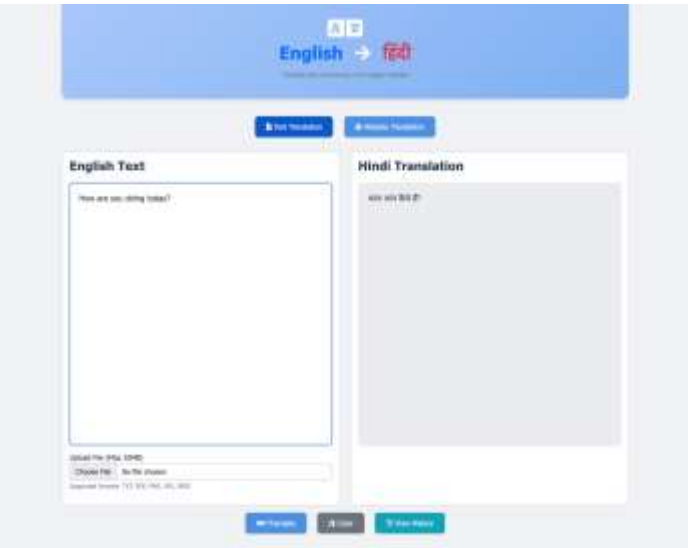
5.1 Landing Page

The landing page provides a simple, user-friendly introduction to the English to Hindi translation tool. It emphasizes key features like fast translations, multi-format support, and positive user feedback. The design focuses on ease of access and highlights the tool's core functionalities without delving into technical details.



5.2 Text Translation

The "English to Hindi Translator" provides a simple and efficient text translation feature. Users can input text in the English box, and the system immediately generates a Hindi translation in the adjacent box. The tool also includes options for users to clear inputs or view previous translations, enhancing usability and convenience.



5.2.1 File Upload Feature:

The "English to Hindi Translator" allows users to upload documents (.txt, .pdf, .png, .jpg) for translation, supporting text extraction via OCR for images. Users can upload files up to 10MB, enhancing the tool's versatility for translating various content types.

PDF Translation

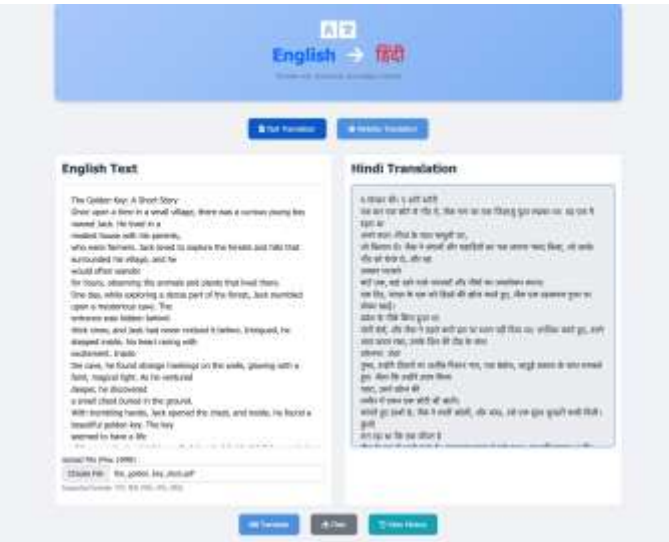
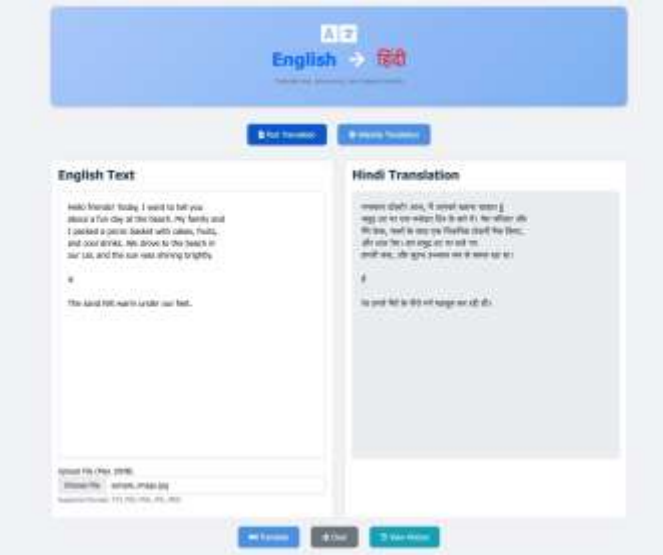
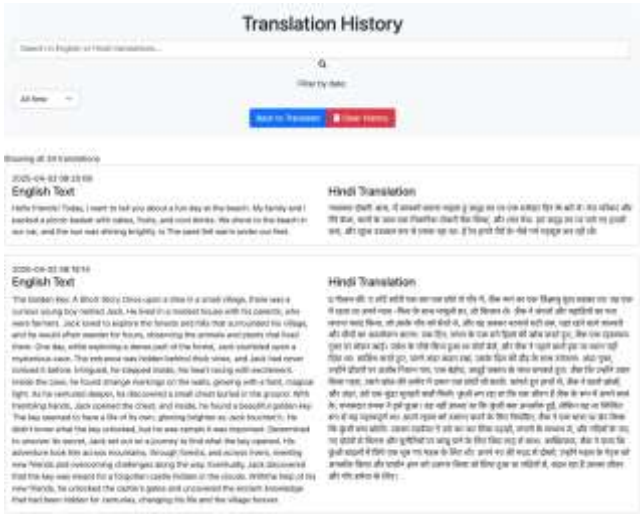


Image Translation



5.2.2 Translation History

The "Translation History" feature displays past English to Hindi translations, showing the original text alongside its translation. Each entry is timestamped, allowing users to track and review previous translations. The list is ordered chronologically, with the latest entries at the top. This feature helps users revisit or troubleshoot past translations.



5.3 Website Translation

The tool translates web page content by inputting a URL. It displays the original English and translated Hindi side-by-side, maintaining text formatting. This feature makes specialized content, like government information, accessible to Hindi speakers.

6. CONCLUSION

The Language Translation System provides a comprehensive web-based solution for English to Hindi translation, featuring specialized government terminology and support for various file formats. It efficiently handles text and website content translations while maintaining high accuracy through custom dictionaries. With its user-friendly interface and effective translation history management, the system is well-suited for government-related translation tasks. Its modular architecture ensures scalability and maintainability, making it a reliable and valuable tool for organizations seeking accurate translation services.

REFERENCES

Sreelekha, S., & Bhattacharyya, P. (2017). *Phrase Pair Mappings for Hindi-English Statistical Machine Translation.* Statistical Machine Translation (SMT) with Augmented Lexical Resources. Enhancing parallel corpus with lexical resources improves translation quality, with incremental improvement in translation accuracy. Requires extensive lexical resources and manual effort.

Srivastava, S., & Tiwari, R. (2019). *Self-Attention Based End-to-End Hindi-English Neural Machine Translation.* Transformer-Based Neural Machine Translation (NMT). Self-attention mechanisms enhance translation performance and improve handling of long-range dependencies but demand substantial computational resources.

Parida, S., Bojar, O., & Dash, S. R. (2019). *Hindi Visual Genome: A Dataset for Multimodal English-to-Hindi Machine Translation.* Multimodal NMT Incorporating Visual Context. Visual context aids in resolving linguistic ambiguities, offering an enriched dataset for training models, but aligning visual and textual data presents complexity.

Mhaskar, S., et al. (2023). *VAKTA-SETU: A Speech-to-Speech Machine Translation Service in Select Indic Languages.* Cascaded ASR, MT, and TTS Models for Speech Translation. Effective speech-to-speech translation for English-Hindi and other pairs, with real-time translation capability, although challenges exist in handling diverse accents and speech variations.

Sreelekha, S., & Bhattacharyya, P. (2017). *Phrase Pair Mappings for Hindi-English Statistical Machine Translation.* Creation of Lexical Resources for SMT. Augmenting training corpus with lexical resources improves translation quality, enhancing vocabulary coverage but requiring manual effort for resource creation.

Srivastava, S., & Tiwari, R. (2019). *Self-Attention Based End-to-End Hindi-English Neural Machine Translation.* Transformer-Based NMT with Self-Attention. Self- attention mechanisms improve translation accuracy, enabling better handling of context and dependencies but with high computational requirements.

Parida, S., et al. (2019). *Hindi Visual Genome: A Dataset for Multimodal English-to-Hindi Machine Translation.* Multimodal Dataset Creation for NMT. Incorporating visual context enhances translation disambiguation, providing a rich resource for training but facing complexity in dataset alignment.

Mhaskar, S., et al. (2023). *VAKTA-SETU: A Speech-to-Speech Machine Translation Service in Select Indic Languages.* Deployment of SSMT System. Effective real- time speech translation for multiple language pairs, beneficial for government and public use, though deployment challenges arise in diverse linguistic settings.