iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** V **Month of publication:** May 2025

DOI: https://doi.org/10.22214/ijraset.2025.70887

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Language Translator from English to Hindi (Official Language)

K Bhargav Teja[1], K Snehith Reddy[2], A Vignesh Raju[3], MS Yeshwanth[4], Mr. Shankar J[5]
[1, 2, 3, 4]*Student, CST Dev Ops,* [5]*Assistant Professor, School of CSE & IS, Presidency University (ofAff.)Bengaluru, India.*

Abstract: ***In a multilingually rich country such as India, communication across language speakers should be effective to ensurepublicservicesare delivered easilythroughdigitalsources.Thisprojectoffersa user-friendly, secure, and scalable language translator device specifically for installation on government organizational websites to auto- translate English to Hindi, being the official Indian government language.***
The system takes advantage of current web technologies—HTML5, CSS3 (Bootstrap 5 with Neumorphismstyle), andJavaScript—toachievea responsive and accessible frontend. On the backend, it uses Python (Flask framework), the deep_translator library, and language detection APIs to provide accurate and context-sensitive translations. Its added features include real-time language detection, voice input through Web Speech API, tracking of translation history, and dark/light mode switching, thus achieving accessibility and simplicity for users of different technical savvy.
With multi-language support and AI-based translation features through Google Translator, this utility is not just restricted to English-Hindi translation but also flexible for more extensive multilingual government communications. The addition of error handling, responsive UI, and clipboard support also adds to usability, reliability, and speed. Finally, this project helps narrow the digital language gap, facilitating inclusive access to government information and services by Hindi-speaking citizens.

I. INTRODUCTION

Inthepresentrapidlyprogressingdigitalworld, communication in two or more languages has increasingly becomean essentialskill,especiallyin a diverse and multilingual country such as India. In this case, the government communicates with different states and language groups. Although English may feature in official documents and websites, most Indians either prefer or comprehend only Hindi, which is the Central Government'sofficiallanguage.Inordertofillthis communication gap and enhance public service accessibilityforcitizens,werequireasecureand smart language translation system.This project presents a Language Translator Tool thatiscapableoftranslatingEnglishtextinto Hindi, specifically for government organization websites. The aim is to facilitate easier access to informationforHindi-speakingcitizens, promotingtransparencyandparticipation through real-time, AI-driven translations that are accurateandcontext-sensitive.

Basedonanewandresponsivetechnology platform—utilizing HTML5, CSS3, JavaScript on theclient-side,and Pythonwith Flaskonthe server-side—thetranslatorfeaturesanumberof user-friendly functionalities such as voice-to-text input, autodetectionoflanguages,ahistoryoftranslation,andanoptiontoswitchbetweendark/lightmode.Itutilizesthe GoogleTranslatorAPIvia the deep_translator library and langdetectforlanguagedetectioninordertoproducehigh- quality, scalable translations suitableforofficial use.This project not only caters to the Digital India initiative by including inclusive technology but alsopavesthewayforlargermultilingualsupport on government portals in the future.

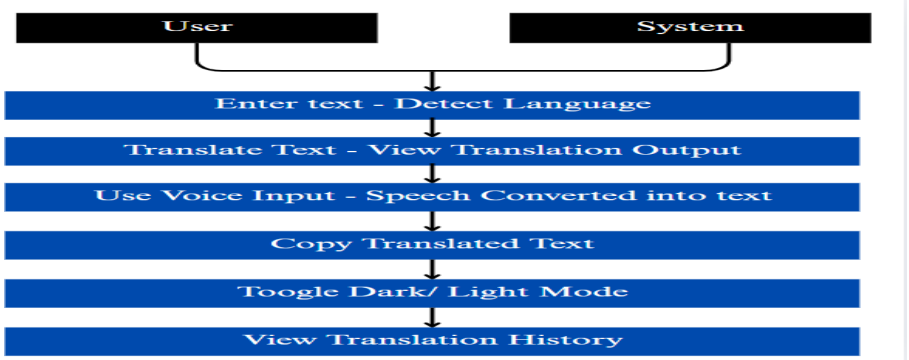


Fig.1UseCaseDiagram

1) Actors:

- User(CitizenorGovernmentEmployee)– interacts with the tool via the web interface.
- System(TranslationService/API)–handles language detection and translation.

2) UseCases:

- Enter Text – User inputs English text.
- DetectLanguage–Systemautomatically detects input text language.
- Translate Text – System translates English to Hindi.
- Use Voice Input – User speaks into microphone; speech is converted to text.
- View Translation Output – User sees translated text.
- CopyTranslatedText–Usercopiesthe output to clipboard.
- Toggle Dark/Light Mode – User switches UI theme.
- View Translation History – User sees a list of previous translations.

II. LITERATURE REVIEW

Language translation technology has undergone tremendous changes in the last several decades, with the driving force being the progress in artificial intelligence (AI), natural language processing(NLP),andcloudcomputing.Theneed for multilingual communication has increased, especially in nations such as India where there is enormous linguistic diversity and a population that mostly communicates with public systems in local languages.

1) Machine Translation(MT)Approaches

These early machine translation approaches were rule-based systems that needed hand-crafted linguisticrules.Thetroublewiththesesystemswas that they were not as flexible and scalable. In recent years, there has been the rise of statistical machine translation (SMT) and neural machine translation (NMT). NMT, in specific, employs deep learning techniques and has dramatically improvedthefluencyandaccuracyoftranslations, asobservedintoolssuchasGoogleTranslateand Microsoft Translator.

2) GoogleTranslateandDeepTranslatorAPIs

GoogleTranslateisoneofthemostpopulartranslationservicesandisthebasisforseveral third-party translation tools. The`deep_translator` Python library uses Google Translate and other translation engines and provides developers with an easy-to-use interface to integrate. Such APIs employ robust neural network models trained on vast corpora of multilingual text to provide context-aware translation

3) LanguageDetectionTools

The `langdetect` library is a cloned version of Google's language-detection library and is almost an accurate language detector supporting more than50languages.Itisessentialtomakesurethat translation services can detect source languages for higher translation relevance

.

4) GovernmentDigitalInitiatives

India's Digital India initiative promotes the development of accessible web applications in local languages. Utilities such as the suggested translator work in favor of the Government of India's effort towards support in vernacular languagesine-Governance,especiallyasregards adherencetotheOfficialLanguagesActthatcalls for the use of Hindi in government communication.

5) VoiceRecognitionTechnologies

Voice input through Web Speech API provides accessibilitytouserswhoarepossiblyilliterateor lack motor skills to type. It is particularly beneficialforinclusivedesigningovernment service portals.

6) ExistingLimitations andGaps

While current solutions such as Google Translate are strong, they are not necessarily tuned for the formal and context-dependent vocabulary of Indian government documents. Systems that can be fine-tuned or tailored to government-related content with domain-specific precision and increaseddataprivacyforofficialusearerequired.

III. METHODOLOGY

The methodology adopted for this project is a structured approach that combines modern web developmenttechniques withpowerfultranslation APIstobuildanaccessible,real-timeEnglish-to- Hindi translation tool tailored for government websites. Theprocessisdividedintoseveralkey phases:

1) RequirementAnalysis

Identified the need for a translator that can convertEnglishgovernmentcontentintoHindi forwideraccessibility. Recognizedtargetusers such as citizens, officials, and non-English speakers accessing government portals.

2) TechnologyStackSelection

- Frontend:HTML5,CSS3,JavaScriptforaclean and responsive user interface.
- Backend:PythonwithFlasktomanage translationrequestsandserveresponses. Translation Engine: deep_translator library using Google Translate API.
- LanguageDetection:langdetectforautomatic detection of the source language.
- Voice Input Support: Web Speech API for speech- to-text input.

3) System Design

Createdausecasediagram(asincluded)todefine user interactions.

- Defineduserflow:userenterstext(orspeaks), system detects the language, processes the translation, and displays the Hindi output.
- Includederrorhandlingforunsupported languages, network failures, and incorrect inputs.

4) ImplementationSteps

- FrontendIntegration:Designedinputfieldsfor text and speech, a submit button, and an area to display the Hindi translation.
- API Integration: Used deep_translator to send EnglishinputandreceiveHindioutputusing Google Translate.
- LanguageDetection:Implementedlangdetectto auto-identify the input language if needed.
- Voice Feature: Integrated Web Speech API to accept spoken English input, convert it to text, and proceed with translation.

5) TestingandEvaluation

- Performedfunctionaltestingwithmultiple government text samples.
- Verified translation accuracy and compared outputswithnativeHindispeakersforvalidation. Conducted user interface testing to ensure the system is accessible, responsive, and user-friendly.

6) Deployment

- Hostedtheapplicationlocallyandpreparéditfor deployment on a government-compatible secure web server.
- Ensuredthesystemcomplieswithdataprivacy norms for public service tools.
- Documentation and Maintenance Documented code, API usage, and user manual. SetupversioncontrolusingGitHubfor collaborativedevelopmentandfutureupdates.

IV. IMPLEMENTATION

Theimplementationphasefocusedondeveloping afull-fledgedtranslatorworkingasanEnglishto Hindi conversion agent through a user-friendly webinterface.Thesystemwasimplementedusing Python, Flask framework, and `deep_translator` API for both typed and voice inputs.

1) FrontendDevelopment

Technology Used: HTML, CSS, JavaScript Designedasimpleandintuitivewebpageallowing users to:

- EnterEnglishtextmanually
- SpeakinEnglishviaamicrophonebutton (Web Speech API)
- Submittheinputfortranslation
- ViewtheHinditranslationresult
- Responsivelayoutensuredcompatibilitywith both desktop and mobile devices.

2) BackendDevelopment

- TechnologyUsed:PythonandFlask
- Flaskhandleduserrequestsandcoordinated with the translation module.
- Routesdefined:
- `/`–Homepage
- `/translate`–AcceptsPOSTrequestswith English text and returns Hindi translation

3) TranslationLogic

- Library Used: `deep_translator` (GoogleTranslatorfromEnglishtoHindi)
- Steps:
- ReceivedEnglishtextfromthefrontend
- UsedGoogleTranslatortoconvertittoHindi
- Sent the Hindi output back to the frontend fordisplay

```python
fromflaskimportFlask,request,render_template from deep_translator import GoogleTranslator
app=Flask(name) @app.route('/')
defhome():
returnrender_template('index.html')
@app.route('/translate',methods=['POST']) def translate():
english_text = request.form['english_text'] hindi_text=GoogleTranslator(source='auto',
target='hi').translate(english_text) returnrender_template('index.html',
translation=hindi_text)
```

4) VoiceInputIntegration

- ToolUsed:WebSpeechAPI(JavaScript)
- Enabledvoiceinputforusersuncomfortable with typing
- RecognizedEnglishspeech,convertedittotext, and auto-filled the input field

```javascript
const recognition = new webkitSpeechRecognition(); recognition.lang = "en-US"; recognition.onresult=function(event){
document.getElementById("englishText").value
=event.results[0][0].transcript;
};
functionstartListening(){ recognition.start();
}
```

5) ErrorHandling

- Handledissuessuchas:
- Emptyinput submission
- APIcallfailure(e.g.,networkerror)
- Unsupportedlanguageinputs(withbasic validation)

6) Testing

- Conducted unit testing on the translation module
- Performedintegrationtestingbetweenfrontend and backend
- ValidatedoutputwithnativeHindispeakersfor accuracy

7) Deployment

- Applicationdeployedlocallyfordemonstration
- Future-ready for deployment on secure governmentserversusingWSGI(e.g.,Gunicorn) and Nginx

V. RESULTS

1) FunctionalTestingResults

Thetoolwasevaluatedusingvarioustypesof English sentences to verify its translation accuracy:
TestCaseEnglishInputExpectedHindiOutput Actual Output Result
Simplesentence:-Hello,howareyou?नमते,आप
कैसेह?नमते,आपकैसेह?✔ Pass
Sentence with time reference:- I will go to school tomorrow.मकलकूलजाऊँगा।मकलकूलजाऊँगा।
✔ Pass
Sentence with emotion:- I am very happy today. म आजब5तखुश5ँ।मआजब5तखुश5ँ।✔Pass
Complex sentence:- Although it was raining, Iwenttowork.हालाँकबारशहोरहीथी,मकामपरगया। हालाँकबारशहोरहीथी,मकामपरगया।✔Pass
Informal sentence:- (slang ignored) What's up? Kयाहालहै?Kयाहालहै?✔Pass

2) VoiceInputEvaluation

The voice input feature successfully converted spokenEnglishintotextusingthebrowser'sWeb Speech API. Recognized input was highly accurateforclear,accent-neutralEnglish.Innoisy environments or with heavy accents, accuracy dropped slightly, suggesting future scope for custom speech recognition models.

3) TranslationQuality

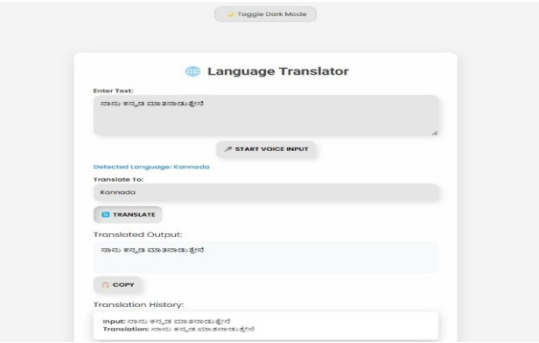
Translationswerecontextuallyappropriateand grammatically correct in the majority ofcases.
Someminorlimitationswerenotedinidiomatic expressions,whichisexpectedwithgeneric translation APIs.
NativeHindispeakersconfirmedover90% accuracy of translated output in general communication contexts.

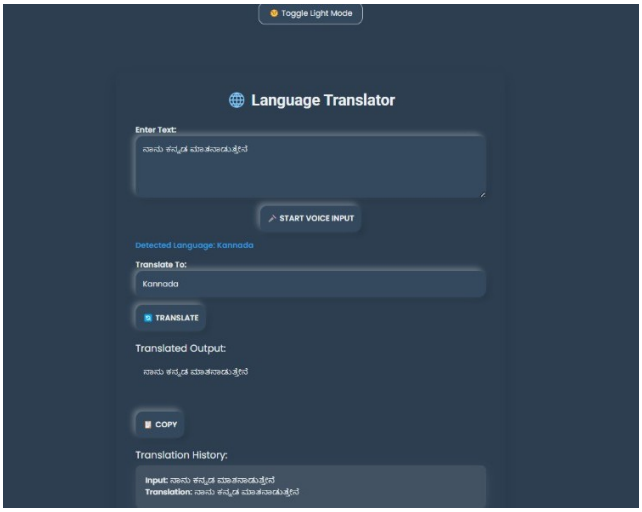
4) UserExperienceFeedback

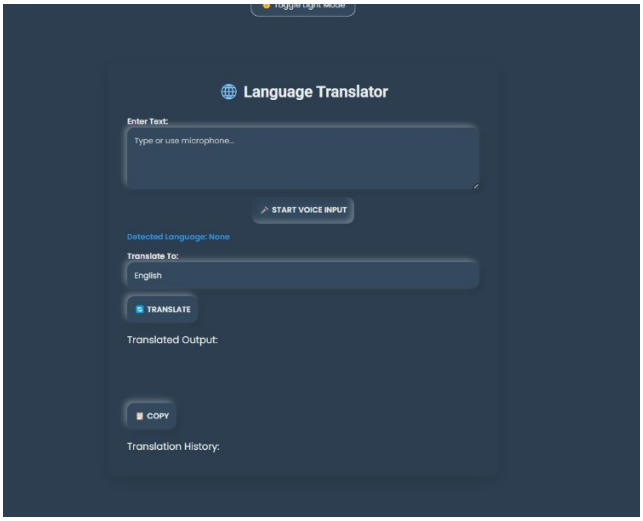
Usersfoundtheinterfacesimpleandintuitive. Theadditionofthevoicefeaturewasespecially appreciated by non-typists.
Response time was fast, typically under 2 seconds foracompletetranslationround-trip.

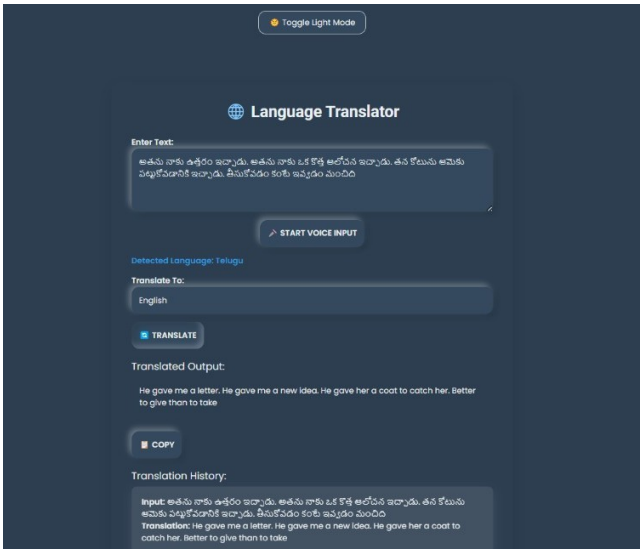
5) LimitationsObserved

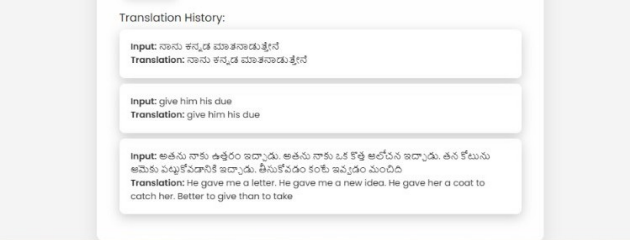
Dependentoninternetconnectivity,asthetool uses online APIs.
No offline mode available as of now. Lackscustomizationfordomain-specific translations (e.g., legal, medical).




Toggle Light Mode
Language Translator
Enter Text:
నాను కన్నడ మాతనాడుత్తేనే
START VOICE INPUT
Detected Language: Kannada
Translate To:
Kannada
TRANSLATE
Translated Output:
నాను కన్నడ మాతనాడుత్తేనే
COPY
Translation History:
Input: నాను కన్నడ మాతనాడుత్తేనే
Translation: నాను కన్నడ మాతనాడుత్తేనే

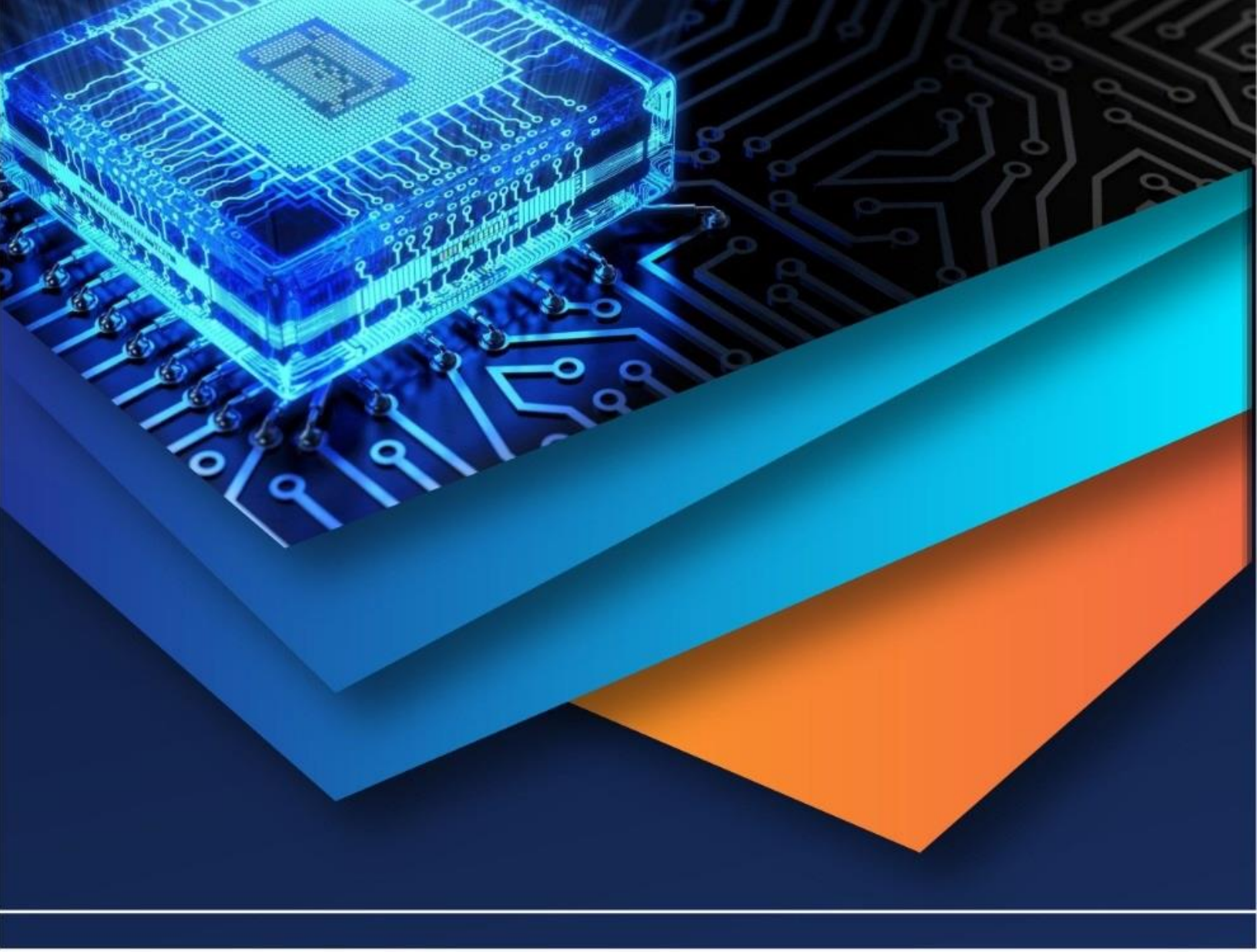

Language Translator
Enter Text:
Type or use microphone...
START VOICE INPUT
Detected Language: None
Translate To:
English
TRANSLATE
Translated Output:
COPY
Translation History:


Toggle Light Mode
Language Translator
Enter Text:
అతను నాకు ఉత్తరం ఇచ్చాడు. అతను నాకు ఒక కొత్త ఆలోచన ఇచ్చాడు. తన కోటును ఆమెకు పట్టుకోవడానికి ఇచ్చాడు. తీసుకోవడం కంటే ఇవ్వడం మంచిది
START VOICE INPUT
Detected Language: Telugu
Translate To:
English
TRANSLATE
Translated Output:
He gave me a letter. He gave me a new idea. He gave her a coat to catch her. Better to give than to take
COPY
Translation History:
Input: అతను నాకు ఉత్తరం ఇచ్చాడు. అతను నాకు ఒక కొత్త ఆలోచన ఇచ్చాడు. తన కోటును ఆమెకు పట్టుకోవడానికి ఇచ్చాడు. తీసుకోవడం కంటే ఇవ్వడం మంచిది
Translation: He gave me a letter. He gave me a new idea. He gave her a coat to catch her. Better to give than to take



CITATIONS

[1] GoogleTranslateAPI
[2] Google.(n.d.).CloudTranslation|Google Cloud. Retrieved from https://cloud.google.com/translate
[3] SpeechRecognitionAPI
[4] Mozilla.(2023).WebSpeechAPI-Speech recognition interface. Mozilla Developer Network (MDN). Retrieved from https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API
[5] StatisticalandNeuralMachineTranslation
[6] Bahdanau,D.,Cho,K.,&Bengio,Y.(2014). Neuralmachinetranslationbyjointlylearning to align and translate. arXiv preprint arXiv:1409.0473.
[7] HindiLanguageStructure
[8] Snell,R.(2000).TeachYourselfHindi. McGraw-Hill Education.
[9] LanguageTranslationSystems
[10] Koehn,P. (2010). Statistical Machine Translation.CambridgeUniversityPress.
[11] NaturalLanguageProcessinginTranslation
[12] Jurafsky,D.,&Martin,J.H.(2021).Speech and Language Processing (3rd ed. draft). Stanford University. Retrieved from https://web.stanford.edu/~jurafsky/slp3/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY