iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69523

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 IV Apr 2025- Available at www.ijraset.com

Language Translator with Machine Learning (English-French)

Srisailanath[1], Nikhil P R[2], P Venkata Rakesh[3], Paniraj T[4], Polishetty Laxmi Sanjana[5]
Dept. of Computer Science and Engineering, Ballari institute of Technology and Management, Ballari, India

Abstract: ***In today's globalized world, the need for efficient language translation tools has become critical. This project aims to develop a "Language Translation System" by integrating GNMT (Google Neural Machine Translation) Translation API and a user-friendly Streamlit interface. The system leverages advanced deep learning models trained on extensive datasets to provide accurate and context-aware translations. By focusing on accessibility and precision, the project seeks to bridge linguistic barriers and facilitate seamless communication. Users can input English text, select the desired target language, and receive real-time translations, ensuring ease of use for all.***

I. INTRODUCTION

In today's, language translation has become an essential point with which to break through the barriers in communication, not only with different regions but also many areas of culture. From business talking to personal communication, translation of the written word becomes essential for a much-globalized society. However, the translation can become tedious when done manually, and secondly, this will surely lead to error as well; it only emphasises the need for an automated solution that gives fast and accurate responses in translating texts.

This project aims to address the polymer failure, with the use of the power of cloud-based neural machine translation for high quality English-to-French translations. The application, intended to be user-friendly and seamless, integrates highly accurate machine learning through Google Neural Machine Translation (GNMT). It is done with Streamlit and is highly intuitive to use; users can simply input their text and receive a translation instantaneously-a great improvement for the accessibility and usability for a wider audience.

Utilizing Google Cloud's Translation API, the system is able to employ a custom-trained model based on an AutoML framework that has been trained across 79,625 pairs of sentences, eventually resulting in an impressive BLEU score of 51.428. Thus, translation works with high accuracy. The primary objective is the creation of a link through different languages, which should subsequently establish a single language enabling fast, effective, and swift cross language communication.

II. METHODOLOGY

A. Block Diagram

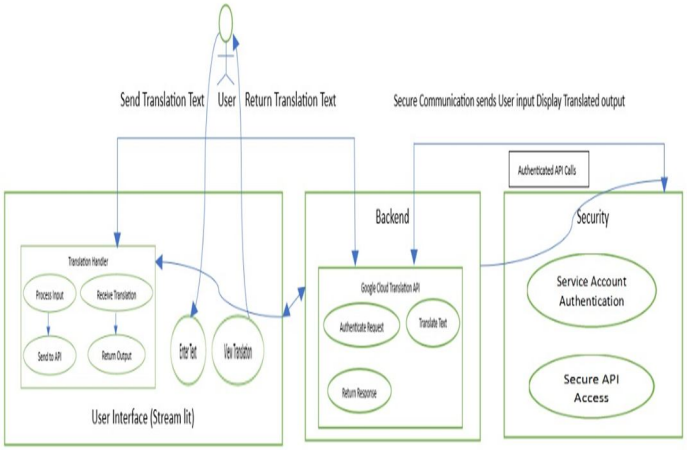


Figure 1: Workflow of Machine Translation Using GoogleCloudAPI

An English-to–French translation service is provided by the system through a cloud-based machine translation model. The software's components are interconnected to collect user input, interpret it, and translate it safely and efficiently.

B. Layers of the Architecture

1) User Interface : The front-end part uses Streamlit; a Python-based framework and has been designed as a single-page application. It allows users to input text in English and then click a button for translation through the interface, so the result can be viewed in French.

2) Application Layer (Back-End): The back end is done in Python with Streamlit and Google Cloud Translation API. It takes user inputs, translates them, and displays the result. It includes services like text processing input, sending to the API, and showing the translated content. The whole authentication process is overseen through a service account of Google Cloud, thus keeping the connection secure.

3) Data Layer: This means that the system makes use of Google Cloud Translation API, eliminating the need for a local database for storing translation models that have been pre-trained as well as processing end-user inputs into the cloud of Google.The translation request is sent and the processed output retrieved and displayed on the UI.

4) External Integrations: The integration of Google Cloud Translation API is adopted into the system for translation and a service account for security authentication as well. Other Python libraries will be required to encode and decode HTML, where necessary.

C. Deployment and Hosting

The application is ideally hosted on Google Cloud Platform (GCP) where App Engine is used for deployment while having the service account's credentials stored securely with GCP Secret Manager. Otherwise, it can still be hosted on Amazon EC2 if preferred.

D. Interaction and Communication

Users usually interact with the system through a web browser where they would feed in text for translation via Streamlit interface. The backend is contacted through function calls via the frontend, and translation happens using Google Cloud Translation API. As for internal use, the system securely sends and receives translation-related data via the Google Cloud SDK thereby offering real-time response.

E. System Design Patterns

This system has a monolithic architecture. Translation handling is centralized in a single web application. The system accounts for the use of Google Cloud service account authentication and HTTPS for API communication as the categorization of security measures. It is scalable in that it supports vertical scaling (adding capacity to an instance) and horizontal scaling (adding distributed instances in the cloud).

F. Technology Stack

The Streamlit (Python) works on the interface, whereas the backend integrates Python with the Google Cloud Translation API. Security applies Google Cloud authentication using JSON key files. In regard to DevOps tools, these include Docker for containerization, if necessary, and CI/CD pipelines with GitHub Actions or Jenkins. Monitoring and logging are done through Google Cloud Logging and Streamlit's built-in debugging options.

G. Future Considerations

In the future, enhancements would include adding Spanish and German to the list of supported languages and a translation history option to maintain records of prior translations. Increased usage would be tackled using Google Cloud auto-scaling. AI-based enhancements such as fine-tuned translation models are also proposed for greater accuracy.

Flowchart diagram

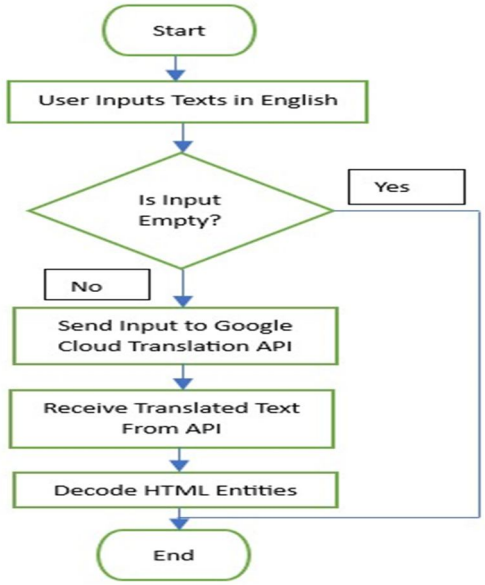


Figure 2: Decision Flow of Language Translation System

A project which translates The project provides a language translation system, converting English text to French using Google NMT under AutoML. It has a web interface created in Streamlit wherein the text is input by the user. This text is then sent to Google Cloud Translation API for a real-time translation. API access is secured via a service account, while the model is guaranteed to be accurate, reflecting a BLEU score of 51.428. The entire system is set up on Google Cloud, making it instantly scalable and extendable for future applications to other languages.

III. WORKING

This use case diagram is for the high-level overview of functions in the Text Translator System accessible to a user. Actions on this diagram are directed toward the capability of translating English text into French efficiently with the help of cloud-based translation service.

The diagram shows that the User has the capability to:

1) Enter Text (English): The user inputs text in English that they wish to translate into French. This is the primary action that initiates the translation process.

2) Select Language (French): The system allows the user to choose French as the target language for translation. This step ensures that the translation output aligns with the user's requirements.

3) Translate to French: Once the input text and language are specified, the system processes the request using the translation model to convert the English text into French accurately.

4) Display Translated Text (French): The translated output in French is presented to the user in a clear and readable format, ensuring ease of use and comprehension.

This use case diagram highlights the streamlined workflow of the Text Translator System, ensuring a user-friendly and efficient experience. The system is designed to provide quick and accurate translations without storing any user data, maintaining privacy and security. The user is empowered with simple yet powerful features to break language barriers effectively.

A. Algorithm

The Sequence-to-Sequence (Seq2Seq) variant of GNMT model is the deep learning model for language translation. The model consists of an encoder which converts the English text into a numerical representation and a decoder which transforms this into the translated French output. Google Neural Machine Translation (GNMT) enhances this architecture with an attention mechanism, making the model more accurate since it will focus on the important words. The words were sequentially predicted by this Softmax layer, giving a fluent translation. Finally, it post-processes the translation and displayed it onto the Streamlit UI.ReLU Activation:

Applies non-linearity to the output of the convolutional layers, allowing the model to learn complex features.

1) User Input Handling: Accepts English text in a Streamlit-based interface, ensuring non-empty input before processing
2) Google Cloud Authentication: Loads the service account JSON key to securely authenticate with the Google Cloud Translation API.
3) Text Preprocessing: Cleans and encodes the input text for compatibility with the translation API.
4) Translation Processing: Sends the input text to the Google NMT model for translation via the Seq2Seq approach.
5) Post-Processing: Decodes HTML entities in the translated output to render it properly.
6) Displaying Translated Output: Displays the French translation onto the Streamlit front end, allowing for entry of any new text by the user.
7) End Process: Ends the process after showing the translation, thus allowing the user to restart or exit.

IV. RESULTS AND DISCUSSION

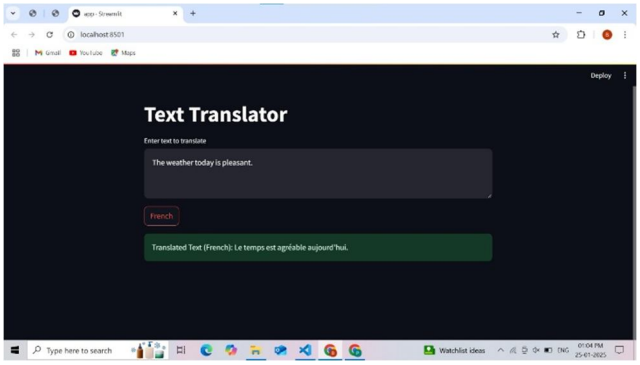


Figure 3: Streamlit-Based Text Translator Interface

V. CONCLUSION

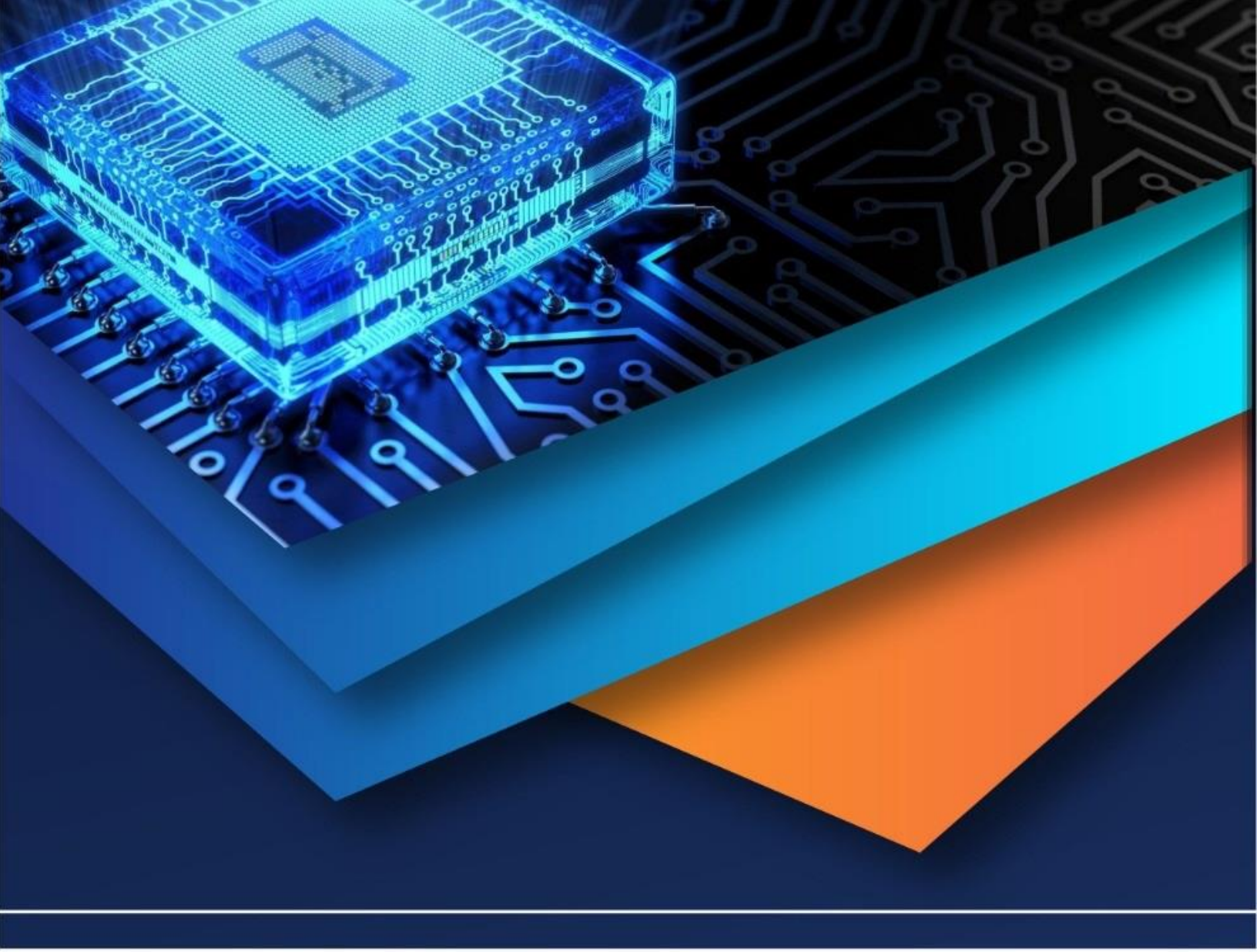
In summary, this project successfully develops a web interface using Python through which the GNMT from Google Cloud can be integrated into a real-time translator. The real-time translations of the application are made possible through the Google Cloud Translation API, and they are as accurate as they are context-related from English to French.

The application is built using Streamlit, which provides a friendly experience for users interfacing with the application while it processes their translation requests efficiently. With robust error handling, the system is able to show increased reliability and stability.

The modular design enables scalability by keeping it open to future enhancement possibilities such as other languages or speech-based features. It thus showcases together the practical application of artificial intelligence in minimizing language barriers for greater global communication through accessible and reliable technology.

REFERENCES

[1] Somya Singh, Ishu Rattan, Darshan Kaur. "Language Translation Using Machine Learning " International Journal of Computer Applications, ISSN: 2320-2882, Volume 12, Issue 4 April 2024.
[2] Srivani EN, Bhavana. "Language Translation Platform on Machine Learning," International Journal of Research Publication and Reviews, ISSN 2582-7421, Vol (5), Issue (5), May (2024), Pages 3280-3283.
[3] Geetha, K., Priyakumari, M. "Machine Learning Based On Multilingual Translation." International Journal of New Innovations in Engineering and Technology, ISSN: 2319-6319l, Vol. 24, Issue 1, Mar. 2024, pp. 1421-1426
[4] Reballiwar L. V., Yergude S. B., Urade V. M., Birewar S. R., Karmarkar B. "Language Translation Using Machine Learning," International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), ISSN (Online) 2581-9429, Volume 3, Issue 1, December 2023, Pages 297-300.
[5] Kharat, V., Chaudhari, U., Kesarwani, K., "Regional Language Translator Using Neural Machine Translation," International Journal of Current Engineering and Scientific Research (IJCESR), ISSN (PRINT): 2393-8374 (ONLINE): 2394-0697 , Vol. 9, Issue 5, 2022.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY