# LANGUAGE TRANSLATION USING MACHINE LEARNING

**A Micro Project Report**

**Submitted by**


## ORUGANTI AMARNADH
## Reg.no:99220041612

## B.Tech- Computer Science and Engineering, Arificial Intelligence &Machine Learning




**Kalasalingam Academy of Research and Education**
**(Deemed to be University)**
**Anand Nagar , krishnankoil - 626126**


**MARCH 2024**

KALASALINGAM
ACADEMY OF RESEARCH AND EDUCATION
(DEEMED TO BE UNIVERSITY)
Under sec. 3 of UGC Act 1956. Accredited by NAAC with "A" Grade

SCHOOL OF COMPUTING

DEPATMENT OF COMPUTER SCIENCE AND ENGINEERING

# BONAFIDE CERTIFICATE

Bonafide record of the work done by Oruganti Amarnadh – 99220041612 in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Specialization of the Computer Science and Engineering, during the Academic Year Even Semester 2023-24


**Ms. K.Bavani**
**Project Guide**
**Assistant Professor**
**Computer Science**
**and Engineering**
**Kalasalingam Academy of**
**Research and Education**
**Krishnan kovil - 626126**

**Mr.D.Gnana Kumar**
**Faculty Incharge**
**Assistant Professor**
**Computer science and**
**Engineering**
**Kalasalingam Academy of**
**Research and Education**
**Krishnan kovil - 626126**

**Mr.R.Raja Sekar**
**Evaluator**
**Assistant Professor**
**Computer science**
**and Engineering**
**Kalasalingam Academy of**
**Research and Education**
**Krishnan kovil - 626126**

# Abstract

This project focuses on converting one language text into any language text by using machine learning. Text language conversion is a fundamental task aimed at transforming of machine learning language conversion systems. This report presents an overview of machine learning. This relies on the Google Translate API provided by the googletrans library. This is implemented using the Tkinter library in Python, the application allows users to input text in one language and receive its translation in another language of their choice. The interface features an input field for entering text, a dropdown menu for selecting the target language, and an output area an output area to display the translated text. Upon clicking the “Translate” button, the input text is sent to the Google Translate API, which utilizes the ‘googletrans’ library to perform the translation.

Contents

# List of Figures

# Chapter 1

# LANGUAGE TRANSLATION USING MACHINE LEARNING

## 1.1 LANGUAGE TRANSLATION USING MACHINE LEARNING

Text language conversion systems utilize sophisticated algorithms to analyze and understand the structure, semantics, and context of textual content in one language, and then generate an equivalent text in another language. These domains, including international business, diplomacy, education, and everyday communication.

## 1.2 Software Tool

A software tool is a computer program that is used to create, debug, maintain, or otherwise support other programs and applications. Software tools are referred to as software programming tools.

Examples of software tools include:

**Code editors**: Jupyter, Google Colab

### 1.2.1 How it is used

Jupyter is an open-source interactive development environment (IDE) for Python, R, Julia, and other programming languages. It allows users narrative text. Jupyter is typically , scientific computing, and machine learning.

## 1.2.2 Objective of the code

**Facilitate Cross-Lingual Communication:** The primary objective of the project is to develop a user-friendly tool that enables individuals to translate text from one language.

**Integration of Machine Translation Technology:** Leveraging machine learning and NLP techniques, the project aims to integrate advanced translation capabilities into the application. By utilizing the Google Translate API and the googletrans library, the project ensures accurate and reliable translation results.

**User-Friendly Interface:** The project focuses on designing an intuitive Tkinter. The interface should be easy to navigate, input text, select target languages, and view translated output seamlessly.

**Support for Multiple Languages:** Another objective of the project is to support translation between languages. By incorporating a dropdown menu for language selection, flexibility to translate text into their desired language from a comprehensive list of options.

**Real-Time Translation:** The project aims to provide real-time translation capabilities, allowing users to receive translated text promptly after inputting their text and selecting the target language. This ensures a smooth and efficient translation experience.

**Educational and Practical Use:** Finally, the project aims to serve both educational and practical purposes. It can be used by language learners to practice translation skills, by professionals , by individuals seeking quick translations for personal or business purposes.

# Chapter 2

# SYSTEM STUDY OF LANGUAGE TRANSLATION USING MACHINE LEARNING

## 2.1 Literature Survey

| S.NO | TITLE | AUTHOR | SUMMARY |
|---|---|---|---|
| 1 | Neural Machine Translation by Jointly Learning to Align and Translate | Bahdanau, D., Cho, K., & Bengio, Y | Introduces neural network architecture for machine translation that learns to align and translate simultaneously. |
| 2 | Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation | Wu, Y. et al. | Describes Google's neural machine translation system, which significantly improves translation quality over previous approaches. |

## 2.2Data Collection

The data is collected in the git hub and enrolled and done int the Jupyter Notebook.

### 2.2.1 Collection of data

In an established systematic fashion that enables one to answer stated research questions, test hypotheses, and evaluate outcomes

## 2.3 Algorithm

The algorithm used in the above code for text language conversion is the Google Translate API, which utilizes machine learning-based approaches for language translation. Specifically, the googletrans library interacts with the Google Translate API to perform the translation task.

Behind the scenes, the Google Translate API likely employs including neural machine translation (NMT), to translate. become the state-of- the- art approach for machine translation tasks due to their ability to capture complex linguistic patterns and context dependencies effectively

While the exact details of the Google Translate API are proprietary and not disclosed publicly, it is neural networks play a significant role in the translation process. These networks are trained on large datasets of parallel texts in multiple languages to learn the mappings between different language pairs and generate accurate translations..

## 2.4 Performance Metrics

Performance metrics are measurements or indicators that assess the effectiveness and efficiency of a system, process, or individual. In various domains, performance metrics help evaluate and optimize performance

## 2.4.1 Performance

When evaluating the performance of a language Translation in jupyter Notebook, several key performance metrics can be considered. These metrics provide insights into the accuracy, efficiency, and reliability of the Language Translation process.

## 1. Automated Metrics

**METEOR (Metric for Evaluation of Translation with Explicit Ordering)**: METEOR is another automated metric that considers precision, recall, and alignment of n-grams between the machine translation and reference translations. more robust to synonyms and paraphrases.

**TER (Translation Edit Rate)**: TER measures the transform the machine-translated text into the reference translation. fluency and adequacy of the translation.

## 2. Human Evaluation

**Human Judgments**: Human evaluators assess the quality of translations based on fluency, adequacy, and overall accuracy. Evaluators can be native speakers or bilingual individuals proficient in both the source and target languages.

**Error Analysis**: Human evaluators identify common errors in the translations, such as mistranslations, grammatical errors, and omissions. They provide qualitative feedback to improve the translation quality.

# Chapter 3

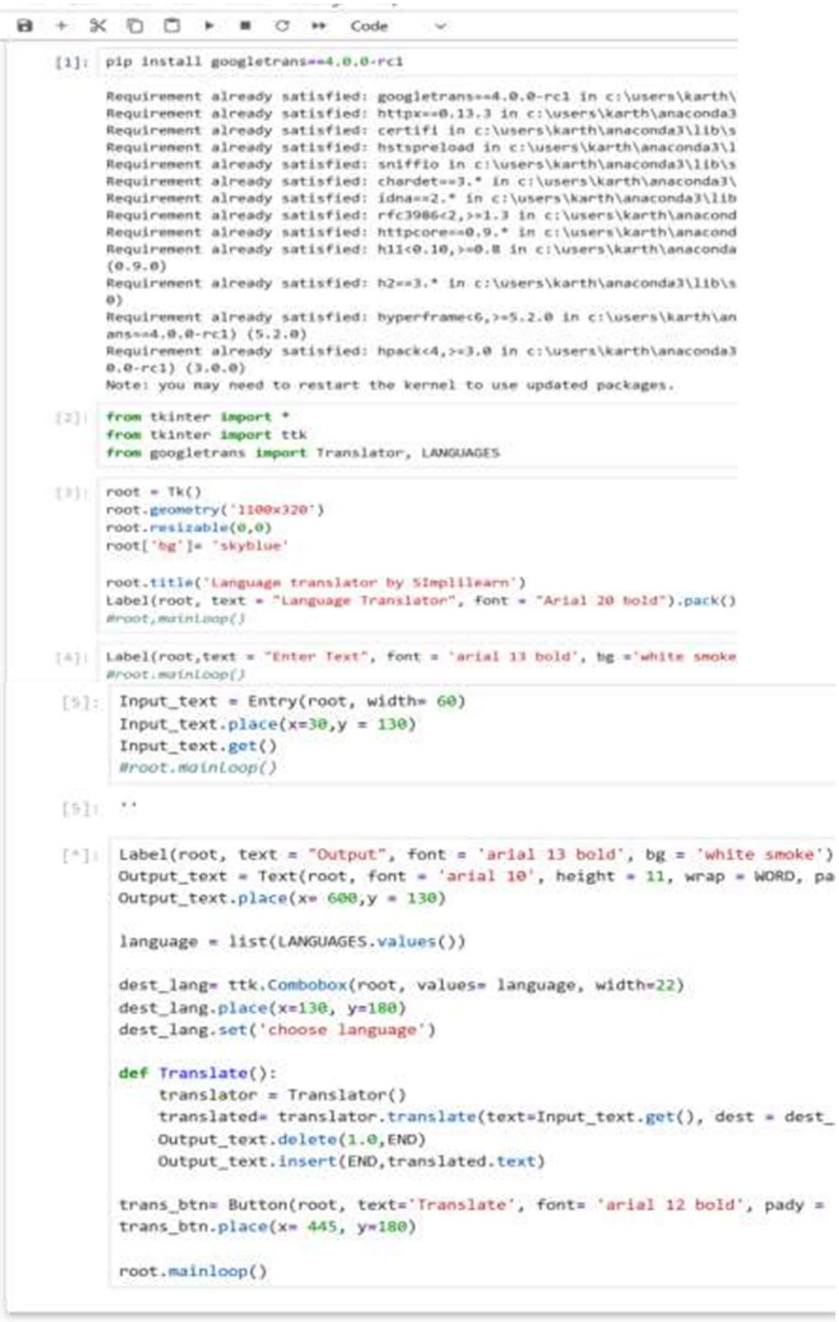
# Implementation

## 3.1 CODE

```
pip install googletrans==4.0.0-rc1
from tkinter import *
from tkinter import ttk
from googletrans import Translator, LANGUAGES
root = Tk()
root.geometry('1100x320')
root.resizable(0,0)
root['bg']= 'skyblue'


root.title('Language translator by ------BEASTLY SYNDICATE------')
Label(root, text = "Language Translator", font = "Arial 20 bold").pack()
Label(root,text = "Enter Text", font = 'arial 13 bold', bg ='white
   smoke').place(x=165, y=90)
Input_text = Entry(root, width= 60)
Input_text.place(x=30,y = 130)
Input_text.get()
Label(root, text = "Output", font = 'arial 13 bold', bg = 'white smoke').place(x=
   780,y = 90)
Output_text = Text(root, font = 'arial 10', height = 11, wrap = WORD, padx= 5,
   pady= 5, width= 50)
Output_text.place(x= 600,y = 130)


language = list(LANGUAGES.values())
```

```
dest_lang= ttk.Combobox(root, values= language, width=22)
dest_lang.place(x=130, y=180)
dest_lang.set('choose language')

def Translate():
    translator = Translator()
    translated= translator.translate(text=Input_text.get(), dest = dest_lang.get())
    Output_text.delete(1.0,END)
    Output_text.insert(END,translated.text)

trans_btn= Button(root, text='Translate', font= 'arial 12 bold', pady = 5, command=
    Translate, bg='orange',activebackground='green')
trans_btn.place(x= 445, y=180)

root.mainloop()
```

## 3.1.1 code implementation

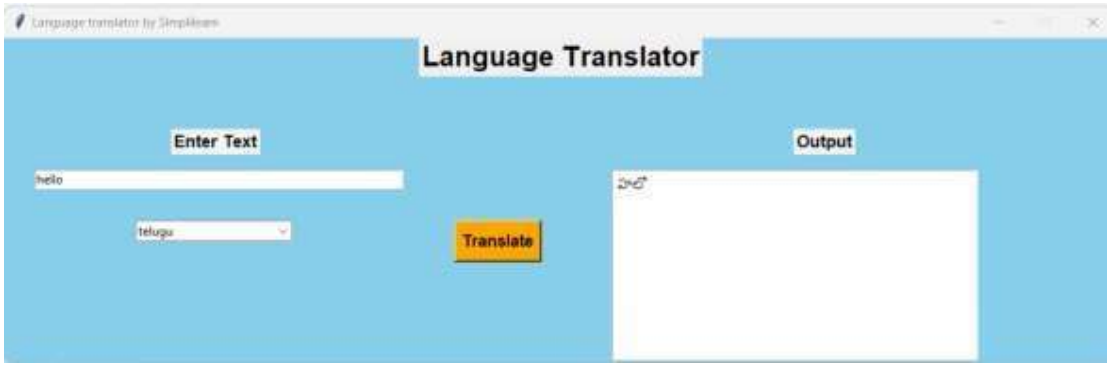


```
[1]: pip install googletrans==4.0.0-rc1

Requirement already satisfied: googletrans==4.0.0-rc1 in c:\users\karth\
Requirement already satisfied: httpx==0.13.3 in c:\users\karth\anaconda3
Requirement already satisfied: certifi in c:\users\karth\anaconda3\lib\s
Requirement already satisfied: hstspreload in c:\users\karth\anaconda3\l
Requirement already satisfied: sniffio in c:\users\karth\anaconda3\lib\s
Requirement already satisfied: chardet==3.* in c:\users\karth\anaconda3\
Requirement already satisfied: idna==2.* in c:\users\karth\anaconda3\lib
Requirement already satisfied: rfc3986<2,>=1.3 in c:\users\karth\anacond
Requirement already satisfied: httpcore==0.9.* in c:\users\karth\anacond
Requirement already satisfied: h11<0.10,>=0.8 in c:\users\karth\anaconda
(0.9.0)
Requirement already satisfied: h2==3.* in c:\users\karth\anaconda3\lib\s
0)
Requirement already satisfied: hyperframe<6,>=5.2.0 in c:\users\karth\an
ans==4.0.0-rc1) (5.2.0)
Requirement already satisfied: hpack<4,>=3.0 in c:\users\karth\anaconda3
0.0-rc1) (3.0.0)
Note: you may need to restart the kernel to use updated packages.

[2]: from tkinter import *
     from tkinter import ttk
     from googletrans import Translator, LANGUAGES

[3]: root = Tk()
     root.geometry('1100x320')
     root.resizable(0,0)
     root['bg']= 'skyblue'

     root.title('Language translator by SImplilearn')
     Label(root, text = "Language Translator", font = "Arial 20 bold").pack()
     #root.mainloop()

[4]: Label(root,text = "Enter Text", font = 'arial 13 bold', bg ='white smoke
     #root.mainloop()

[5]: Input_text = Entry(root, width= 60)
     Input_text.place(x=30,y = 130)
     Input_text.get()
     #root.mainloop()

[5]: ''

[*]: Label(root, text = "Output", font = 'arial 13 bold', bg = 'white smoke')
     Output_text = Text(root, font = 'arial 10', height = 11, wrap = WORD, pa
     Output_text.place(x= 600,y = 130)

     language = list(LANGUAGES.values())

     dest_lang= ttk.Combobox(root, values= language, width=22)
     dest_lang.place(x=130, y=180)
     dest_lang.set('choose language')

     def Translate():
         translator = Translator()
         translated= translator.translate(text=Input_text.get(), dest = dest_
         Output_text.delete(1.0,END)
         Output_text.insert(END,translated.text)

     trans_btn= Button(root, text='Translate', font= 'arial 12 bold', pady =
     trans_btn.place(x= 445, y=180)

     root.mainloop()
```

The above code was executed in the Jupyter Notebook.

## 3.2 Result

In this 1st output we can see that the input text is converted into TELUGU language.




In this 2nd output we can see that the input text is converted into TAMIL language.

# Chapter 4

# Conclusion and Future Work

## 4.1 Conclusion

The primary to translate from English to any other language. As many studies have focused on the English language and other European languages in the past, However, there aren't any existing studies done on the local language. So, the other languages has been chosen for this research. The neural network architecture used for this research.

## 4.2 Future Work

If you're considering future work for Language Translation in, here are some your **Application:**

### 1. Improved User Interface

1. Enhance the user interface as progress bars or loading indicators to provide feedback during the translation process.
2. Implement tooltips or help buttons to assist users in understanding the functionalities and usage of the application.

### 2. Language Detection

1. Integrate language detection functionality to automatically identify the language of the input text, eliminating the need for users to manually select the source language.

### 3. Optimization and Performance

1. Optimize the translation process for speed and efficiency, particularly for large volumes of text or simultaneous translation requests.
2. Explore caching strategies to store previously translated text and minimize redundant translation requests.

## 4. Multimodal Translation

1. Extend the application to support translation of multimedia content, such as images, audio, or video, in addition to text.
2. Investigate methods for integrating speech-to-text and text-to-speech functionalities for more versatile translation capabilities.

## 5. Customization and Personalization

1. Allow users to customize translation preferences, such as preferred language variants or specialized terminology dictionaries.
2. Implement user profiles or settings to save translation history, preferences, and personalized dictionaries across sessions.

## Reference


1) https://github.com/topics/language-translator.
2) https://data-flair.training/blogs/language-translation-machinelearning/.

# Appendix

## Certificate

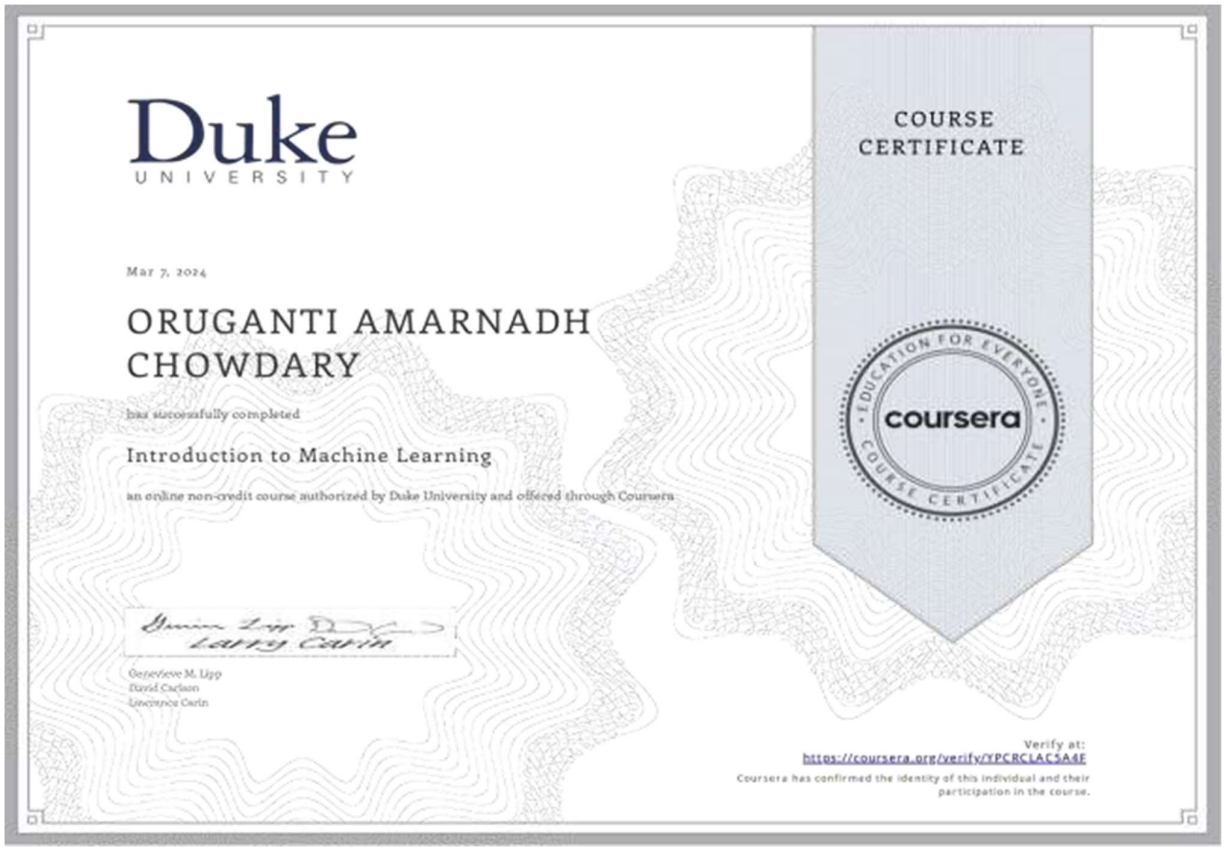



Figure 6.1: Certificate details