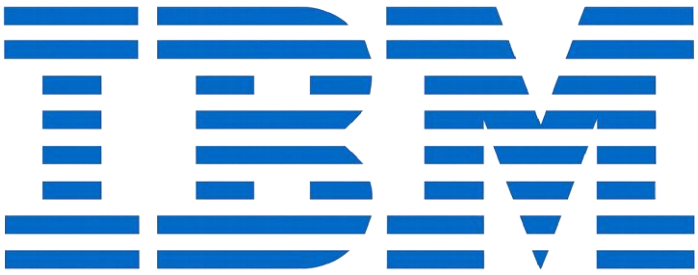
IBM PBEL

(Project Based Experiential Learning)

Project Report on

AI Powered-Multi-Language-Translator



Project Focus:
Generative AI
Mentor: Mr. Harendra Singh Rajpoot, IBM
Program: IBM PBEL (Project Based Experiential Learning)

Submitted to:
Mr. Harendra Singh Rajpoot
IBM

Submitted by :
Piyush Sain
Masters of Computer Applications 2nd year
Hi-Tech Institute of Engineering and Technology
Ghaziabad, UP

Acknowledgment

I would like to express my sincere gratitude to my project mentor, Mr. Harendra Singh Rajpoot, for his invaluable guidance and support throughout this project. His expertise in machine learning and natural language processing was instrumental in the successful completion of this fake news detection system.

Mr. Rajpoot's constructive feedback and insightful suggestions were crucial for improving the code's efficiency and the model's accuracy. His encouragement and availability for discussion helped me overcome various challenges and deepen my understanding of the concepts involved. This project would not have been possible without his continuous mentorship.

Introduction

Language barriers continue to be a major obstacle in communication across the world. With increasing globalization, real-time multilingual communication tools are essential in domains like healthcare, education, diplomacy, and business. Traditional translation systems often lack accuracy or fail to scale across diverse languages.

This project addresses this need by creating a multi-language translation tool powered by advanced AI models. It aims to:

- Support translation across more than 50 languages.
- Deliver near-human translation quality.
- Leverage open-source tools and pre-trained models.
- Enable future extensibility into speech-to-text and text-to-speech.

We used Google Colab for development and testing and Hugging Face Transformers to access state-of-the-art translation models.

References:

1. **Dataset**
    - Bisaillon, Clément. (2018). *Fake and Real News Dataset*. Retrieved from Kaggle: https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset
2. **Libraries**
    - **Scikit-learn:** Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830. Retrieved from https://scikit-learn.org/stable/
    - **Pandas:** The pandas Development Team. (2024). *pandas: Python Data Analysis Library*. Retrieved from https://pandas.pydata.org/

- **NLTK:** Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media. Retrieved from https://www.nltk.org/
- **spaCy:** Explosion. (2024). *spaCy: Industrial-strength Natural Language Processing*. Retrieved from https://spacy.io/

Tools And Technology used

Tool	Purpose
Google Colab	Development, testing, GPU acceleration
Hugging Face Transformers	Access to pre-trained multilingual models
Python	Programming language
Torch/TensorFlow	Backend frameworks
NLTK / spaCy	Optional for text pre/post-processing
Gradio/Streamlit	(Optional) UI for translation app

Why Google Colab?

- Free GPU/TPU.
- Easy integration with Hugging Face.
- No setup required.

Why Hugging Face?

- Repository of top-performing NLP models.
- Supports multilingual, zero-shot, and fine-tuned NMT models.

System Design

Architecture Overview:

1. **Input Layer**: Accepts source text and language code.
2. **Preprocessing**: Tokenization using model tokenizer.
3. **Model Inference**: MarianMT or mBART50 performs translation.
4. **Postprocessing**: Detokenization and formatting.
5. **Output**: Display translated text.

Workflow:

- User inputs: "Hello" → Source Language: English, Target: French
- Tokenizer converts to input IDs
- Model returns translation token IDs
- Output decoded and presented as: "Bonjour"

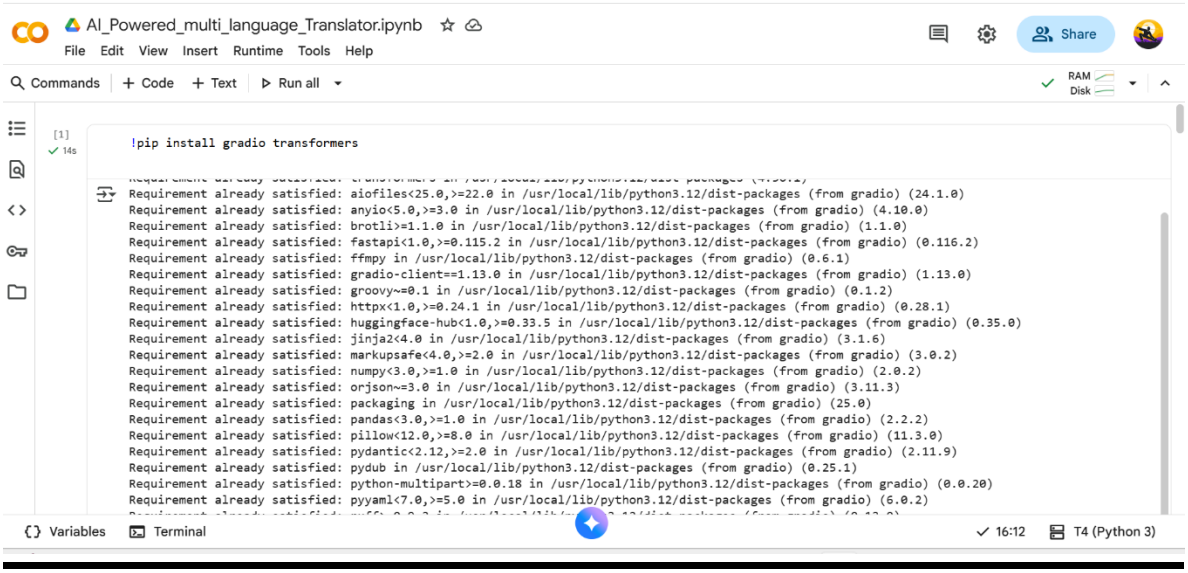
Model Selection:

- **MarianMT**: Best for specific language pairs.
- **mBA**
- **RT50**: Best for multilingual, zero-shot tasks.

- # Implementation

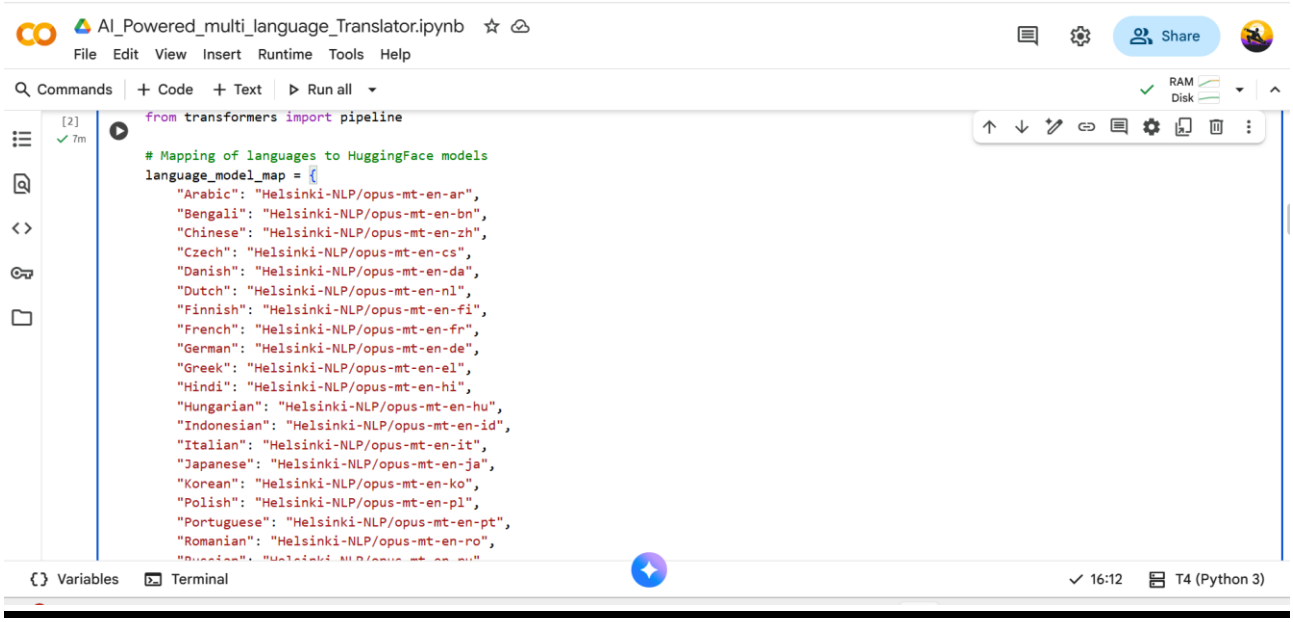
1

-

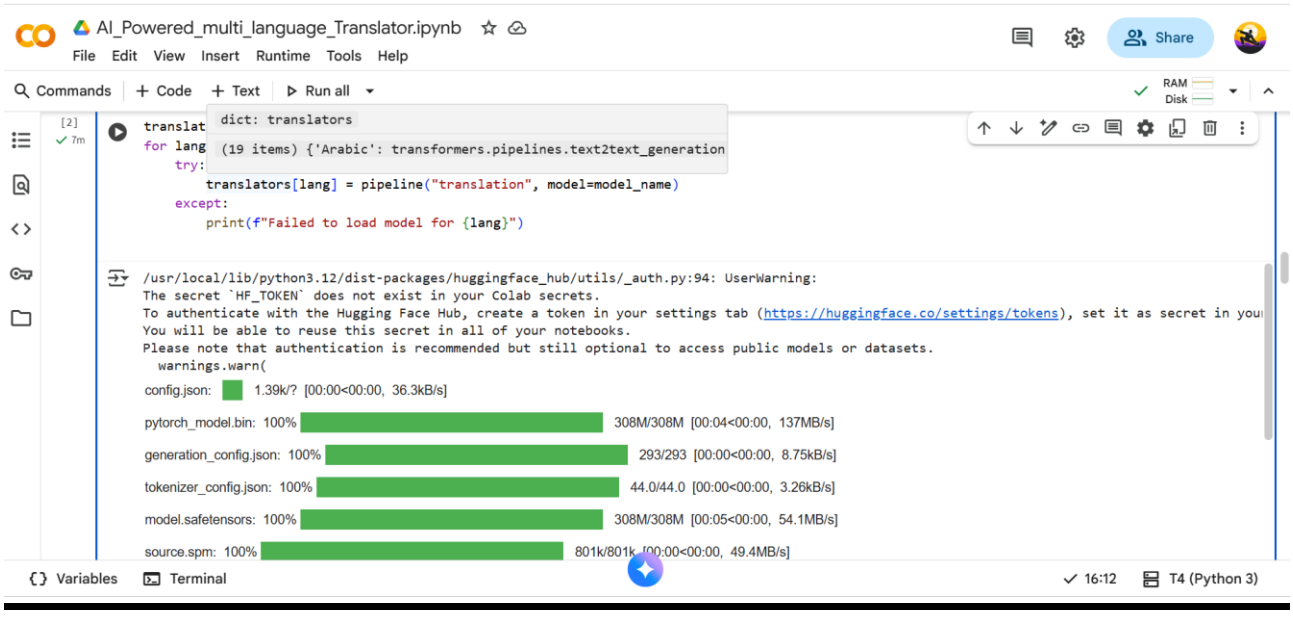


-

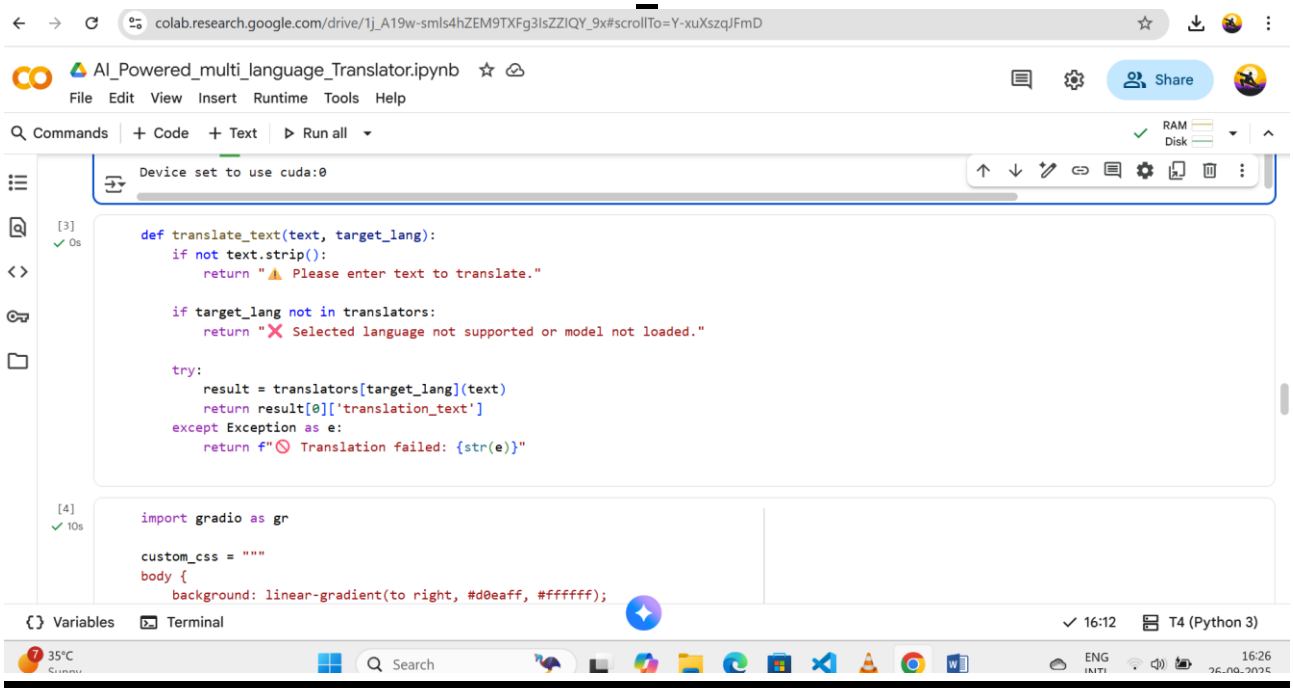
2


AI_Powered_multi_language_Translator.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
RAM Disk
[2] 7m
from transformers import pipeline
# Mapping of languages to HuggingFace models
language_model_map = {
"Arabic": "Helsinki-NLP/opus-mt-en-ar",
"Bengali": "Helsinki-NLP/opus-mt-en-bn",
"Chinese": "Helsinki-NLP/opus-mt-en-zh",
"Czech": "Helsinki-NLP/opus-mt-en-cs",
"Danish": "Helsinki-NLP/opus-mt-en-da",
"Dutch": "Helsinki-NLP/opus-mt-en-nl",
"Finnish": "Helsinki-NLP/opus-mt-en-fi",
"French": "Helsinki-NLP/opus-mt-en-fr",
"German": "Helsinki-NLP/opus-mt-en-de",
"Greek": "Helsinki-NLP/opus-mt-en-el",
"Hindi": "Helsinki-NLP/opus-mt-en-hi",
"Hungarian": "Helsinki-NLP/opus-mt-en-hu",
"Indonesian": "Helsinki-NLP/opus-mt-en-id",
"Italian": "Helsinki-NLP/opus-mt-en-it",
"Japanese": "Helsinki-NLP/opus-mt-en-ja",
"Korean": "Helsinki-NLP/opus-mt-en-ko",
"Polish": "Helsinki-NLP/opus-mt-en-pl",
"Portuguese": "Helsinki-NLP/opus-mt-en-pt",
"Romanian": "Helsinki-NLP/opus-mt-en-ro",
Variables Terminal
16:12 T4 (Python 3)

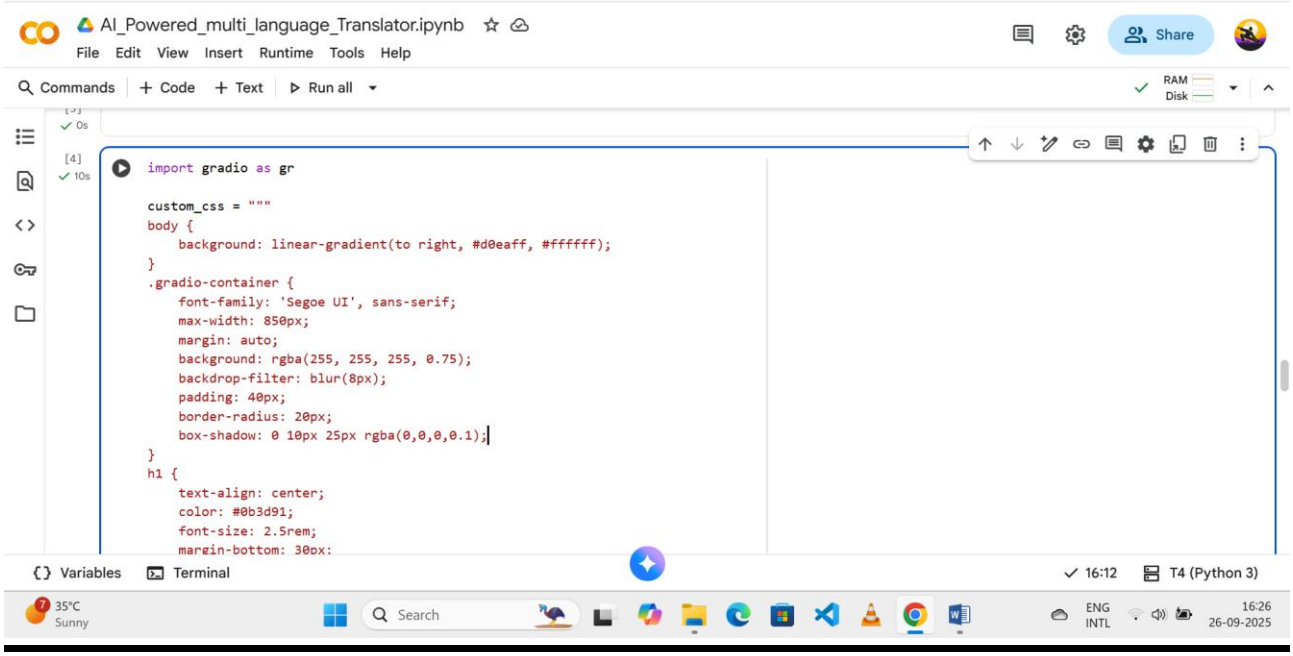
3


AI_Powered_multi_language_Translator.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
RAM Disk
[2] 7m
translat
dict: translators
(19 items) {'Arabic': transformers.pipelines.text2text_generation
for lang
try:
translators[lang] = pipeline("translation", model=model_name)
except:
print(f"Failed to load model for {lang}")
/usr/local/lib/python3.12/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secret in you
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
warnings.warn(
config.json: 1.39k/? [00:00<00:00, 36.3kB/s]
pytorch_model.bin: 100% 308M/308M [00:04<00:00, 137MB/s]
generation_config.json: 100% 293/293 [00:00<00:00, 8.75kB/s]
tokenizer_config.json: 100% 44.0/44.0 [00:00<00:00, 3.26kB/s]
model.safetensors: 100% 308M/308M [00:05<00:00, 54.1MB/s]
source.spm: 100% 801k/801k [00:00<00:00, 49.4MB/s]
Variables Terminal
16:12 T4 (Python 3)

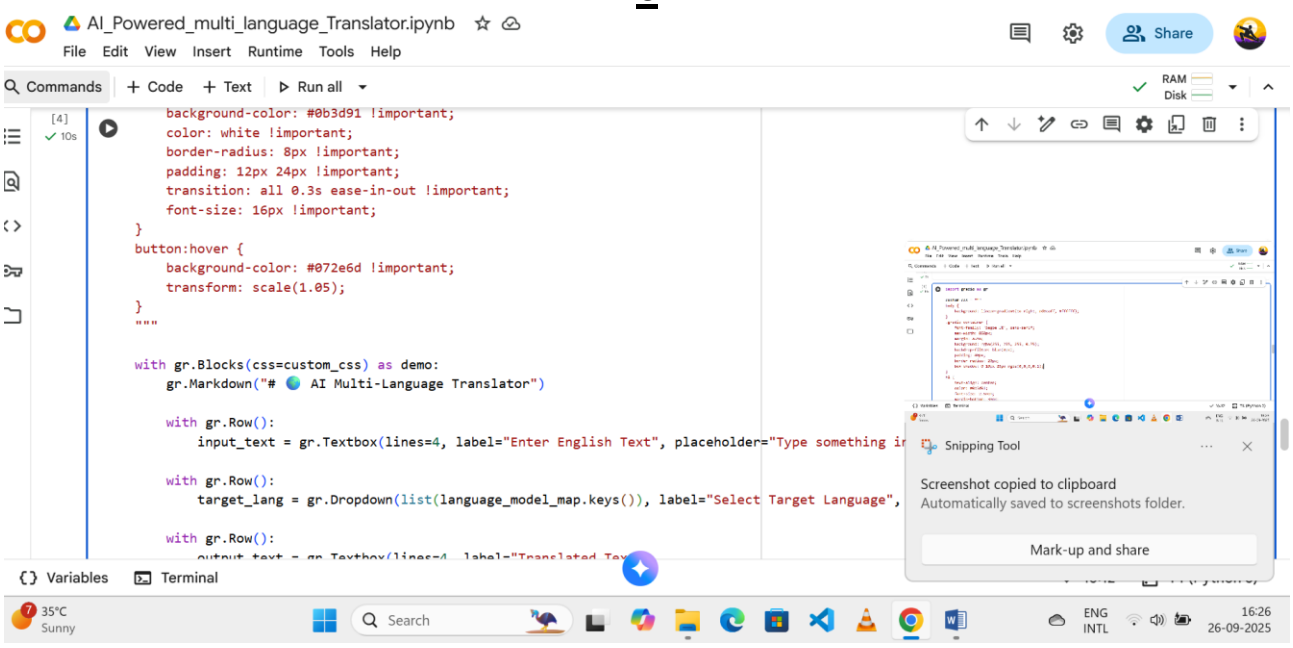
4


colab.research.google.com/drive/1j_A19w-smls4hZEM9TXFg3IsZZIQY_9x#scrollTo=Y-xuXszqJFmD
AI_Powered_multi_language_Translator.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
RAM Disk
Device set to use cuda:0
[3] 0s
def translate_text(text, target_lang):
if not text.strip():
return "⚠️ Please enter text to translate."
if target_lang not in translators:
return "❌ Selected language not supported or model not loaded."
try:
result = translators[target_lang](text)
return result[0]['translation_text']
except Exception as e:
return f"🚫 Translation failed: {str(e)}"
[4] 10s
import gradio as gr
custom_css = """
body {
background: linear-gradient(to right, #d0eaff, #ffffff);
Variables Terminal
16:12 T4 (Python 3)
35°C
Search
ENG INTL
16:26
26-09-2025

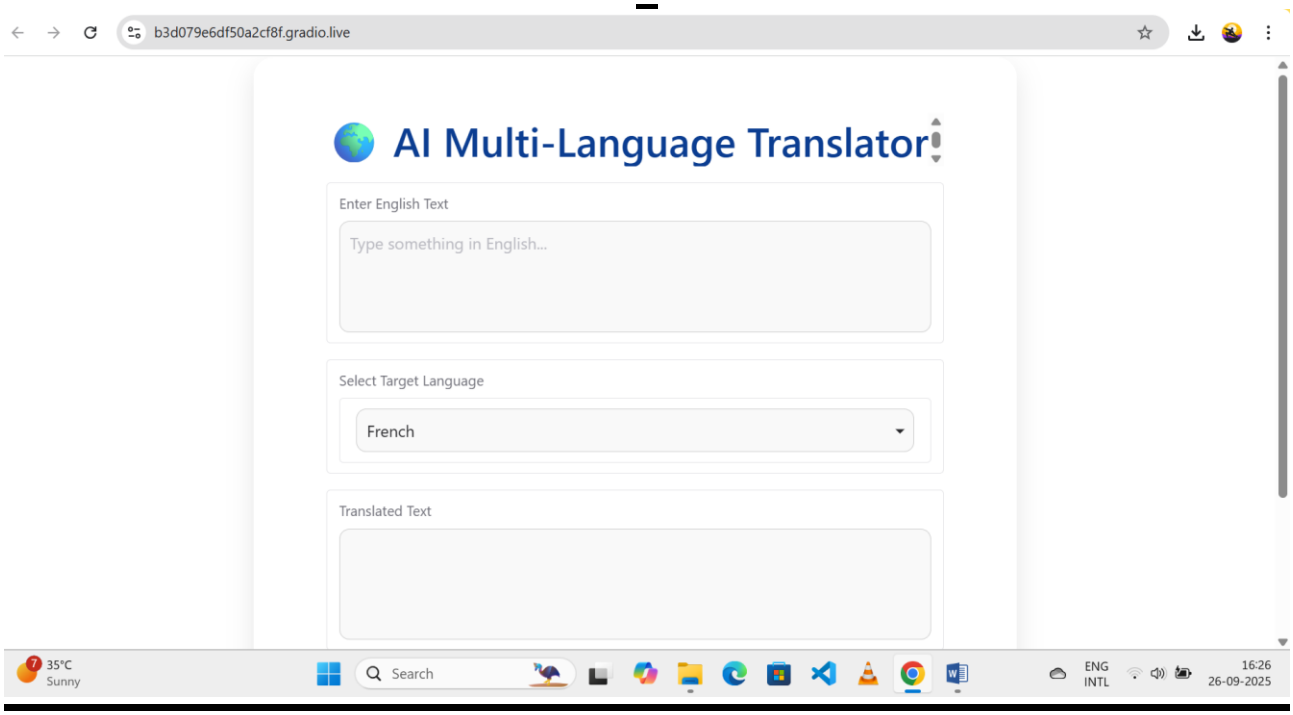
5


AI_Powered_multi_language_Translator.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
RAM Disk
[4] 10s
import gradio as gr
custom_css = """
body {
background: linear-gradient(to right, #d0eaff, #ffffff);
}
.gradio-container {
font-family: 'Segoe UI', sans-serif;
max-width: 850px;
margin: auto;
background: rgba(255, 255, 255, 0.75);
backdrop-filter: blur(8px);
padding: 40px;
border-radius: 20px;
box-shadow: 0 10px 25px rgba(0,0,0,0.1);
}
h1 {
text-align: center;
color: #0b3d91;
font-size: 2.5rem;
margin-bottom: 30px;
Variables Terminal
16:12 T4 (Python 3)
35°C Sunny
Search
ENG INTL
16:26
26-09-2025

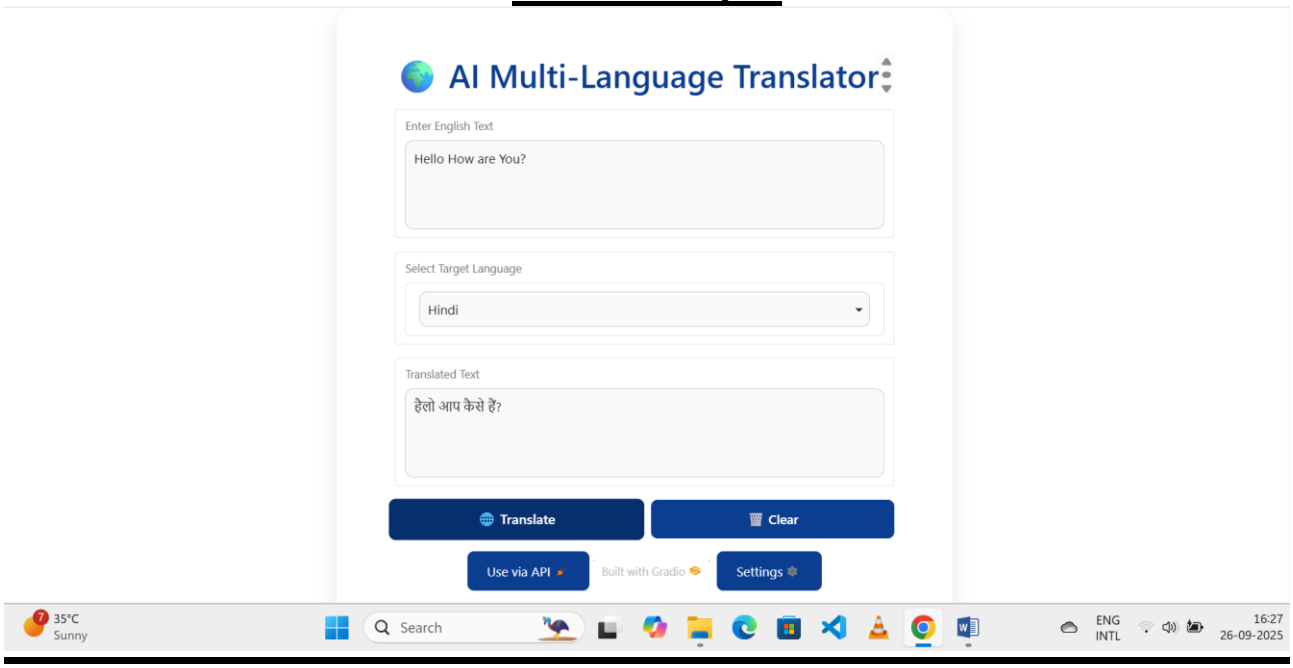
6


AI_Powered_multi_language_Translator.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
RAM
Disk
[4]
10s
background-color: #0b3d91 !important;
color: white !important;
border-radius: 8px !important;
padding: 12px 24px !important;
transition: all 0.3s ease-in-out !important;
font-size: 16px !important;
}
button:hover {
background-color: #072e6d !important;
transform: scale(1.05);
}
"""
with gr.Blocks(css=custom_css) as demo:
gr.Markdown("# AI Multi-Language Translator")
with gr.Row():
input_text = gr.Textbox(lines=4, label="Enter English Text", placeholder="Type something in
with gr.Row():
target_lang = gr.Dropdown(list(language_model_map.keys()), label="Select Target Language",
with gr.Row():
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Variables
Terminal
35°C
Sunny
Search
ENG
INTL
16:26
26-09-2025

7


b3d079e6df50a2cf8f.gradio.live
AI Multi-Language Translator
Enter English Text
Type something in English...
Select Target Language
French
Translated Text
35°C
Sunny
Search
ENG
INTL
16:26
26-09-2025

8. Final output



Challenges, Limitations, and Future Work

Challenges:

- Handling idioms, context, and cultural references.
- Low-resource languages have limited training data.
- Speech input/output requires additional modules (e.g. Whisper, TTS).

Limitations:

- Current implementation supports only text translation.
- No personalization or domain adaptation.
- Dependent on cloud (Colab) environment.

Future Enhancements:

- Add **speech-to-text** (e.g. Whisper by OpenAI).
- Add **text-to-speech** for audio output.
- Build a **web app** using Streamlit or Gradio.
- Fine-tune models for specific domains (e.g. medical, legal).
- Integrate **interactive correction** feedback from users.

Conclusion:
This project successfully implemented a functional AI-powered multi-language translator using modern NLP tools. The system supports numerous languages, demonstrates high-quality

translations, and serves as a foundation for more advanced multilingual applications. With the help of Hugging Face's models and Google Colab's compute resources, development was fast and scalable

References

- Vaswani, A., et al. (2017). *Attention is All You Need*. https://arxiv.org/abs/1706.03762

- **Hugging Face Transformers Library Documentation**
https://huggingface.co/docs/transformers

- **MarianMT (Helsinki-NLP) Model Card**
Hugging Face model hub: https://huggingface.co/Helsinki-NLP

- **MarianMT English-to-French Example Model**
https://huggingface.co/Helsinki-NLP/opus-mt-en-fr

- **mBART50 Model Card (facebook/mbart-large-50-many-to-many-mmt)**
https://huggingface.co/facebook/mbart-large-50-many-to-many-mmt

- **Tokenization with Hugging Face Tokenizers**
https://huggingface.co/docs/tokenizers

- **Google Colab - Research Environment**
https://colab.research.google.com

- **Hugging Face Datasets Library**
https://huggingface.co/docs/datasets