

IMAGE-TO-TEXT EXTRACTION USING COLPALI+ BYALDI+QWEN2_VL MODEL

Name: Niharika Amritkar

Contact: a.niharika146@gmail.com

GitHub Link: <https://github.com/NiharikaAmritkar/Image-to-Text-Qwen-VL-Colpali.git>

App URL: <https://document-app-lrf3cb0y6l.streamlit.app/>

OBJECTIVE:

To Develop and deploy a web-based prototype that demonstrates the ability to perform Optical Character Recognition (OCR) on an uploaded image (in picture format) containing text in both Hindi and English. The web application should also implement a basic keyword search functionality based on the extracted text. The prototype must be accessible via a live URL.

METHODOLOGY:

1. TASK 1:

- Conducted Research about the Model from various sources like Huggingface, Github, Youtube.
- Understood its functionalities and how it is built.
- The model is built and loaded in Kaggle Notebook as it provides good computing resources like GPU P100 and good memory storage.
- Successfully executed the implementation of 4bit and 8bit precision model of Qwen2-VL model + Colpali.
- Successfully got the output of Hindi as well as English documents that were given as an input.

2. TASK 2:

- Created a basic web application using Streamlit application.

3. TASK 3:

- Used streamlit.io for deploying the same model.
- Built and deployed using Streamlit Codespaces as it provides computing resources.

RESULTS:

- 1. Hindi text Extraction successfully done with the model (Kaggle Notebook output)**

```
print(output_text)
```

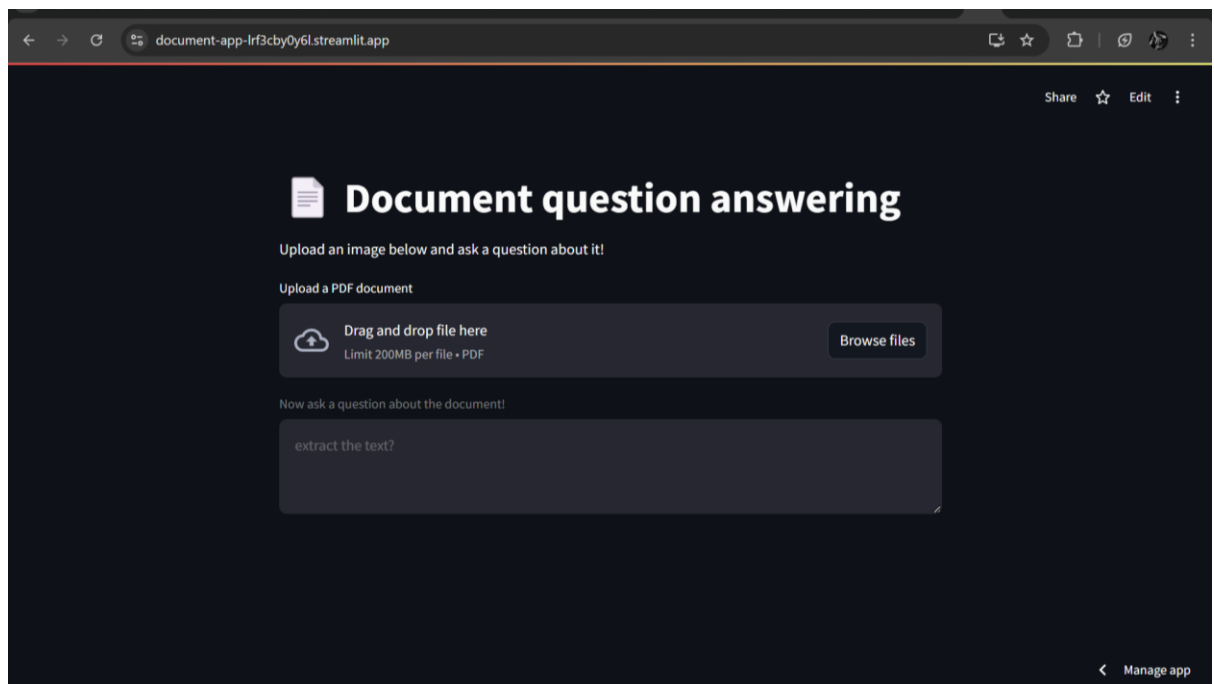
['The text in the image is in Hindi. Here is the translation:\n\n"पर खंडहर अपने-अपने खंडहर हे। राजनीकांत का मन उसको खोखली दीवारों में जाकर भले ही छिप लेत, पर उनके पैर उद्धर नहीं उठते हैं। वह अतीत है। मन में कहण-मधुर भावुकता को जगाने का अदृष्ट उपादान। उसे साकार करना गलती होगी। वर्तमान का उद्धर नाटक नष्ट हो जाएगा। बेहरो']

2. English text Extraction successfully done with the model (VS Code output)

```
print(output_text)

["Certainly! Here are the details extracted from the text you provided:\n\n---\n\n**Machine Learning U-net Model for Medical Image Segmentation**\n\n**Project's aim was to segment affected lung regions due to corona virus and extract relevant features.**\n\n**Conducted research for selection of appropriate model for the assignment.**\n\n**Created a U-Net model for segmentation of COVID-19 affected lung regions.**\n\n**Ported Kaggle data and achieved an accuracy of 56.12%.**\n\n**ADDITIONAL**\n\n**Community Service of 6 weeks for an NGO targeting the welfare of underprivileged women and girls"]
```

3. Web application



DETAILS FOR DEPLOYMENT:

- In order to run the project locally, install all the requirements for requirements.txt file provided in the GitHub link.
- Make sure you have GPU with cuda support to run the project locally on your machine.
- Run the streamlit_app.py to see the results.
- There are chances that file may crash or take huge time to load, in this case run the Jupyter notebook in the Kaggle environment. Make sure to pass the address of the documents wherever specified.

OUTPUT: the output of the project will be plain text extracted from the image/ document.

CHALLENGES:

- The model did not run and execute in Google Colab notebook due to limited resources of memory and GPU, even for the quantized models.
- Could not deploy the model on local machine due to limited resources.
- The model takes too long to run after the deployment but if resources are provided, it runs perfectly.
- The whole projects runs well on Kaggle Notebook but take too long to run as an app.

NOTE: the application takes too long to run, please consider running the Jupyter notebook. It has the exact same code which is deployed for the app. This is because of limited computing resources or some minor error which is getting generated at the time of deployment. However, the project runs smoothly and efficiently in Kaggle environment.