

MA

buatakan script streamlit ocr dengan format yang sama menjadi doc atau pdf



Untuk membuat sebuah aplikasi OCR (Optical Character Recognition) menggunakan Streamlit yang dapat mengkonversi teks yang dihasilkan menjadi dokumen PDF atau DOC, Anda bisa menggunakan pustaka Textract, Pytesseract, dan ReportLab atau python-docx. Berikut ini adalah contoh skrip untuk aplikasi Streamlit OCR yang dapat menyimpan teks yang dihasilkan dalam format PDF atau DOC.

```
python                                                                    Copy code

import streamlit as st
import pytesseract
from PIL import Image
from io import BytesIO
import base64
import cv2
import os
import tempfile
from reportlab.pdfgen import canvas
from docx import Document
from docx.shared import Pt

# Set pytesseract executable path
pytesseract.pytesseract.tesseract_cmd = 'path/to/tesseract'

def main():
    st.title("OCR - Optical Character Recognition")

    img_file = st.file_uploader("Upload an image", type="image")

    if img_file is not None:
        image = Image.open(img_file)
        st.image(image, caption="Uploaded Image", use_column_width=True)
        if st.button("Extract text"):
            ocr_result = ocr_core(image)
            st.markdown("**Extracted Text**")
            st.write(ocr_result)

            if ocr_result:
                save_format = st.selectbox("Save as:", ["PDF", "DOC"])
                if st.button(f"Save as {save_format}"):
                    if save_format == "PDF":
                        save_as_pdf(ocr_result)
                    elif save_format == "DOC":
                        save_as_doc(ocr_result)

def ocr_core(image):
    gray_img = cv2.cvtColor(np.array(image), cv2.COLOR_BGR2GRAY)
    text = pytesseract.image_to_string(gray_img, lang="eng")
    return text

def save_as_pdf(text):
    output = BytesIO()
    c = canvas.Canvas(output)
    c.setFont("Helvetica", 12)
    c.drawString(50, 800, text)
    c.showPage()
    c.save()
    output.seek(0)
    b64 = base64.b64encode(output.getvalue()).decode()
    href = f'<a href="data:application/pdf;base64,{b64}">Download PDF</a>'
    st.markdown(href, unsafe_allow_html=True)

def save_as_doc(text):
    doc = Document()
    paragraph = doc.add_paragraph()
    run = paragraph.add_run(text)
    run.font.size = Pt(12)
    with tempfile.NamedTemporaryFile(delete=False, suffix=".docx"):
        doc.save(fp.name)
        st.download_button("Download DOC", fp.name, "extracted_text.docx")
        os.unlink(fp.name)

if __name__ == "__main__":
    main()
```

Jangan lupa untuk mengganti `'path/to/tesseract'` dengan jalur yang sesuai ke eksekutabel Tesseract di sistem Anda. Skrip di atas akan membuat aplikasi Streamlit yang memungkinkan Anda mengunggah gambar, mengekstrak teks menggunakan OCR, dan menyimpan teks yang dihasilkan dalam format PDF atau DOC.

👍 🗨 📄 🔊

MA

buatakan requirment.txt nya juga



Untuk membuat aplikasi OCR dengan Streamlit, Anda perlu menginstal beberapa pustaka Python yang diperlukan. Buat file bernama `requirements.txt` dengan konten berikut:

```
makefile                                                                    Copy code

streamlit==1.3.0
pillow==9.0.1
opencv-python-headless==4.7.4
pytesseract==0.3.8
reportlab==3.6.3
python-docx==0.8.12
```

Simpan file `requirements.txt` ini di folder yang sama dengan skrip aplikasi Streamlit Anda. Kemudian, Anda dapat menginstal semua dependensi yang diperlukan dengan menjalankan perintah berikut di terminal atau Command Prompt:

```
bash                                                                    Copy code

pip install -r requirements.txt
```

Setelah menginstal semua dependensi, Anda bisa menjalankan aplikasi Streamlit dengan perintah berikut:

```
bash                                                                    Copy code

streamlit run your_script_name.py
```

Gantilah `your_script_name.py` dengan nama file skrip aplikasi Streamlit Anda.

👍 🗨 📄 🔊

MA

bagaimana langkah langkahnya # Set pytesseract executable path
pytesseract.pytesseract.tesseract_cmd = 'path/to/tesseract'



Untuk menggunakan pytesseract di aplikasi Anda, Anda harus mengatur jalur (path) ke eksekutabel Tesseract. Berikut adalah langkah-langkah untuk mengaturnya:

1. Instalasi Tesseract OCR:

Pertama, Anda perlu menginstal Tesseract OCR di sistem Anda. Bergantung pada sistem operasi yang Anda gunakan, langkah-langkahnya mungkin berbeda:

- Windows:
 - Unduh Tesseract OCR dari tautan berikut: <https://github.com/UB-Mannheim/tesseract/wiki>
 - Jalankan file instalasi dan ikuti langkah-langkahnya.
 - Catat lokasi instalasi Tesseract (misalnya, `C:\Program Files\Tesseract-OCR`).
- Ubuntu / Debian:

```
csharp                                                                    Copy code

sudo apt-get install tesseract-ocr -y
sudo apt-get install libtesseract-dev -y
```

- macOS:

```
bash                                                                    Copy code

brew install tesseract
```

2. Temukan eksekutabel Tesseract:

- Windows: Eksekutabel biasanya terletak di folder instalasi Tesseract, seperti `C:\Program Files\Tesseract-OCR\tesseract.exe`.
- Ubuntu / Debian / macOS: Anda dapat menemukan jalur ke eksekutabel Tesseract dengan menjalankan perintah berikut di terminal:

```
bash                                                                    Copy code

which tesseract
```

Ini akan menampilkan jalur ke eksekutabel, seperti `/usr/local/bin/tesseract`.

3. Atur jalur eksekutabel Tesseract di skrip Python Anda:

Gantilah `'path/to/tesseract'` dengan jalur yang sesuai ke eksekutabel Tesseract di sistem Anda, seperti yang Anda temukan pada langkah sebelumnya. Contoh:

```
python                                                                    Copy code

pytesseract.pytesseract.tesseract_cmd = r'C:\Program Files\Tesseract-OCR\tesseract.exe'
```

atau

```
python                                                                    Copy code

pytesseract.pytesseract.tesseract_cmd = r'/usr/local/bin/tesseract'
```

Sekarang Anda telah mengatur jalur eksekutabel Tesseract, aplikasi Anda akan dapat menggunakan pytesseract untuk melakukan OCR pada gambar yang diunggah.

👍 🗨 📄 🔊

MA

cara mengatasi ini NameError: name 'np' is not defined

Untuk mengatasi kesalahan "NameError: name 'np' is not defined", Anda perlu mengimpor pustaka NumPy dengan alias 'np'. Kesalahan ini muncul karena Anda menggunakan 'np' di kode Anda, tetapi belum mengimpor pustaka NumPy. Tambahkan baris berikut di awal skrip Python Anda untuk mengimpor NumPy:

```
python                                                                    Copy code

import numpy as np
```

Setelah menambahkan baris ini, kesalahan "NameError: name 'np' is not defined" harus hilang, dan Anda dapat melanjutkan dengan menggunakan pustaka NumPy dalam kode Anda.

👍 🗨 📄 🔊

MA

cara mengatasi ini NameError: name 'np' is not defined