Extracting DOIs from a PDF (requires manual check still because of line breaks)

```
import pdfplumber
import re
import csv
import os
from datetime import datetime

def extract_dois_from_pdf(pdf_path):
    """
    Extracts DOI numbers from the given PDF file and returns a list of DOIs.
    Removes the final period if it's part of the DOI.
    Fixes DOIs split across line breaks.
    """
    dois = []
    with pdfplumber.open(pdf_path) as pdf:
        for page in pdf.pages:
            # Extract text from the page
            text = page.extract_text()
            if text:
                # Step 1: Join the lines into a single block of text
                text = ' '.join(text.splitlines())

                # Step 2: Fix broken DOIs that split after a period (.) or a slash (/)
                # We look for a line break after a period or slash and then merge it with the next part.
                text = re.sub(r'([\.\/])\s*(?=\d)', r'\1', text)  # Merge split DOI after period or slash

                # Step 3: Use regex to find DOI numbers
                doi_pattern = r'\b10\.\d{4,9}/[-._;()/:A-Z0-9]+(?=\b|\s|$)'
                found_dois = re.findall(doi_pattern, text, re.IGNORECASE)

                # Step 4: Remove the final period if it's part of the DOI
                cleaned_dois = [doi.rstrip('.') for doi in found_dois]
                dois.extend(cleaned_dois)

    return dois
```

```
def save_dois_to_csv(dois, pdf_path):
    """
    Save the extracted DOIs to a CSV file with consecutive numbering.
    The output file name contains the input PDF name and a timestamp.
    """
    # Get the base name of the input PDF file (without extension)
    base_name = os.path.splitext(os.path.basename(pdf_path))[0]

    # Get the current timestamp in the format YYYY-MM-DD_HH-MM-SS
    timestamp = datetime.now().strftime('%Y-%m-%d_%H-%M-%S')

    # Generate the output CSV file name
    output_csv_path = f"{base_name}_{timestamp}_dois.csv"

    with open(output_csv_path, mode='w', newline='', encoding='utf-8') as file:
        writer = csv.writer(file)
        # Write DOIs with consecutive numbering in the first column
        for idx, doi in enumerate(dois, start=1):
            # Prefix "https://doi.org/" to each DOI
            doi_url = f"https://doi.org/{doi}"
            writer.writerow([idx, doi_url])  # Write the index and the prefixed DOI

    print(f"DOIs have been saved to {output_csv_path}")
    return output_csv_path

def main():
    pdf_path = '0446.1.00.pdf'  # Path to the input PDF file

    # Extract DOIs from PDF
    dois = extract_dois_from_pdf(pdf_path)

    if dois:
        print(f"Found {len(dois)} DOIs. Saving to CSV...")
        # Save the extracted DOIs to a CSV file
        save_dois_to_csv(dois, pdf_path)
    else:
        print("No DOIs found in the PDF.")

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

Revision #2
Created 2 February 2025 08:48:41 by Vincent
Updated 4 February 2025 10:48:19 by Vincent