OCR Workshop

Indholdsfortegnelse

Introduktion:

I denne workshop skal vi arbejde med OCR scanning. OCR står for Optical Character Recognition og er den teknik der gør pdf-filer søgbare og gør så man kan markere tekst i filerne. Med OCR-scanning af PDF-filer kan vi altså få vores computer til at ændre syn på pdf-filer. Computeren har i første omgang set pdf-filerne som billeder af tekst og når vi er færdige med workshoppen vil vi kunne få computeren til at læse filerne som tekst. Når vi er færdige, kan vi i vores scannede filer søge efter nøgleord, markere tekst nemt og kopiere ud til citater. OCR-scanning har dog nogle begrænsninger, som vi skal være opmærksomme på. OCR-scanningen bliver aldrig bedre end den fysiske scanning af teksten. Hvis man ikke har en god tekst til at starte med vil man heller ikke få et særligt godt OCR-resultat.

Installation:

For at kunne lave OCR scanning af vores filer skal vi bruge et Command Line Program der hedder OCRmyPDF. Først og fremmest: Command Line Interface (CLI) er når man interagere med sin computer i PowerShell/Terminal. OCRmyPDF er det program vi skal installere. Vi skal gøre forskellige ting, alt efter om man sidder ved en Windows-maskine eller en Mac-maskine.

Windows:

Åben PowerShell som administrator og kør efterfølgende linjer af kode ved at kopiere et punkt af gangen, trykke enter, skrive ”a” og trykke enter igen. Gør det samme for alle seks punkter i den rækkefølge de har her:

1. `Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))`
2. `choco install python3`
3. `choco install --pre tesseract`
4. `choco install ghostscript`
5. `choco install pngquant`
6. `pip install ocrmypdf`

Når dette er gjort, skal vi have lært computeren nogle forskellige sprog. På dette link finder i nogle udvalgte sprog filer, som skal hentes, hvis i har hentet filerne fra mailen inden workshoppen har i dem allerede liggende på computeren: https://github.com/VictorHarbo/OCR_workshop

Mac:

Følg dette link: https://docs.conda.io/en/latest/miniconda.html og installer .pkg-filen. Når denne fil er installeret, åbnes terminal og følgende linjer skrives en ad gangen efterfulgt af et enter:

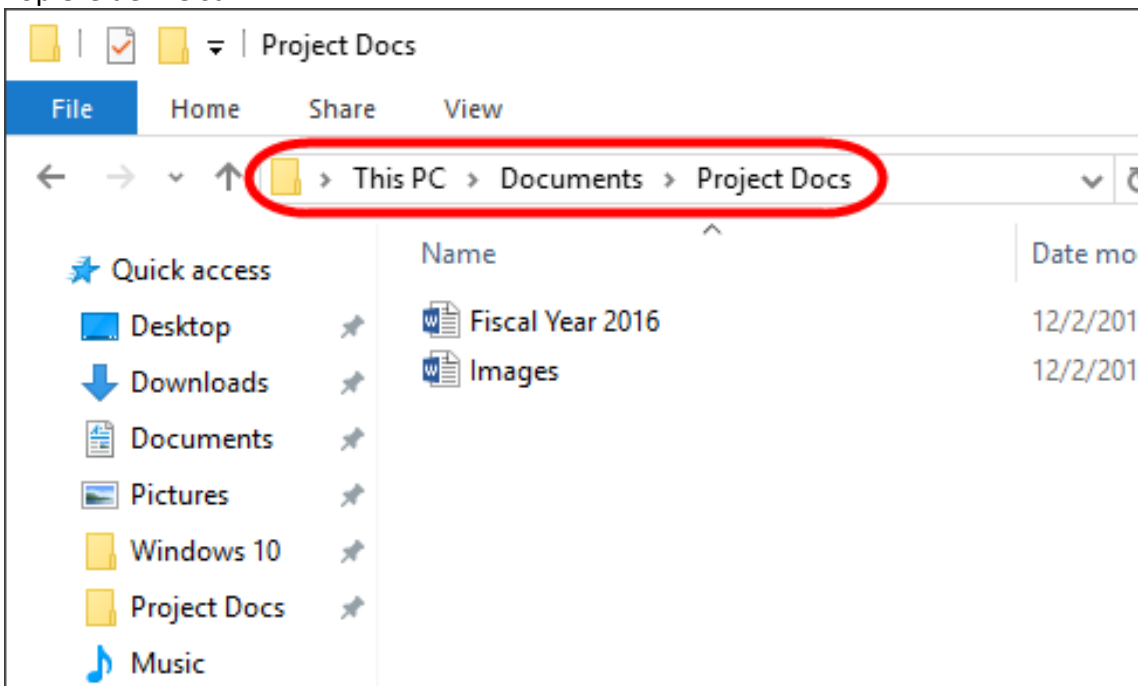
1. `conda install -c conda-forge ghostscript`
2. `conda install -c conda-forge tesseract`
3. `pip install ocrmypdf`

Find rundt i din computer igennem Command Line Interface (CLI)

Som nævnt under installationen er OCRmyPDF et command line program, det betyder at vi skal bruge vores computer på en anden måde end hvad vi er vandt til. Vi får nemlig ikke lov til at arbejde særligt meget med et skrivebord eller en flot visuel mappestruktur. I stedet for skal vi navigere igennem en tekstbaseret jungle i henholdsvis PowerShell på windows og terminal på mac. For at kunne navigere rundt er der en command vi skal kende. Den hedder cd. Denne cd-command har intet med de gamle cd'er på dine hylder at gøre, den står derimod for "change directory". Det er altså vores command til at bevæge os rundt i vores computer.

Nemmeste navigering:

Der er mange forskellige måder at bruge cd-commanden på. Det nemmeste, hvis man ikke er vandt til at navigere igennem command line er, at finde den fulde sti til den placering på sin computer, som man forsøger at finde på command linen. Det lyder lidt indviklet, så her er en bedre forklaring. På Windows kan man i sin stifinder trykke på den gule mappe i det nedenfor markerede område og kopiere denne sti:



På Mac har man to muligheder: Enten trækker man den mappe man vil have stien til ind i sin terminal, eller også markerer man den mappe man vil have stien til og trykker på: option + command + c på samme tid og så kan man sætte stien ind med command + v genvejen.

Uddybning af navigerings muligheder:

Som tidligere nævnt, så er der mange forskellige måder at navigere rundt i sin computer med cd-commanden, nedenfor har jeg forsøgt at samle nogle af de mest brugbare metoder at finde rundt i sin computer:

Command:	**Forklaring**:

cd	Command der skifter mappe, den skal efterfølges af hvor man gerne vil hen i sin computer.
cd ./mappe	Skifter til mappen "mappe" under den mappe, som vi befinder os i nu.
cd ..	Skifter til mappen ovenover den vi er i nu.
cd ~/	Skifter til computerens hjemmeplacering, som kan være et godt udgangspunkt.
Cd c:/users/name/billeder	Skifter til mappen billeder under name, under users på c-drevet.
dir	viser os indholdet af den nuværende mappe
ls	viser os indholdet af den nuværende mappe

Brug af OCRmyPDF:

Grundlæggende brug:

Vi er nu klar til at gå i gang med at arbejde med OCRmyPDF. For at vi kan bruge programmet skal vi i PowerShell/terminal navigere hen til den fil vi gerne vil OCR-scanne. Dette gøres med cd-commanden:
Windows eksempel: cd C:\Users\user\Desktop\OCR_Workshop\
Mac eksempel: cd ~/Desktop/OCR_Workshop

Når vi har navigeret hen til vores fil, skal vi ”kalde” på scriptet og den fil vi vil køre programmet på og hvilken fil resultatet skal gemmes i. Dette gøres ved at skrive følgende i PowerShell/terminal:

- `ocrmypdf filnavn.pdf nyt_filnavn.pdf`

Dette er den helt grundlæggende funktionalitet i programmet. Man kan lave rigtigt mange smartere ting med det også. Fx kører den ovenfornævnte command filen igennem engelsk OCR. For at ændre dette og alle andre ting skal man modificere den linje kode man kører med noget der hedder et -flag. -flags er stykker af kode man sætter efter ocrmypdf for at fortælle programmet hvad det skal gøre. Fx er flaget for sprog -l for ”language”. Et godt sted at starte er med -h flaget. Det står for ”help” og giver os en masse information om programmet og dets muligheder.

Flere funktioner:

Nedenfor følger en forklaring af nogle af de mest brugbare flags:

Help flag: -h	Dette flag giver en lang menu og forklaring af de forskellige muligheder der er i programmet.
Language flag: -l	Dette flag gør det muligt at anvende andre sprog end engelsk. Man kan også anvende flere sprog på en gang. Flaget bruges sådan her for dansk sprog: `-l dan` Hvis man ønsker flere sprog på en gang bruges det sådan her: `-l eng+dan` for engelsk og dansk.
Force OCR flag: -f	Dette flag bruges, hvis filen allerede indeholder et OCR lag, men det er af dårlig kvalitet og man vil have lavet et nyt OCR-lag.

Skip text flag: -s	Dette flag bruges, hvis der er dele af filen der allerede har OCR og man vil gemme den del, men lave OCR-scanning på resten.
Deskew flag: -d	Dette flag forsøger at rette op på skævt scannede filer, så teksten står lige efterfølgende.
Rotate pages flag: -r	Dette flag forsøger at rotere siderne, hvis programmet registrerer, at teksten vender vandret i stedet for lodret. (Min erfaring er, at den ikke er så god til at registrere det i danske filer, der er dobbeltsidet). Kan efterfølges af --rotate-pages-threshold 0.01 hvis man er sikker på at alle sider skal vendes og programmet ikke opfanger det.
Sidecar flag: --sidecar [FILE]	Dette flag trækker OCR-laget ud i en .txt fil udover at lave OCR på .pdf-filen. Den kan være god at bruge, hvis man er interesseret i hvor god kvalitet OCR-laget har fået.
Remove Background flag: --remove-background	Dette flag forsøger at fjerne baggrundsfarven fra filerne. Det vil sige, at hvis PDF-filen er på gulligt papir, vil flaget forsøge at gøre filen hvid. Der er ingen mellemrum i flaget.