

Belege auslesen ohne Abtippen

Wie du Rechnungen, Belege und Scans mit dem kostenlosen Werkzeug DocStrange in saubere Daten verwandelst. Im Browser, als Stapel über die Cloud oder komplett lokal auf deinem eigenen Rechner.

Der schnelle Weg: im Browser

Für einzelne Belege oder zum Ausprobieren. Kein Programm installieren, nur ein kostenloses Konto.

1. Öffne **docstrange.nanonets.com** im Browser und klick auf „Start Free-Trial“.
2. Registriere dich oder melde dich mit einem Google-Konto an.
3. Zieh deinen Beleg in das Upload-Feld, ein PDF oder ein Foto, oder wähl die Datei vom Rechner.
4. Wähl als Ausgabeformat **CSV**, wenn das Ergebnis in Excel soll. Für reinen Text **Markdown**, für einzelne Felder **JSON**.
5. Klick auf **RUN**. Je nach Größe steht das Ergebnis nach ein paar Sekunden da.
6. Lad die Datei herunter. Eine CSV öffnest du per Doppelklick in Excel, dort steht alles sauber in Spalten.

Deine Belege laufen in diesem Modus über die Server von Nanonets. Für unkritische Belege vertretbar, für sensible Unterlagen nicht. Mehr dazu im Abschnitt Datenschutz.

Viele Belege auf einmal: der Stapel-Weg (über Cloud)

Wenn regelmäßig Stapel anfallen, richtest du das einmal als kleines Programm ein. Danach legst du beliebig viele Belege in einen Ordner und bekommst eine fertige Tabelle. Ehrlich vorweg: Die Einrichtung lädt im Hintergrund mehrere Gigabyte herunter, plan eine halbe Stunde und etwas Speicherplatz ein. Auch hier laufen die Belege über die Cloud von Nanonets. Den komplett lokalen Weg findest du im nächsten Abschnitt.

1. **Python installieren.** Python ist die Grundlage, auf der das Skript läuft, kostenlos. Geh auf **python.org**, lad die aktuelle Version und öffne die Datei. Unter Windows ein wichtiger Schritt, den viele übersehen: Setz ganz unten das Häkchen bei „Add Python to PATH“, bevor du auf „Install Now“ klickst. Fehlt das Häkchen, kennt dein Rechner den Befehl später nicht, und nichts funktioniert. Falls du es vergisst, installier Python einfach noch einmal und hol das Häkchen nach.
2. **Terminal öffnen.** Das Terminal ist das Fenster für Befehle. Windows: Windows-Taste drücken, **cmd** tippen, Eingabeaufforderung öffnen. Mac: Cmd und Leertaste, **Terminal** tippen, Enter.
3. **DocStrange installieren und anmelden.** Tipp nacheinander:

```
pip install docstrange
docstrange login
```

Der erste Befehl lädt mehrere Gigabyte, das dauert. Kommt die Meldung, dass pip nicht gefunden wird, hat das PATH-Häkchen gefehlt. Der zweite Befehl öffnet den Browser für eine kostenlose Anmeldung. Erst dadurch hast du 10.000 Belege im Monat frei. Ohne Anmeldung nur ein paar pro Tag.

- 4. Skript anlegen.** Öffne einen Texteditor (Windows: Editor, Mac: TextEdit im reinen Textmodus). Kopier den folgenden Text vollständig hinein. Er enthält ein fertiges Schema für deutsche Rechnungen, also die Liste der Felder, die ausgelesen werden sollen.

```
from docstrange import DocumentExtractor
import csv, glob, os, json

extractor = DocumentExtractor() # Cloud-Modus, nutzt deinen Login

# Diese Felder werden ausgelesen. Zeilen ergaenzen oder loeschen nach Bedarf.
schema = {
    "rechnungsnummer": "string",
    "rechnungsdatum": "string",
    "lieferant_name": "string",
    "lieferant_ustid": "string",
    "netto_betrag": "number",
    "ust_satz": "number",
    "ust_betrag": "number",
    "brutto_betrag": "number",
    "faelligkeit": "string",
    "iban": "string",
    "leistung": "string",
}

muster = ["belege/*.pdf", "belege/*.jpg", "belege/*.png"]
dateien = [p for m in muster for p in glob.glob(m)]

zeilen = []
for pfad in dateien:
    antwort = extractor.extract(pfad).extract_data(json_schema=schema)
    felder = antwort.get("structured_data", antwort)
    felder["quelldatei"] = os.path.basename(pfad)
    zeilen.append(felder)
    if len(zeilen) == 1:
        print("Kontrolle:", json.dumps(felder, ensure_ascii=False, indent=2))

spalten = list(schema.keys()) + ["quelldatei"]
with open("rechnungen.csv", "w", newline="", encoding="utf-8-sig") as f:
    w = csv.DictWriter(f, fieldnames=spalten, extrasaction="ignore")
    w.writeheader()
    w.writerows(zeilen)

print(len(zeilen), "Belege ausgelesen -> rechnungen.csv")
```

Die Liste im oberen Teil, das **schema**, ist der Teil den du anpasst. Links steht der Feldname, rechts der Typ: **string** für Text, **number** für eine Zahl. Brauchst du zusätzlich die Bestellnummer oder die Kostenstelle, ergänz eine Zeile, etwa "bestellnummer": "string", . Felder die du nicht brauchst, löschst du raus.

Speicher die Datei auf dem Desktop als **rechnungen.py**. Am Ende muss **.py** stehen, nicht **.txt**.

- 5. Belege auslesen.** Leg auf dem Desktop einen Ordner **belege** an und kopier deine Rechnungen hinein, als PDF oder Foto. Im Terminal: **cd Desktop** und Enter, dann **python rechnungen.py** und Enter. Das Skript zeigt zur Kontrolle den ersten Beleg an und schreibt am Ende **rechnungen.csv**, die du in Excel öffnest. Jede Zeile ein Beleg, jede Spalte ein Feld, dazu eine Spalte mit dem Dateinamen zur Zuordnung zum Original.

Komplett lokal: nichts verlässt deinen Rechner

Für sensible Belege mit Personen- oder Mandantendaten. Die Verarbeitung läuft vollständig auf deinem eigenen Rechner, kein Beleg geht an fremde Server. Voraussetzung ist eine NVIDIA-Grafikkarte mit

CUDA-Unterstützung, also die leistungsstarke Sorte aus Gaming- oder KI-Rechnern. Ein gewöhnlicher Büro-Laptop hat das nicht.

Lokale Web-Oberfläche (GUI)

DocStrange bringt eine eigene Oberfläche mit, die lokal im Browser läuft. Drag-and-Drop wie die Online-Version, aber alles bleibt auf deinem Rechner.

1. Installiere die Web-Variante:

```
pip install "docstrange[web]"
```

2. Starte die Oberfläche:

```
docstrange web
```

3. Öffne **http://localhost:8000** im Browser. Beim ersten Start lädt das Programm die nötigen Modelle herunter, das dauert einmalig. Danach ziehst du Belege per Drag-and-Drop hinein, wählst das Format und bekommst das Ergebnis. Wähl in den Einstellungen den GPU-Modus, damit die Verarbeitung lokal bleibt.

Lokaler Stapel-Weg (Skript)

Für den Stapel-Weg änderst du im Skript eine einzige Zeile. Statt `DocumentExtractor()` schreibst du:

```
extractor = DocumentExtractor(gpu=True) # Lokal, braucht NVIDIA/CUDA
```

Der Rest des Skripts bleibt identisch. Die Verarbeitung dauert beim ersten Durchlauf länger, weil die Modelle heruntergeladen werden. Danach geht es schnell, und kein einziger Beleg verlässt deinen Rechner.

Ohne NVIDIA-Karte kann DocStrange auch auf dem Hauptprozessor (CPU) laufen, ist dann aber deutlich langsamer. Für gelegentliche Belege brauchbar, für größere Stapel nicht praktikabel.

Wo deine Belege landen

Im Überblick:

Weg	Belege gehen an	Brauchst du
Browser / Stapel Cloud	Nanonets-Server	Nur Internet
Lokale GUI / Lokaler Stapel	Nirgendwohin	NVIDIA-Grafikkarte

Für unkritische Belege ist der Cloud-Weg vertretbar. Bei sensiblen Unterlagen mit Personen- oder Mandantendaten solltest du den lokalen Weg nehmen. Die Entscheidung hängt an deiner Hardware: Hast du eine NVIDIA-Karte, richte einmal den lokalen Weg ein. Hast du keine, schick keine vertraulichen Belege über die Cloud.

Was DocStrange nicht kann

Kopiergeschützte PDFs verarbeitet es nicht ohne Weiteres, die musst du vorher entsperren oder als Bild speichern. Handschrift erkennt es nur eingeschränkt. Die ausgelesene Tabelle ist eine Vorlage, kein Ersatz für die Kontrolle: Den letzten Abgleich vor der Buchung macht ein Mensch.

Bessere Ergebnisse: woran die Erkennung hängt

Fotografier gerade von oben statt schräg, mit gleichmäßigem Licht ohne harte Schatten oder Spiegelungen. Ein Scan ist einem Foto fast immer überlegen. Ein Beleg pro Datei ist sauberer als mehrere auf einem Blatt. Am genauesten liest das Werkzeug digitale PDFs, also direkt am Rechner erzeugte Rechnungen, weil der Text dort schon als Text vorliegt und nicht erst aus einem Bild erkannt werden muss.

Kontakt

David Paci · [webse.at](mailto:info@webse.at)

Weiherweg 4, 9500 Villach

info@webse.at · +43 650 4352 452

www.webse.at

