Research document проекта

Система автоматического анализа электронного документа (PDF)

1.Введение

1.1 Актуальность проблемы

Электронные документы в формате PDF широко используются в различных сферах деятельности, таких как бизнес, образование и юриспруденция. Однако, извлечение информации из PDF-документов часто связано с определенными трудностями, такими как структурные ограничения формата и наличие скрытого текста. Эти препятствия делают автоматический анализ PDF важной задачей для повышения эффективности обработки цифровой документации.

1.2 Задачи исследования

1. Исследование методов обработки и извлечения текста, изображений и таблиц из PDF, включая случаи, когда они скрыты от копирования.
2. Определение природы и структуры формата PDF, выявление его особенностей и ограничений.
3. Анализ существующих решений и инструментов для работы с PDF.

2. Обзор литературы и технологий

1. Статья «Распознавание текста с помощью OCR» посвящена использованию OCR (оптического распознавания символов) для распознавания текста на изображениях, с акцентом на движок с открытым исходным кодом Tesseract. Tesseract OCR: Это самая популярная библиотека для OCR, которая использует нейронные сети для распознавания текста. Она применяет адаптивное распознавание — двухэтапный процесс, который повышает точность результатов. Необходимы обученные языковые модели для каждого языка. Фильтрация и обработка изображений с помощью библиотек, таких как OpenCV, могут значительно улучшить качество распознавания. Однако Tesseract имеет ограничения в точности (до 70% при идеальных условиях), особенно при плохом освещении или низком качестве изображений.
   https://habr.com/ru/articles/471542/
2. В статье « PDF с точки зрения программиста» обсуждаются особенности и ограничения формата PDF с позиции программиста, работающего с его чтением и записью. PDF был разработан Adobe в конце 1980-х как формат для электронного отображения документов, сохраняющий их оригинальный

вид на различных устройствах и платформах, но не предназначенный для редактирования. Несмотря на эволюцию и добавление разнообразного контента, базовая цель осталась прежней. Формат сохраняет вид, размещая символы, графику и изображения с использованием векторных и растровых команд. Векторные PDF формируются через печать на PDF-принтер, а растровые PDF создаются через сканирование, при этом каждая страница представляется как изображение. В отличие от текстовых форматов (DOC, RTF, DOCX), создаваемых для редактирования, PDF часто генерируется из других форматов через виртуальные принтеры, превращая визуальный вид документа в набор графических команд, что затрудняет изменение и манипуляцию содержимым.
https://habr.com/ru/companies/contentai/articles/108459/

3. В статье «Работа с PDF-файлами в Python» представлены основные инструменты и библиотеки, которые предоставляют разнообразные возможности для парсинга и текстового анализа PDF-документов. PyPDF2: Для извлечения информации, разделения и объединения документов, обрезки страниц и добавления водяных знаков. Поддерживает зашифрованные документы. PDFMiner.six: Подходит для анализа и преобразования PDF-документов, поддерживает PDF 1.7 и языки CJK. PDFQuery: Упрощает извлечение данных из PDF, используя PDFMiner, lxml и pyquery.
https://waksoft.susu.ru/2020/02/17/rabota-s-pdf-fajlami-v-python-chast-i-chtenie-i-razbor/