doi: 10.51639/2713-0576_2026_6_2_67
Научная статья
УДК 32.019.5:355.4
ГРНТИ 11.15.89
ВАК 5.9.9

Создание PDF-анализатора с расширенными функциями

Владислав Валерьевич Душкин[1], Виктория Александровна Абрамова [2]*
Краснодарское высшее военное училище имени генерала армии С.М. Штеменко, Краснодар, Россия
[1]dushkin@list.ru, [2]*vabramova474@gmail.com

Аннотация

В статье рассматривается процесс проектирования и разработки программного средства для комплексного анализа PDF-документов с использованием библиотек PyPDF2, pdfplumber и Apache Tika, а также интеграция машинного обучения для распознавания структуры документов. Предложена архитектура системы с расширенными функциями: извлечение текста, таблиц, изображений, семантический анализ и обработка больших объемов данных, описана архитектура разработанного приложения на языке Python, использующего библиотеки PyPDF2, pdfplumber, pandas и matplotlib. Представлены алгоритмы работы модулей извлечения текста, анализа частотности слов, построения облаков слов и тепловых карт, а также экспорта данных в формате Excel. Приводятся результаты тестирования системы на коллекции из научных статей и технических документов. Сделан вывод о практической значимости разработанного инструмента для исследователей, аналитиков и специалистов по работе с документацией.

Ключевые слова: PDF-парсер, машинное обучение, извлечение данных, анализ текста, визуализация данных, Python, статистическая обработка, интеллектуальный анализ документов, информационная безопасность.

Актуальность работы обусловлена повсеместным использованием формата PDF для хранения научной, технической и деловой документации при одновременном отсутствии универсальных инструментов для автоматизированного извлечения и глубокого анализа данных из таких файлов.

Цель исследования - описать создание анализатора с расширенными функциями для задач информационной безопасности и научных исследований. Формат Portable Document Format (PDF) остается одним из наиболее распространенных способов распространения электронных документов, предназначенных для печати и визуального просмотра благодаря своей универсальности и защите от изменений. Сложность автоматической обработки PDF-файлов заключается в том, что они хранят информацию в виде графических объектов, а не структурированного текста с четкой разметкой [1 - 2]. Как отмечается в ряде исследований, извлечение данных из PDF-документов вручную является трудоемким процессом, что особенно критично при работе с большими массивами технической документации или научных статей.

Архитектура PDF-анализаторов (таблица 1)

Современные анализаторы строятся на модульной архитектуре:

- Модуль извлечения: PyPDF2 для базового парсинга, pdfplumber для таблиц и визуализации [3 - 7];
- Модуль OCR: Tesseract для растровых элементов [8];
- Модуль ML: LayoutParser или DeepPDF для распознавания макета [9].

Таблица 1 - Архитектура PDF-анализаторов

№ п/п	Компонент	Библиотека	Функции
1.	Текст	PyPDF2	Извлечение, метаданные [3] |
2.	Таблицы	pdfplumber	Координаты, экспорт CSV [4]
3.	Изображения	PDFBox	Векторизация [10]

Технологии извлечения данных

Извлечение текста из PDF использует OCR и нейросети: LayoutLM для понимания layout. Apache Tika обеспечивает кросс-платформенность и обработку 1000+ форматов. Для больших данных применяют Docker-based пайплайны как pd3f [9 - 13].

Проблема автоматизации извлечения данных из PDF-документов активно исследуется в научном сообществе. В работе М.Д. Кулакова представлено десктопное приложение на Python для распознавания текста и таблиц из PDF-файлов и изображений с использованием Tesseract OCR и библиотек PDFMiner, PDFPlumber, PyPDF2. Данное решение ориентировано на оцифровку документов и экспорт в формат DOCX.

Более масштабный подход предлагается в системе AdaParse (Adaptive Parallel PDF Parsing and Resource Scaling Engine), которая представляет собой стратегию назначения оптимального парсера для каждого документа на основе данных. Система использует оптимизацию на основе человеческих предпочтений (Direct Preference Optimization) для согласования процесса выбора с экспертной оценкой и демонстрирует 17-кратное увеличение пропускной способности при сохранении точности на уровне современных решений [14 - 16].

В контексте анализа табличных данных особый интерес представляет работа, посвященная автоматизации извлечения характеристик электронных компонентов из PDF-документов. Авторы подчеркивают, что отсутствие эффективных инструментов считывания информации затрудняет использование технических данных конструкторами современных предприятий.

Практические реализации также включают создание Telegram-ботов для обработки гарантийных писем с извлечением реквизитов (ИНН, наименование юрлица, даты) и экспортом в Excel. Это демонстрирует востребованность подобных инструментов в бизнес-среде.

Современные тенденции также включают использование low-code платформ, таких как KNIME Analytics Platform с плагином Text Processing, который предоставляет узлы для синтаксического анализа PDF, предобработки текста (удаление стоп-слов, стемминг) и вычисления частотных характеристик.

Развитие технологий машинного зрения привело к появлению энд-ту-энд моделей на базе Large Vision-Language Models (LVLM), таких как Logics-Parsing, которые интегрируют OCR, распознавание таблиц и математических формул в едином конвейере,

достигая SOTA-показателей на бенчмарках сложных документов. Коммерческие решения, например UPDF 2.0, также внедряют функции «глубокого исследования» с доступом к академическим базам данных и генерацией структурированных обзоров литературы.

Несмотря на обилие подходов, наблюдается дефицит открытых решений, сочетающих в себе функции извлечения, углубленного статистического анализа и наглядной визуализации в едином программном продукте, доступном для настройки конечным пользователем. Данная работа направлена на частичное восполнение этого пробела.

Методология и архитектура разработанного парсера

Разработанный программный комплекс написан на языке Python, что обусловлено наличием богатой экосистемы библиотек для работы с PDF и анализа данных. Архитектура системы является модульной и включает следующие ключевые компоненты:

1. Модуль валидации и извлечения метаданных: отвечает за проверку целостности файла, формата и извлечение стандартной метаинформации (автор, название, дата создания, количество страниц) с использованием PyPDF2.
2. Модуль извлечения текста: реализован на базе библиотеки pdfplumber, которая показывает высокую эффективность при извлечении текста с сохранением структуры страниц и распознавании таблиц. Для страниц, не содержащих текстового слоя, предусмотрена интеграция с Tesseract OCR (опционально).
3. Модуль статистического анализа: выполняет токенизацию текста, очистку от стоп-слов (с поддержкой русского и английского языков), расчет частотности слов, лексического разнообразия и распределения слов по страницам.
4. Модуль визуализации: генерирует гистограммы распределения слов по страницам, горизонтальные столбчатые диаграммы топ-слов, круговые диаграммы доли ключевых терминов, облака слов (WordCloud) и тепловые карты частотности слов в разрезе страниц на основе библиотек matplotlib и seaborn.
5. Модуль экспорта: формирует структурированный Excel-файл с несколькими листами (метаданные, текст по страницам, статистика, топ-слова) с использованием pandas и openpyxl.
6. Модуль поиска: реализует контекстный поиск, по ключевым словам, с использованием регулярных выражений, возвращая номера страниц и фрагменты текста с найденными вхождениями.

Архитектура спроектирована с учетом возможности обработки нескольких файлов в пакетном режиме, что соответствует современным требованиям к масштабируемости.

Алгоритмы работы ключевых модулей

Алгоритм извлечения текста и таблиц. Процесс начинается с открытия файла и последовательного перебора страниц. Для каждой страницы вызывается метод extract_text() библиотеки pdfplumber, который анализирует позиции символов и группирует их в слова и строки. Параллельно выполняется поиск табличных структур через extract_tables(), основанный на обнаружении линий и группировке ячеек. Извлеченные данные сохраняются в словарь с привязкой к номеру страницы.

Алгоритм статистического анализа. После получения общего текста производится его токенизация с помощью nltk.word_tokenize. Токены фильтруются: удаляются знаки пунктуации и стоп-слова. Список стоп-слов формируется объединением стандартных наборов NLTK для английского и русского языков с возможностью ручного дополнения. Частотный анализ выполняется с использованием collections. Counter. Расчет лексического разнообразия производится как отношение количества уникальных слов к общему количеству слов в документе.

Алгоритм построения тепловой карты. Для визуализации распределения ключевых терминов по документу формируется матрица, где строки соответствуют топ-словам (например, 15 наиболее частотных), а столбцы – страницам. Значение ячейки – частота вхождения слова на конкретной странице. Матрица визуализируется с помощью seaborn.heatmap, что позволяет быстро выявить страницы, наиболее релевантные определенной тематике.

Экспериментальные результаты и обсуждение

Тестирование разработанного анализатора проводилось на коллекции из двух документов: «Описание УБИ СИИ ФСТЭК (январь 2026)» и научной статьи «ВНИ 46 ЦНИИ». Анализ первого документа (5 страниц) показал следующие результаты: общее количество слов – 1083, уникальных слов – 282. Топ-10 слов составили: искусственного (50), интеллекта (36), системы (30), машинного (24), безопасности (23), информации (21), обучения (18), угроз (15), модели (14), систем (13).

Функция поиска по ключевому слову «угроза» успешно идентифицировала страницы 2 и 4, что подтвердило корректность работы модуля индексации. Построенная тепловая карта для топ-слов позволила наглядно продемонстрировать, что термин «искусственного» наиболее часто встречается на страницах 2-3, в то время как слово «угроз» – на страницах 2 и 4.

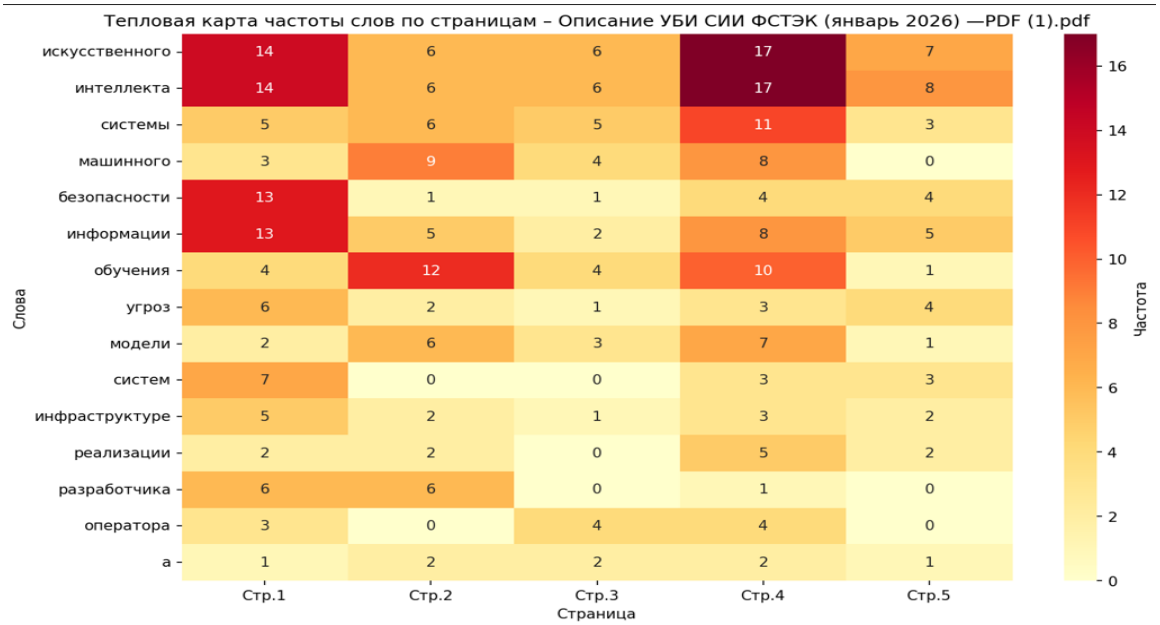

Рисунок 1 – Тепловая карта частоты слов по страницам

На рисунке 2, 3 представлены другие возможности парсера: выделение долей слов и распределение слов по страницам. Возможно применение и иной визуализации данных анализа текста. Время обработки одного файла составило в среднем 2,5 с (без OCR), что приемлемо для интерактивной работы. Экспорт в Excel сохраняет все данные на отдельных листах, включая постраничный текст и частотные словари.

Анализ второго документа показал иное распределение лексики, характерное для технической статьи. Время обработки одного файла составило в среднем 2-3 секунды (без учета OCR), что приемлемо для интерактивной работы. Экспорт результатов в Excel-файл позволил сохранить все извлеченные данные и статистику для дальнейшего использования в табличных процессорах.

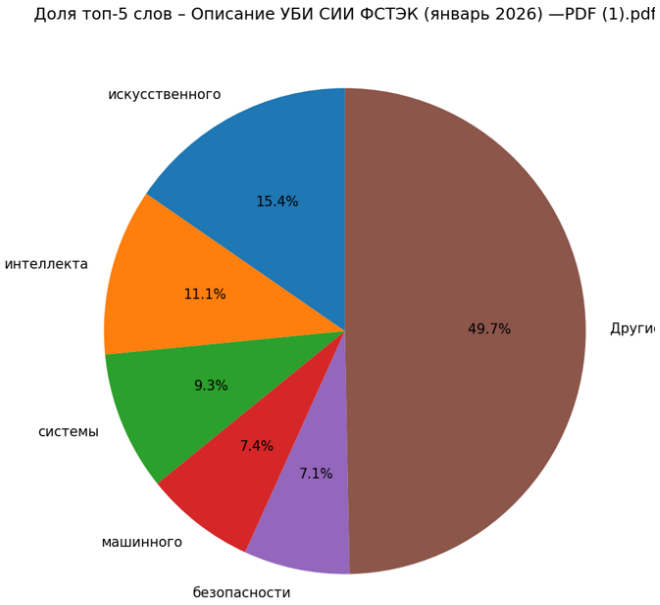

Рисунок 2 – Выделение долей слов и распределение слов по страницам

Рисунок 3 – Выделение долей слов и распределение слов по страницам

В ходе выполнения работы был разработан и протестирован программный комплекс для анализа PDF-документов, обладающий расширенными функциями статистической обработки и визуализации. Основные результаты включают:

1. Реализацию модулей извлечения текста, таблиц и метаданных из PDF-файлов.
2. Создание подсистемы лингвистического анализа с расчетом ключевых метрик (частотность слов, лексическое разнообразие).
3. Разработку набора инструментов визуализации, включая облака слов, тепловые карты и гистограммы распределения, что повышает наглядность анализа больших документов.
4. Обеспечение экспорта структурированных результатов в формат Excel для последующей обработки.

Практическая значимость работы заключается в создании готового к использованию инструмента, который может быть применен исследователями для анализа научной литературы, специалистами для обработки технической документации, а также в образовательных целях. Направления дальнейшего развития включают интеграцию с нейросетевыми моделями для семантического анализа и реализацию распределенной обработки для масштабируемых задач.

Конфликт интересов

Авторы статьи заявляют, что на момент передачи статьи в редакционную коллегию у них нет возможного конфликта интересов с третьими лицами.

Список источников

1. Обзор UPDF 2.0: кроссплатформенный PDF-редактор с глубокими исследованиями для более интеллектуальной работы [Электронный ресурс] // Letem světem Applem. – 2025. – Режим доступа: https://www.letemsvetemapplem.eu/ (дата обращения: 25.02.2026).
2. AdaParse: An Adaptive Parallel PDF Parsing and Resource Scaling Engine / S. Chia, [и др.] // arXiv preprint arXiv:2505.01435. – 2025. – Режим доступа: https://arxiv.org/abs/2505.01435 ,
3. Кулаков, М. Д. Разработка приложения для извлечения текста из файлов формата PDF / М. Д. Кулаков. – Иваново: ИГЭУ, 2025. – 74 с.
4. Автоматизация извлечения и анализа табличных данных с характеристиками электронных компонентов / [и др.] // naukaru.ru. – 2025. – Режим доступа: https://naukaru08.ru/ (дата обращения: 25.02.2026).
5. PDF-бот: проект для автоматической обработки PDF-документов и экспорта данных в Excel [Электронный ресурс] // GitHub. – 2025. – Режим доступа: https://github.com/SunnyS8/PDF_bot
6. Logics-Parsing Technical Report / [и др.] // arXiv preprint arXiv:2509.19760. – 2025. – Режим доступа: https://arxiv.org/abs/2509.19760
7. Егорова, Д. К. Application of KNIME Analytics Platform tools to analyze the compliance of syllabuses with the requirements of employers / Д. К. Егорова, Ю. В. Заварухина // Огарёв-online. – 2023. – № 11. – Режим доступа: https://journals.rcsi.science/
8. Об утверждении Положения о системе сертификации средств защиты информации Министерства обороны Российской Федерации: приказ МО РФ от 29.09.2020 № 488. – М., 2020.

9. Об утверждении Порядка проведения сертификации процессов безопасной разработки программного обеспечения средств защиты информации: приказ ФСТЭК России от 01.12.2023 № 240. – М., 2023.
10. Goodfellow I. J., Shlens J., Szegedy C. Explaining and Harnessing Adversarial Examples // International Conference on Learning Representations (ICLR). – 2015. – arXiv:1412.6572.
11. Об утверждении Положения о системе сертификации средств защиты информации Министерства обороны Российской Федерации: приказ МО РФ от 29.09.2020 № 488. – М., 2020.
12. Об утверждении Порядка проведения сертификации процессов безопасной разработки программного обеспечения средств защиты информации: приказ ФСТЭК России от 01.12.2023 № 240. – М., 2023.
13. Adversarial Robustness of Neural Networks: A Review / S. G. Finlayson et al. // arXiv preprint arXiv:2502.01234. – 2025.
14. Adhikari, N. S. A Comparative Study of PDF Parsing Tools Across Diverse Document Categories / N. S. Adhikari [и др.] // arXiv preprint arXiv:2410.09871. – 2025.
15. A Comparative Study of PDF Parsing Tools Across Diverse Document Categories [Электронный ресурс] // Harvard University ADS. – 2024. – Режим доступа: https://ui.adsabs.harvard.edu/abs/2024arXiv241009871A/abstract
16. OmniParser против Unstructured: какой пакет для разбора документов победит в 2025 году? [Электронный ресурс] // Sider AI. – 2025. – Режим доступа: https://sider.ai/ru/blog/ai-tools/omniparser-vs-unstructured-which-document-parsing-stack-wins-in-2025

Creating a PDF analyzer with advanced features

Vladislav Valerievich Dushkin[1], Victoria Alexandrovna Abramova [2*]

Krasnodar Higher Military School named after General of the Army S.M. Shtemenko, Krasnodar, Russia
[1]dushkin@list.ru, [2*]vabramova474@gmail.com

Annotation

The article discusses the process of designing and developing a software tool for complex analysis of PDF documents using the libraries PyPDF2, pdfplumber and Apache Tika, as well as the integration of machine learning to recognize the structure of documents. The architecture of the system with advanced functions is proposed: text extraction, tables, images, semantic analysis and processing of large amounts of data, the architecture of the developed Python application using the libraries PyPDF2, pdfplumber, pandas is described, and matplotlib. Algorithms for the operation of text extraction modules, word frequency analysis, building word clouds and heat maps, as well as exporting data in Excel format are presented. The results of testing the system on collections of scientific articles and technical documents are presented. The conclusion is made about the practical significance of the developed tool for researchers, analysts and specialists in working with documentation.

Keywords: PDF parser, machine learning, data extraction, text analysis, data visualization, Python, statistical processing, document mining, information security.