Информационные процессы / Information processes

Оригинальная статья / Original article
https://doi.org/10.31432/1994–2443.2025.11

Разработка прототипа системы распознавания и классификации корпоративных документов

И.В. Перлов ✉, С.А. Селиванов, А.В. Синицын, Ш.М. Шахгусейнов

Федеральное государственное бюджетное образовательное учреждение высшего образования «МИРЭА — Российский технологический университет»
проспект Вернадского, 78, г. Москва, 119454, Российская Федерация
✉ *perlovivan@yandex.ru*

Аннотация. *Актуальность*. В современных условиях становится важным повышение точности и скорости обработки документов.
Цель. Разработка системы конвертации, распознавания и классификации корпоративных документов в нередактируемых форматах.
Материалы и методы. В разработке использовался язык программирования Python 3.10, библиотеки scikit-learn 1.6, joblib и poppler, модуль Razdel, PyTorch 2.2, Hugging Face Transformers 4.39. пакеты PyPDF2 / pdfminer.six / pdfplumber; инструмент Tesseract OCR 5 с использованием pytesseract. Для устранения разрывов строк и уменьшения шума использовался пакет OpenCV-python. Веб-интерфейс строился на Vite и React с использованием Bootstrap 5.
Результаты. Разработан прототип системы, позволяющий эффективно конвертировать документ из нередактируемого формата в редактируемый в форме определенного документа.
Выводы. Использование технологий искусственного интеллекта ускоряет рабочие процессы, уменьшает окно ошибок. Решение интегрируется в рабочие процессы, но для обучения классификации требуется большое количество данных.

Ключевые слова: искусственный интеллект; извлечение информации; классификация документов; оптическое распознавание символов; извлечение сущностей; автоматизация документооборота

Финансирование. Финансирование отсутствовало.

Для цитирования: Перлов И.В., Селиванов С.А., Синицын А.В., Шахгусейнов Ш.М. Разработка прототипа системы распознавания и классификации корпоративных документов. *Информация и инновации*. 2025;20(2):41-57. https://doi.org/10.31432/1994-2443.2025.11

Development of a prototype system for recognizing and classifying corporate documents

Ivan V. Perlov ✉, Sergey A. Selivanov, Alexander V. Sinitsyn, Shamhal M. Shakhguseynov

Federal State Budgetary Educational Institution of Higher Education
"MIREA — Russian Technological University"
78, Vernadsky Avenue, Moscow, 119454, Russian Federation
✉ *perlovivan@yandex.ru*

Abstract. *Relevance*. In today's environment, improving the accuracy and speed of document processing is becoming increasingly important.
Target. Development of a system for converting, recognizing, and classifying corporate documents in non-editable formats.
Materials and Methods. The development utilized Python 3.10, the scikit-learn 1.6 library, joblib and poppler, the Razdel module, PyTorch 2.2, and Hugging Face Transformers 4.39. The PyPDF2 / pdfminer.six / pdfplumber packages; and the Tesseract OCR 5 tool using pytesseract. The OpenCV-python package was used to eliminate line breaks and reduce noise. The web interface was built on Vite and React using Bootstrap 5.
Results. A prototype system was developed that enables efficient document conversion from a non-editable format to an editable one within a specific document.
Conclusions. The use of artificial intelligence technologies accelerates workflows and reduces the error window. The solution integrates into workflows, but classification training requires a large amount of data.

Keywords: artificial intelligence; information extraction; document classification; optical character recognition; named entity recognition; automation of document flow

Funding. No funding.

For citation: Perlov I.V., Selivanov S.A., Sinitsyn A.V., Shakhguseynov S.M. Development of a prototype system for recognizing and classifyingcorporate documents. *Information and Innovations*. 2025;20(2):41-57. (In Russ.). https://doi.org/10.31432/1994-2443.2025.11

ВВЕДЕНИЕ

В условиях стремительной цифровой трансформации объём электронных документов, циркулирующих в корпоративной, государственной и научно-образовательной среде, неуклонно растёт, а форматы нередактируемых файлов — прежде всего PDF, сканы и фотоснимки — остаются преобладающими носителями информации. Их распространённость объясняется универсальностью формата и гарантированным сохранением визуального оформления, однако именно эти преимущества оборачиваются серьёзными препятствиями при последующей обработке: традиционные методы оптического распознавания символов сталкиваются с многообразием кодировок, шрифтов и встраиваемых графических объектов, что приводит к падению точности извлечения текста и искажению содержимого. Для российской деловой практики проблема усугубляется регламентированным оборотом большого числа унифицированных бланков и форм; неэффективное ручное переписывание или пост-коррекция распознанных документов затягивает бизнес-процессы и повышает риск ошибок.

Актуальность исследования определяется необходимостью комплексной автоматизации всего цикла работы с такими документами: от извлечения текста до его структурирования, тематического отнесения и переноса в редактируемые форматы. Интеграция современных методов искусственного интеллекта позволяет объединить на единой технологической платформе OCR-распознавание, классификацию с помощью искусственного интеллекта по типовым шаблонам и задачу именованного распознавания сущностей, а затем автоматически заполнять стандартизированные редактируемые формы. Подобный сквозной подход обеспечивает воспроизводимое качество данных, устраняет ручной труд и существенно сокращает время реагирования организаций на входящий документопоток.

Целью данной работы является разработка системы конвертации с функциями распознавания и классификации различных типов нередактируемых корпоративных документов для коррекции в офисном формате с применением методов искусственного интеллекта.

МАТЕРИАЛЫ И МЕТОДЫ

Язык программирования Python 3.10 стал единым исполнением для всего прототипа: он сочетает динамическую типизацию с богатой экосистемой научных пакетов и позволяет запускать трансформерные модели на CPU или GPU без смены среды.

NumPy — фундаментальная библиотека векторных вычислений, реализующая массивы ndarray и приближающая производительность к C/Fortran-коду; все остальные компоненты стека опираются на её тензорную модель памяти.

Библиотека scikit-learn 1.6 предоставляет реализацию TF–IDF-векторизации и линейного SVM-классификатора (LinearSVC с ядром liblinear), что обеспечивает быструю и интерпретируемую тематическую классификацию текстов.

Библиотека joblib используется для сериализации моделей scikit-learn: он сохраняет матрицы весов в сжатом бинарном формате, совместимом с различными версиями Python.

Razdel — лёгкий русскоязычный токенизатор, устойчивый к пунктуационному шуму нормативных текстов и не требующий загрузки крупных словарей.

Библиотеки pandas и tqdm обеспечивают потоковое формирование корпусов

и полоску прогресса при многопоточном разборе XML-файлов, что упрощает подготовку разметки со слабым надзором.

Современную трансформерную часть реализует связка PyTorch 2.2 и Hugging Face Transformers [1] 4.39. PyTorch обеспечивает автоматическое дифференцирование и CUDA-ускорение, Hugging Face Transformers — высокоуровневые классы AutoModel и Trainer, позволяющие дообучить ruBERT-модель. Библиотека Accelerate 0.27 автоматически конфигурирует устройство вычислений и управляет распределёнными градиентами. Для расчёта показателей качества применяется пакет seqeval, специализированный на метриках последовательной разметки (точность, полнота, F1-мера для NER [2]). Весы модели сохраняются в формате safetensors, устойчивом к частичной порче данных и потоковому чтению.

Подготовка и хранение датасетов основаны на Hugging Face Datasets 3.6: библиотека поддерживает ленивую загрузку из JSON, параллельную предобработку и транзакционный экспорт в Arrow-формате.

Извлечение текстового и табличного содержимого из PDF-документов реализовано на Python с использованием PyPDF2, pdfminer.six и pdfplumber. PyPDF2 позволяет управлять геометрией страниц и выделять отдельные регионы документа; pdfminer.six предоставляет посимвольный доступ к текстовым блокам и информации о шрифтах, а pdfplumber облегчает извлечение структурированных таблиц в виде вложенных списков. Для работы с графикой применяется Pillow (PIL) и pdf2image, рендерящие страницы в растровые изображения. В случаях отсутствия текстового слоя или наличия изображений используется OCR на базе Tesseract 5 [3], а OpenCV-python устраняет артефакты разрыва строк и уменьшает шум контраста с последующим извлечением контуров ячеек.

Фронтенд приложения построен на Vite с использованием React и Bootstrap 5. Интерфейс предоставляет авторизацию двумя способами: через форму (валидация логина и пароля на клиенте) и OAuth-авторизацию от Google с помощью пакета @react-oauth/google и декодера jwt-decode. Основной компонент приложения реализует drag-and-drop загрузку документов, проверку их типа и размера, а также отображение результатов извлечения сущностей с сохранением их в состоянии React. Формируемые данные могут быть отредактированы пользователем и отправлены в виде структуры с текстом, сущностями и выбранным шаблоном документа.

Для автоматической генерации выходного делового документа применён пакет python-docx. Он даёт программный доступ к структуре DOCX, что позволяет искать плейс-холдеры по цвету шрифта, заменять их на извлечённые реквизиты и сохранять итоговый файл с исходным оформлением Word.

РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

1. Постановка задачи

Задача данного прототипа — показать возможность и эффективность использования технологий искусственного интеллекта при работе с нередактируемыми типами документов. Она должна самостоятельно получить данные из документа, классифицировать по типу и определить необходимые сущности для заполнения редактируемого шаблона и выдать пользователю. Для этого необходимо собрать скрипт распознавания данных с помощью технологий OCR, который из документа будет получать текст и передавать его в модули классификатора документа и извлечения сущностей, которыми заполнят шаблон.

2. Серверная часть

2.1. Архитектура прототипа

Прототип прост в своей структуре — авторизация, файл загружается, обрабатывается и на выходе получается заполненный шаблон в редактируемом формате. Изобразим диаграмму DFD (Data Flow Diagram) на рис. 1:

Рис. 1. *Верхний уровень DFD-диаграммы прототипа*
Fig. 1. *Top level of the prototype DFD diagram*

И теперь декомпозируем, рис. 2:

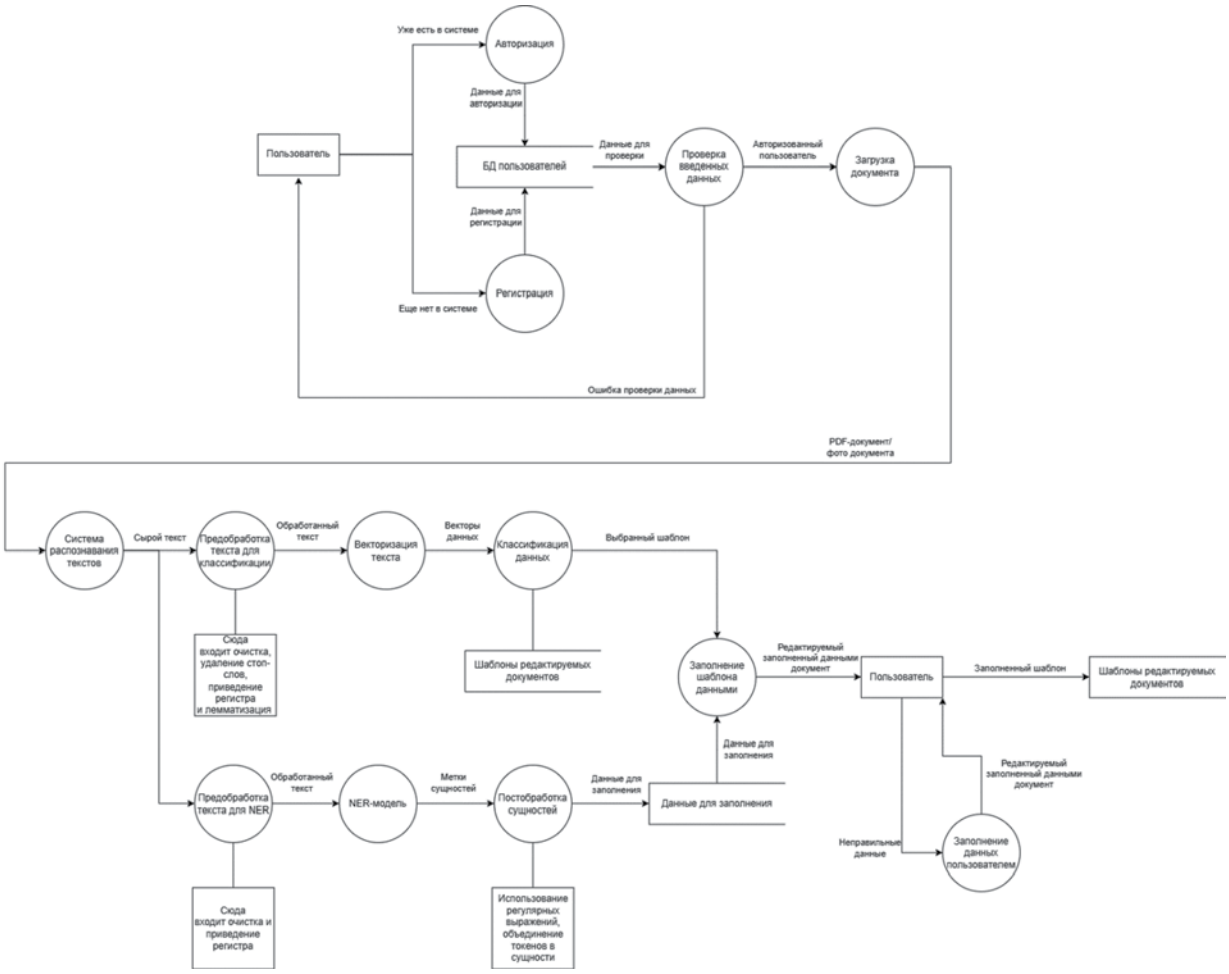


Рис. 2. *Декомпозиция DFD — диаграммы прототипа*
Fig. 2. *Decomposition of DFD — prototype diagram*

Авторизовавшись или зарегистрировавшись, пользователь, загружая документ, передает его в модуль распознавания текстов, задача которой определить содержимое документа. Сырой текст дальше передается в две «ветки» — одна отвечает за классификацию всего документа, вторая — за распознавание сущностей в нем. У каждой идет своя предобработка — для классификатора это очистка, удаление стоп-слов, приведение регистра и лемматизация, а для NER — только очистка и приведение регистра. После этого NER-модель начинает находит токены в тексте, а классификатор сначала отдельно векторизует данные и потом только определяет шаблон. Для NER же дополнительно происходит постобработка сущностей, чтобы с помощью регулярных выражений найти недостающие сущности. И потом эти данные объединяются — с сервера берется нужный шаблон и заполняется данными в нужные места документа в редактируемом формате. Если же данные были неправильно заполнены — пользователь заполняет их вручную или выбирает из предложенных.

2.2. Обучение распознавателя

Для повышения точности распознавания слабоконтрастных или искажённых символов были дообучены модели Tesseract 5 на двух специализированных датасетах: OCR-Cyrillic-Printed-8 (синтетические строки на кириллице) и Brno Mobile OCR Dataset (B-MOD) — изображения текста, снятые с мобильных устройств при различных уровнях сложности условий съёмки [4]. Обе выборки были конвертированы в формат, необходимый для системы Tesstrain (.tif и .gt.txt), с последующей генерацией .box и .lstmf файлов. Использовалось дообучение на основе предобученных моделей rus и eng, ограниченное 10 000 итерациями.

Результаты (рис. 3-4) продемонстрировали значительное снижение Character Error Rate (CER) [5] на тренировочных данных. Модель custom_rus достигла минимальной ошибки 8.888 %, стабилизировавшись на уровне 10.300 %, в то время как custom_eng показала минимум 11.981 %, но с признаками переобучения на поздних итерациях. Хотя на стандартных документах прирост точности был незначителен, на низкокачественных и архивных сканах дообученные модели показывают более высокий процент точности распознавания символов, чем стандартные модели. Дообученные модели были успешно интегрированы в общий пайплайн извлечения текстовых данных из загружаемых в систему документов.

2.3. Архитектура распознавателя

Данный компонент системы представляет собой многофункциональный модуль для извлечения текстовой и табличной информации из PDF-документов, включая документы с изображениями страниц. Он был разработан как часть системы обработки юридических и корпоративных документов, включая случаи, когда оригинальные тексты представлены в виде сканированных изображений без текстового слоя. Основная цель скрипта — предоставить универсальный подход к анализу содержимого PDF, независимо от сложности его структуры.

Для документов с текстовым слоем применяется связка библиотек pdfminer и pdfplumber: первая отвечает за посимвольное извлечение текстовых блоков и информации о шрифтах, вторая — за точное распознавание таблиц, представленных в структурированном виде. Извлечённые таблицы преобразуются в удобный для дальнейшего анализа формат (markdown-подобный), а текстовые блоки сортируются по координате

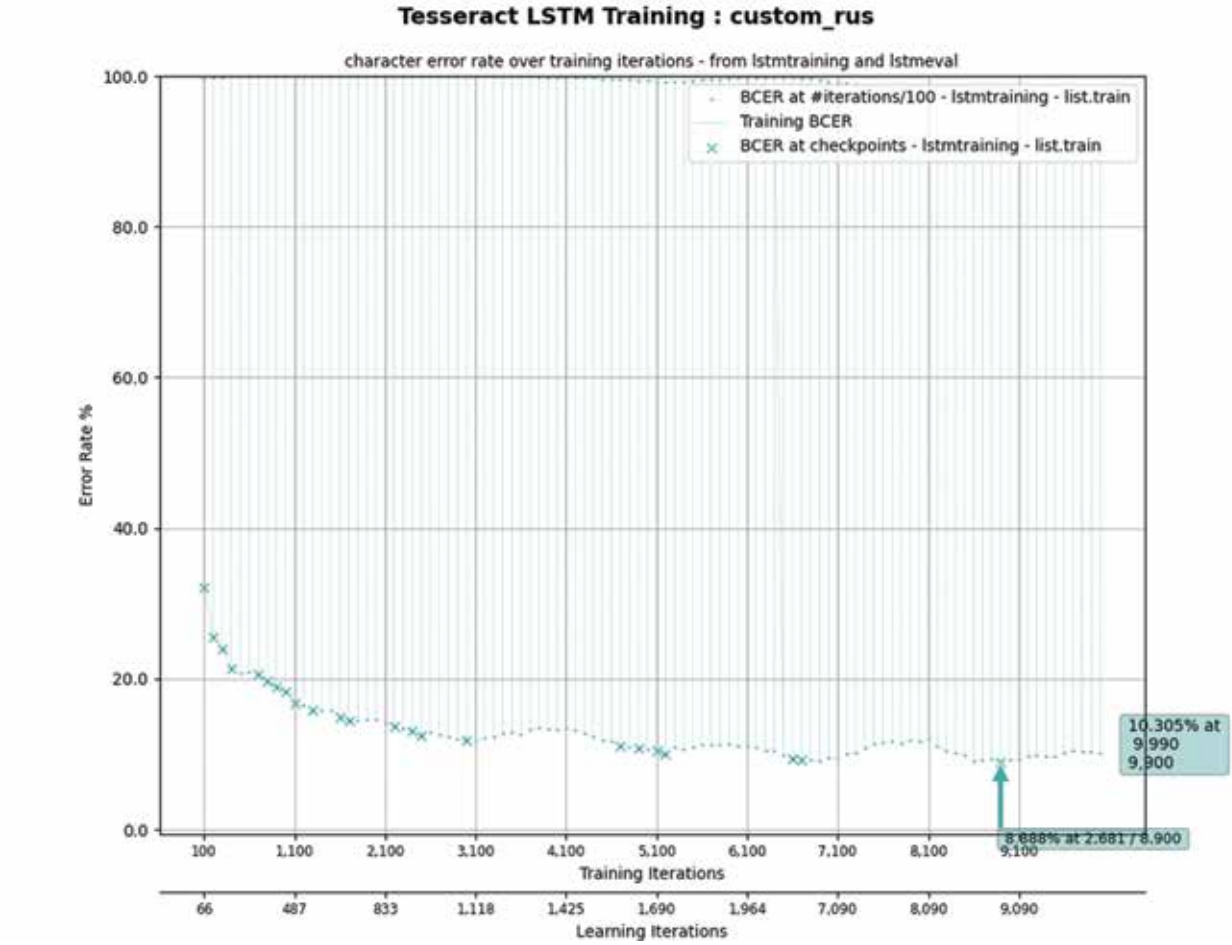


Рис. 3. *Результаты обучения для русских символов*
Fig. 3. *Learning results for Russian characters*

Y для восстановления логического порядка.

В случаях, когда документ представлен в виде изображения или содержит встроенные векторные элементы без текстового слоя, скрипт переходит к обработке на уровне изображения. Сначала осуществляется обрезка интересующего элемента (LTFigure) и конвертация его в растровый формат PNG с использованием PyPDF2 и pdf2image. Далее применяется комплексная обработка средствами OpenCV: бинаризация методом Оцу, морфологическая фильтрация с использованием структурных элементов (ядер) для выделения горизонтальных и вертикальных линий таблиц, объединение их в общую маску и извлечение контуров ячеек. Полученные ячейки распознаются инструментом pytesseract. На выходе формируется таблица с распознанным текстом, что позволяет корректно обрабатывать даже зашумленные и графически сложные документы.

2.4. Датасет для классификатора и модуля извлечения сущностей

Для начала нужно рассказать о корпусе данных, который использовался для обучения классификатора. RusLawOD — это корпус текстов законодательных актов Российской Федерации и их метаданных за период с 1991 по 2023 год, содержит в своей облегченной версии около 281233

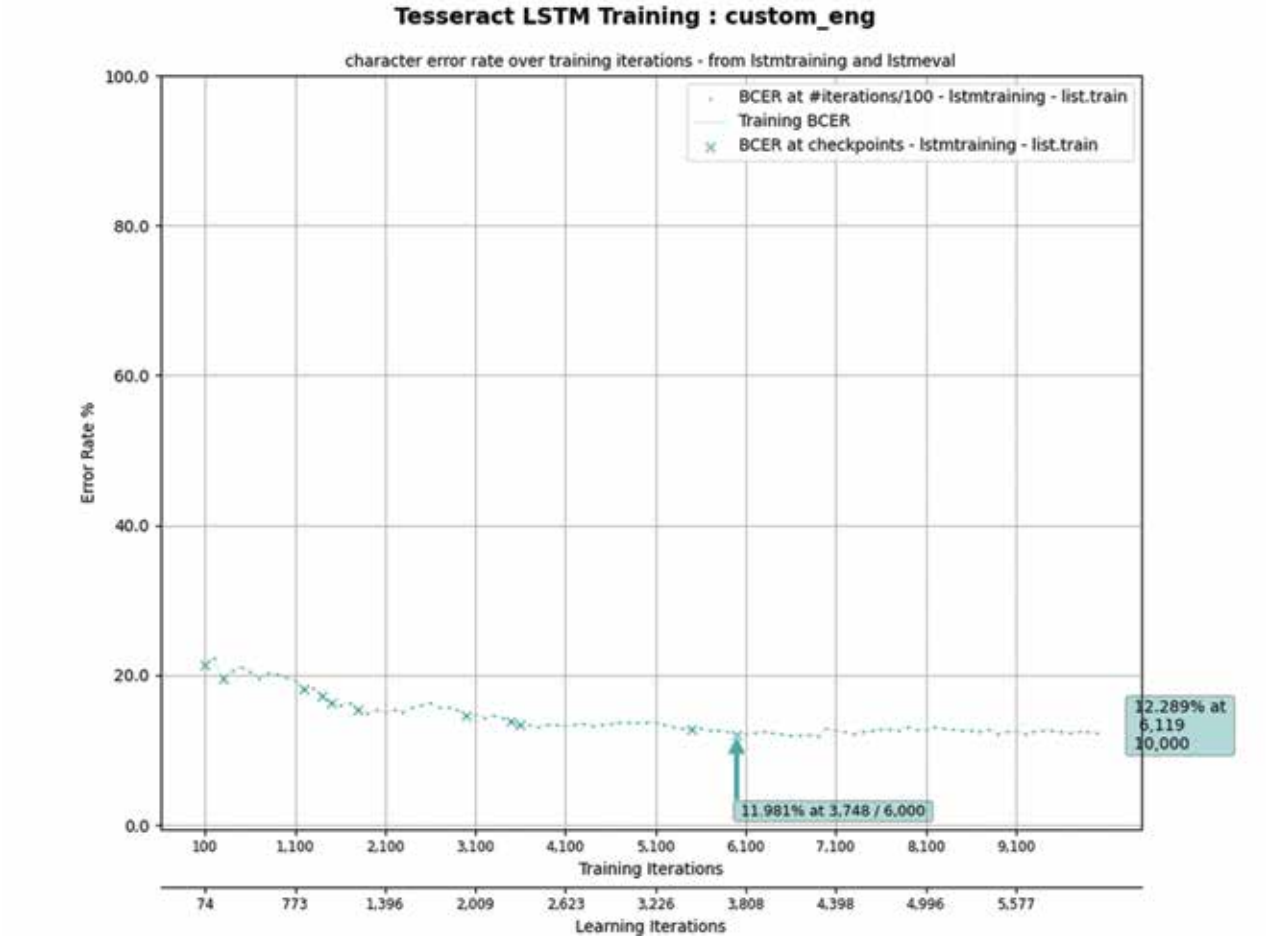


***Рис. 4**. Результаты обучения для английских символов*
***Fig. 4.** Learning results for English characters*

документов в формате XML. Структура каждого XML-документа такова:

– в тэге <body> находится тег textIPS, который содержит весь текст из документа, загруженного в информационно-правовую систему «Законодательство Российской Федерации»;

– дальше идут теги act, meta, identification, внутри которых нас интересуют 4 тега для задачи определения сущностей:

- doc_author_normal_formIPS со значением val — принявший правовой акт орган власти, может отсутствовать;
- docdateIPS — содержит дату подписания документа по сведениям ИПС "Законодательство России", строка вида дд.мм.гггг может отсутствовать у документов;
- docNumberIPS — строка юридического номера документа. Может содержать значение б/н, когда такой номер отсутствует официально. Может отсутствовать у документов, которые не были опубликованы в ИПС Законодательство России;
- signedIPS — строка с ФИО человека, подписавшего документ.

– doc_typeIPS — вид документа, строка с фиксированными значениями из классификатора принимающих органов. Может отсутствовать у документов, кото-

рые не были опубликованы в ИПС Законодательство России. Будет использоваться для классификации документов.

2.5. Архитектура классификатора

Перейдем к скрипту классификации ruslawod_classifier.py. На этапе предварительной обработки он параллельно разбирает XML-файлы RusLawOD. Длинные документы усекались до 10 000 токенов, для токенизации использовалась библиотека Razdel, обеспечивающая корректное разбиение русского официально-делового текста. После очистки корпус разделялся стратифицированно (80 % — обучение, 20 % — тестовые). Архитектура модели состоит из двуграммного TF–IDF-векторизатора [6] и линейного SVM [7] (ядро liblinear) с автоматической балансировкой классов. Обучение на 228 тыс. актов заняло 15 минут на CPU; итоговая точность 0,998, макро-F1-мера 0,976. Итоговый векторизатор и веса классификатора сериализуются при помощи joblib, образуя файл ruslawod_tfidf_svm.joblib.

Диаграмма DFD процесса обучения представлена на рис. 5.

2.6. Архитектура модуля извлечения сущностей

Для извлечения сущностей «орган-издатель, дата, номер, подпись» [8] использована двухэтапная схема. Сначала скрипт build_ner_dataset.py формирует слабонаблюдаемый датасет [9]: из тех же XML берутся соответсвующие поля, затем регулярными выражениями эти строки ищутся в полном тексте и переводятся в BIO-разметку. Параллельный парсинг 280 тыс. файлов и токенизация позволили собрать ≈ 195 тыс. размеченных пред-

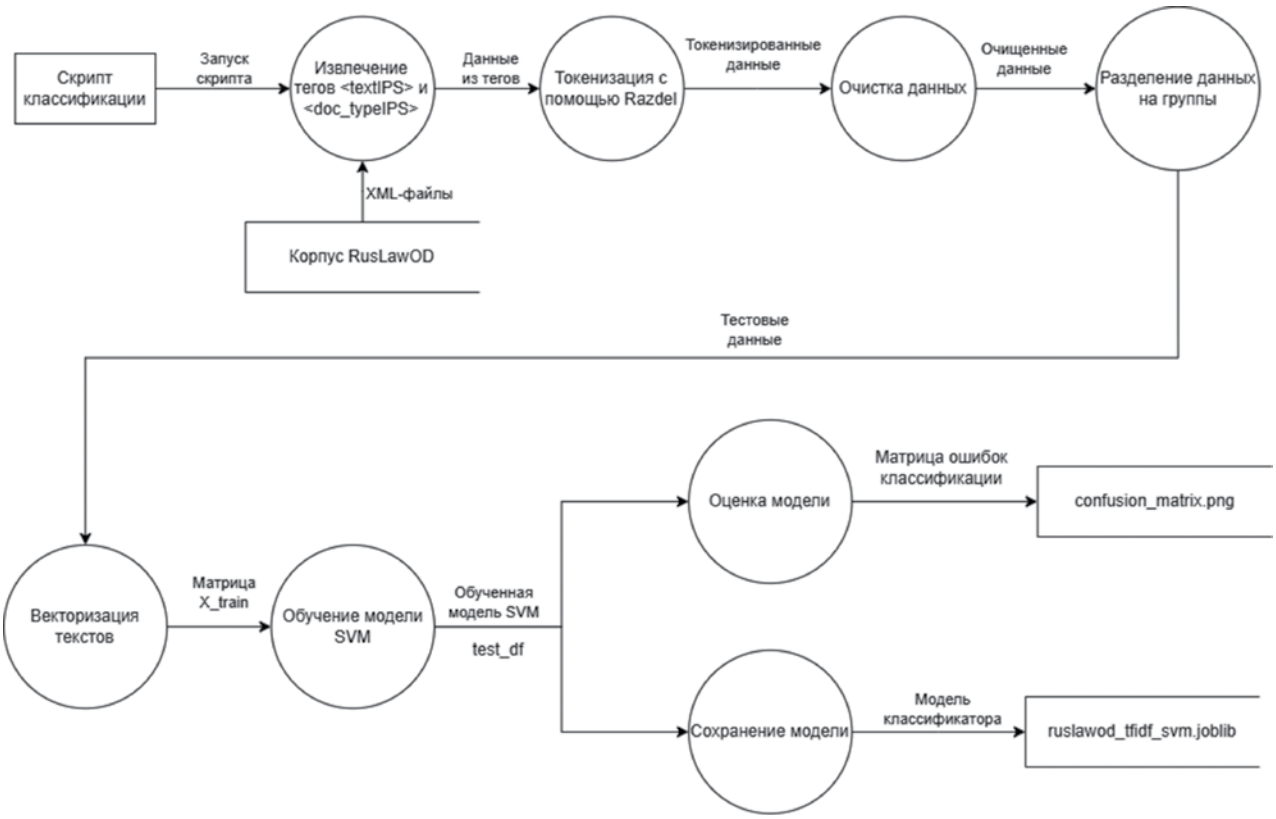


Рис. 5. *Диаграмма DFD процесса обучения классификатора*
Fig. 5. *DFD diagram of the classifier training process*

ложений; они сохранены в JSON-файлах train/valid/test.

Дообучение трансформерной модели [10] выполняет скрипт train_hf_ner.py. В качестве базового чекпойнта выбран ruBERT-base [11] (12 слоёв, 768 скрытых нейронов). Выходная голова классификации токенов переинициализируется на девять классов (би-теги и фон O). Параметры обучения: оптимизатор AdamW, скорость обучения $5 \cdot 10^{-5}$, весовой спад $1 \cdot 10^{-2}$, размер пакета 16, длина последовательности 256 токенов, число эпох 3. Обучение на GPU заняло 2 ч. времени.

Контрольные метрики рассчитывает скрипт classification_report.py. Он загружает сохранённые веса, выполняет инференс на test-сплите датасета и с помощью пакета seqeval формирует отчёт. На контрольной выборке получены: микровзвешенная F1-мера 0,998; по классам AUTHORITY — 0,944, DATE — 0,981, DOC_NUM — 0,974, SIGNATORY — 0,684. Низкая полнота подписи объясняется отсутствием тега <signedIPS> в ≈ 30 % актов.

Диаграмма DFD для скриптов NER — ниже на рис. 6.

2.7. Интеграция компонентов

Интеграцию компонентов в единый конвейер реализует скрипт process_legal_pdf.py. Входом служит произвольный PDF-файл нормативного акта. Текст извлекается гибридным конвейером: при наличии текстового слоя — через pdfminer.six, иначе страница рендерится pdftoppm (poppler) и распознаётся Tesseract OCR 5. Далее текст нормализуется и передаётся на TF-IDF-классификатор. Предсказанная

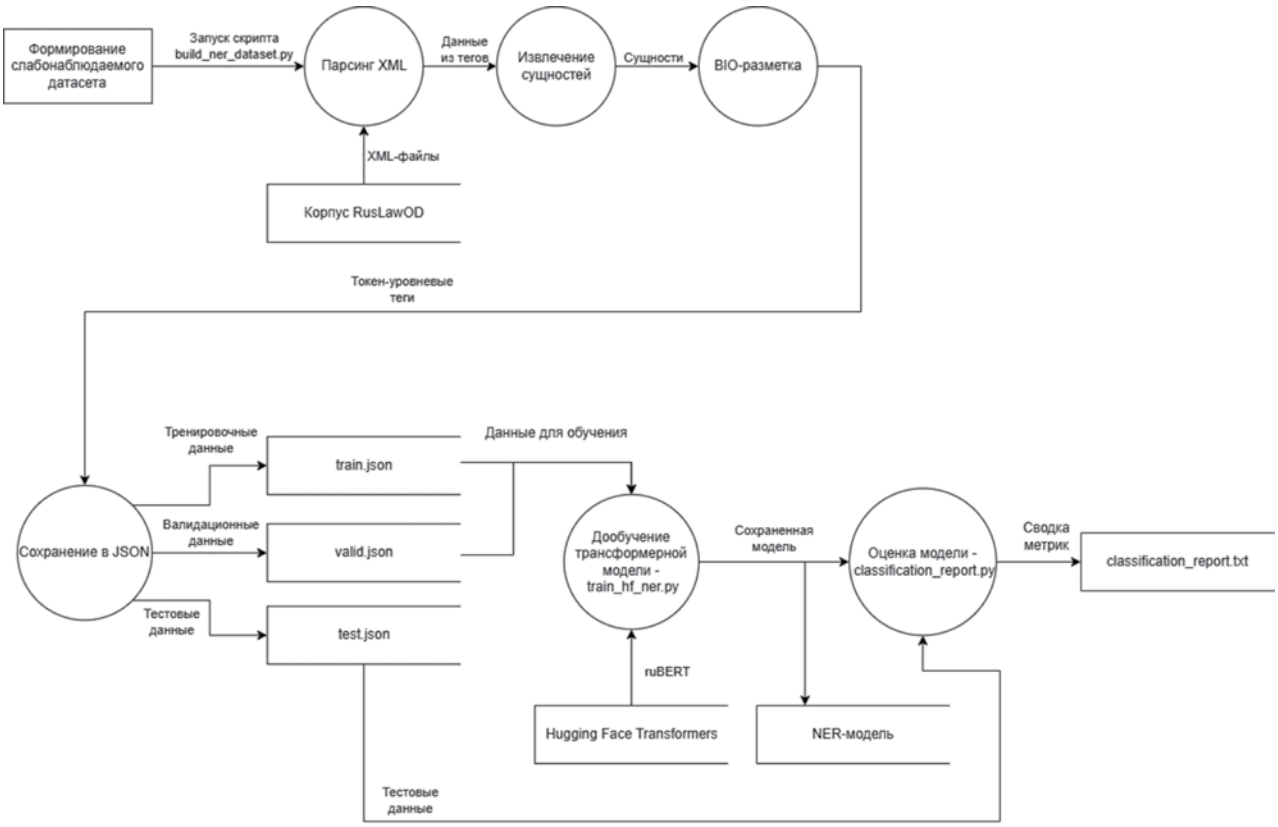


Рис. 6. *Диаграмма DFD процесса обучения модели извлечения сущностей*
Fig. 6. *DFD diagram of the entity extraction model training process*

метка типа документа и сам текст поступают в BERT-NER-пайплайн, который возвращает агрегированные сущности. Дополнительный пост-процесс добавляет дату и номер регулярными выражениями, если модель их пропустила. Результат сохраняется в *.out.json. Таким образом, один вызов скрипта реализует полный цикл «PDF → класс акта → реквизиты».

3. Клиентская часть

3.1. Компонент загрузки файлов

В верхней части страницы размещён компонент загрузки файлов. Он поддерживает два способа добавления документа:

«Кнопка загрузки»: при нажатии открывается диалоговое окно выбора файла.

«Перетаскивание файла»: пользователь может перетащить файл в специально обозначенную зону (drag-and-drop), которая визуально выделена и содержит соответствующую инструкцию (например: «Перетащите файл сюда или нажмите, чтобы выбрать»).

После загрузки файла происходит его автоматическая отправка на сервер и последующая обработка.

3.2. Основной раздел работы с документом

Расположен ниже компонента загрузки и состоит из двух частей:

Центральный блок — извлечённый текст.

В этой области отображается весь текст, извлечённый из загруженного документа. Это текстовое поле с возможностью прокрутки, при необходимости редактируемое.

Правая боковая панель — список сущностей.

В виде вертикального списка расположены выпадающие списки (компоненты типа dropdown), каждый из которых содержит определённую категорию извлечённых сущностей: наименование органа, ФИО, даты, другие ключевые данные (например, должности, адреса и т. п.).

Каждый выпадающий список позволяет просматривать, редактировать или удалять найденные значения.

3.3. Раздел формирования итогового документа

Ниже основного раздела размещён блок взаимодействия с шаблонами:

Выбор шаблона.

Выпадающий список или компонент выбора, позволяющий пользователю выбрать один из доступных шаблонов документов, хранящихся в системе.

Кнопка «Отправить».

При нажатии на кнопки:

Извлечённый и, при необходимости, откорректированный текст.

Список выбранных и подтверждённых сущностей.

Информация о выбранном шаблоне.

— всё это отправляется на сервер.

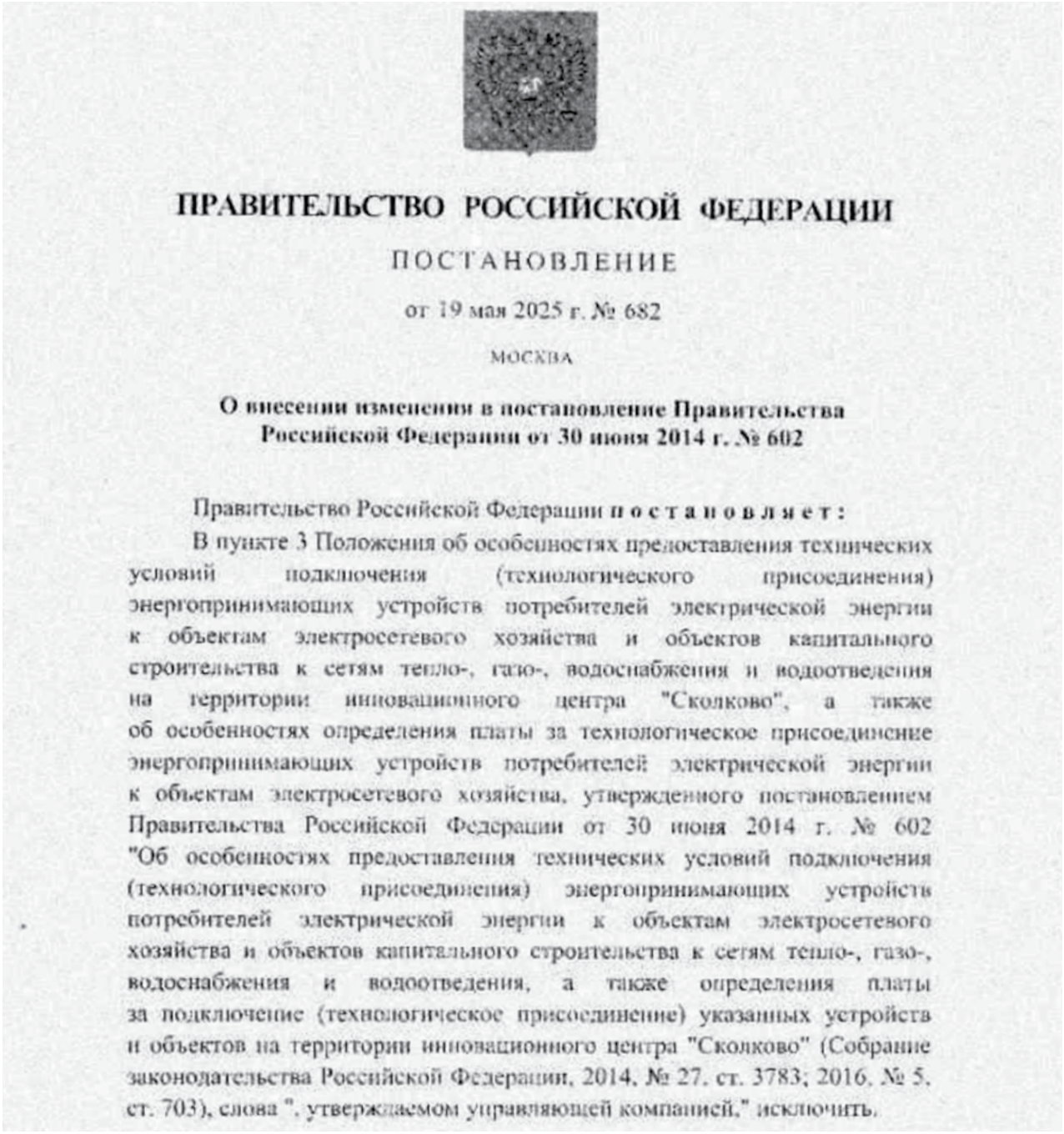
Сервер на основе полученных данных формирует финальный документ в формате Word (.docx), который автоматически предлагается пользователю для скачивания.

4. Демонстрация работы прототипа

Запустим интеграционный скрипт на документе, представленном на рис. 7 ниже.

У данного изображения было искусственно уменьшено качество — добавлены шумы, уменьшено разрешение и повышена яркость — все в целях тестирования на нечетком изображении. Результат распознавания и классификации представлены на рис. 8.

Как видно, тип документа определен верно, так же как и номер, дата документа, орган принявший его. Отсутствие подписи объясняется малым количеством данных для обучения. JSON выглядит немного иначе (содержит информацию о местах начала и конца сущностей), но вся основная информация вынесена для удобства восприятия в вышеуказанный текстовый



ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

ПОСТАНОВЛЕНИЕ

от 19 мая 2025 г. № 682

МОСКВА

О внесении изменения в постановление Правительства Российской Федерации от 30 июня 2014 г. № 602

Правительство Российской Федерации **п о с т а н о в л я е т :**

В пункте 3 Положения об особенностях предоставления технических условий подключения (технологического присоединения) энергопринимающих устройств потребителей электрической энергии к объектам электросетевого хозяйства и объектов капитального строительства к сетям тепло-, газо-, водоснабжения и водоотведения на территории инновационного центра "Сколково", а также об особенностях определения платы за технологическое присоединение энергопринимающих устройств потребителей электрической энергии к объектам электросетевого хозяйства, утвержденного постановлением Правительства Российской Федерации от 30 июня 2014 г. № 602 "Об особенностях предоставления технических условий подключения (технологического присоединения) энергопринимающих устройств потребителей электрической энергии к объектам электросетевого хозяйства и объектов капитального строительства к сетям тепло-, газо-, водоснабжения и водоотведения, а также определения платы за подключение (технологическое присоединение) указанных устройств и объектов на территории инновационного центра "Сколково" (Собрание законодательства Российской Федерации, 2014, № 27, ст. 3783; 2016, № 5, ст. 703), слова ", утверждаемом управляющей компанией," исключить.

Рис. 7. *Тестовый нередактируемый файл*
Fig. 7. *Test uneditable file*
Источник / Source: документ / document

файл. Скрипт выполнялся около 10 секунд. Далее скрипт заполнит шаблон полученными данными, результат на рис. 9.

Имеется ряд интересных работ по тематике данной статьи. В работе [12] представлены усовершенствованные методы оптического распознавания символов (OCR) для распознавания выражений в отсканированных документах. Исследование демонстрирует, как новые алгоритмы

```
Тип акта: Постановление

=== Извлечённые реквизиты ===
LABEL_0: | правительство россиискои федерации постановление от 19 мая 2025 г. №
LABEL_5: 682
LABEL_3: 19.05.2025
LABEL_0: москва о внесении изменения в постановление правительства россиискои федерации от 30 июня 2014 г. № 602
LABEL_1: правительство
LABEL_2: россиискои федерации
LABEL_0: постановляет : в пункте 3 положения об особенностях предоставления технических условии подключения ( техн

=== Текст ===
|ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ПОСТАНОВЛЕНИЕ
от 19 мая 2025 г. № 682
МОСКВА
О внесении изменения в постановление Правительства
Российской Федерации от 30 июня 2014 г. № 602

Правительство Российской Федерации постановляет:

В пункте 3 Положения об особенностях предоставления технических
условий подключения (технологического присоединения)
энергопринимающих устройств потребителей электрической энергии
к объектам электросетевого хозяйства и объектов капитального
строительства к сетям тепло-, газо-, водоснабжения и водоотведения
на территории инновационного центра "Сколково", а также
об особенностях определения платы за технологическое присоединение
энергопринимающих устройств потребителей электрической энергии
к объектам электросетевого хозяйства, утвержденного постановлением
Правительства Российской Федерации от 30 июня 2014 г. № 602 |
"Об особенностях предоставления технических условий подключения
(технологического присоединения) энергопринимающих устройств

‘ потребителей электрической энергии к объектам электросетевого
хозяйства и объектов капитального строительства к сетям тепло-, газо-,
водоснабжения и водоотведения, а также определения платы
за подключение (технологическое присоединение) указанных устройств
и объектов на территории инновационного центра "Сколково" (Собрание
законодательства Российской Федерации, 2014, № 27, ст. 3783; 2016, № 5,
ст. 703), слова ", утверждаемом управляющей компанией," исключить.
```

Рис. 8. *Результат распознавания и классификации документа*
Fig. 8. *Result of document recognition and classification*

повышают точность обработки сложных формул в научных текстах.

В статье [13] описано применение методов машинного обучения для классификации переведённых и оригинальных корпоративных годовых отчётов. Результаты показывают высокую эффективность модели при выявлении языковых и стилистических различий.

В [14] проведен масштабный обзор научных публикаций, посвящённых внутренним корпоративным документам.

ПРАВИТЕЛЬСТВО РОССИИСКОИ ФЕДЕРАЦИИ

ПОСТАНОВЛЕНИЕ

от 19.05.2025 г. № 682

МОСКВА

О внесении изменения в постановление Правительства Российской Федерации от 30 июня 2014 г. № 602

Правительство Российской Федерации постановляет:

В пункте 3 Положения об особенностях предоставления технических условий подключения (технологического присоединения) энергопринимающих устройств потребителей электрической энергии к объектам лектросетевого хозяйства и объектов капитального строительства к сетям тепло-, газо-, водоснабжения и водоотведения на территории инновационного центра "Сколково", а также 06 особенностях определения платы за технологическое присоединение энергопринимающих устройств потребителей электрической энергии к объектам электросетевого хозяйства, утвержденного постановлением Правительства Российской Федерации ог 30 июня 2014 г. № 602 | "Об особенностях предоставления технических условий подключения (технологического присоединения) энергопринимающих устройств ‘ потребителей электрической энергии к объектам электросетевого хозяйства и объектов капитального строительства к сетям тепло-, газо-, водоснабжения и водоотведения, а также определения платы за подключение (технологическое присоединение) указанных устройств и объектов на территории инновационного центра "Сколково" (Собрание законодательства Российской Федерации, 2014, № 27, ст. 3783; 2016, № 5, ст. 703), слова ", утверждаемом управляющей компанией," исключить.

Рис. 9. *Результат заполнения шаблона*
Fig. 9. *Result of filling out the template*

ЗАКЛЮЧЕНИЕ

Предложена и реализована система автоматической конвертации нередактируемых документов в редактируемые форматы с использованием современных методов искусственного интеллекта. Архитектура решения включает гибридный модуль извлечения текста из PDF-документов, классификатор типов актов на основе линейного SVM и модуль извлечения сущностей, основанный на дообученной трансформерной модели ruBERT. Клиентская часть обеспечивает интуитивный веб-интерфейс с визуализацией извлечённой информации и поддержкой выбора шаблонов. Результаты экспериментов подтверждают высокую точность классификации и извлечения

реквизитов, включая документы со слабоконтрастным или зашумлённым текстом. Разработанное решение масштабируемо, легко интегрируется в существующие бизнес-процессы и снижает вероятность ошибок, ускоряя обработку нормативных и корпоративных документов.

ВКЛАД АВТОРОВ

И.В. Перлов — сбор данных, концептуализация, анализ информации, подготовка текста.

С.А. Селиванов — анализ информации, концептуализация.

А.В. Синицын — концептуализация, сбор и анализ информации.

Ш.М. Шахгусейнов — сбор данных, концептуализация, анализ информации, подготовка текста.

CONTRIBUTION OF THE AUTHORS

Ivan V. Perlov — data collection, conceptualization, text preparation.

Sergey A. Selivanov — information analysis, conceptualization.

Alexander V. Sinitsyn — conceptualization, information collection and analysis.

Shamhal M. Shakhguseynov — data collection, conceptualization, text preparation.

КОНФЛИКТ ИНТЕРЕСОВ

Авторы заявляют об отсутствии конфликта интересов.

CONFLICT OF INTEREST

The authors declare no conflict of interests.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Su J., Ahmed M., Lu Yu., Pan Sh., Bo W., Liu Yu. RoFormer: Enhanced transformer with Rotary Position Embedding. *Neurocomputing*. 2024;568:127063. https://doi.org/10.1016/j.neucom.2023.127063
2. Romero-Fresco P. Subtitling through Speech Recognition: Respeaking. Manchester: St. Jerome, 2011. 261 p. ISBN 9781905763283.
3. Park J., Lee E., Kim Y., Kang I., Koo H.I., Cho N.I. Multi-Lingual Optical Character Recognition System Using the Reinforcement Learning of Character Segmenter. *IEEE Access*. 2020;8:174437-174448. https://doi.org/10.1109/ACCESS.2020.3025769
4. Memon J., Sami M., Khan R.A. Handwritten Optical Character Recognition (OCR): Comprehensive Systematic Literature Review (SLR). *IEEE Access*. 2020;8:142642-142668. https://doi.org/10.1109/ACCESS.2020.3012542
5. Hossain A., Ali M. Recognition of Handwritten Digit using Convolutional Neural Network (CNN). *Global Journal of Computer Science and Technology*. 2019;19(2):27-33. https://doi.org/10.34257/GJCSTDVOL19IS2PG27
6. Wani N., Mangire G., Kumar A., Solse N., Gaikwad P.S. Legal Document Classification using TF-IDF and KNN. *International Journal of Advanced Research in Science, Communication and Technology*. 2022;2(1):590-595. https://doi.org/10.48175/IJARSCT-7522

7. Nasu Iu., Lanin V.V. Development of Legal Document Classification System Based on Support Vector Machine. *Trudy ISP RAN / Proc. ISP RAS*. 2023;35(2):49-56. https://doi.org/10.15514/ISPRAS2023-35(2)-4
8. Yulianti E., Bhary N., Abdurrohman J., Dwitilas F.W., Nuranti E.Q., Husin H.S. Named entity recognition on Indonesian legal documents: a dataset and study using transformer-based models. *International Journal of Electrical and Computer Engineering (IJECE)*. 2024;14(5):5489-5501. https://doi.org/10.11591/ijece.v14i5.pp5489-5501
9. Leitner E., Rehm G., Moreno-Schneider J. Fine-Grained Named Entity Recognition in Legal Documents. *Lecture Notes in Computer Science*. 2019;11702:272-287. https://doi.org/10.1007/978-3-030-33220-4_20
10. Wadud M.A.H., Mridha M.F., Shin J., Nur K., Saha A.K. Deep-BERT: Transfer Learning for Classifying Multilingual Offensive Texts on Social Media. *Comput Syst Sci Eng*. 2023;44(2):1775–1791. https://doi.org/10.32604/csse.2023.027841
11. Kalyan K.S., Rajasekharan A., Sangeetha S. AMMU: A survey of transformer-based biomedical pretrained language models. *Journal of Biomedical Informatics*. 2022 Feb;126:103982. https://doi.org/10.1016/j.jbi.2021.103982
12. Al-Askary Y.B., Al-Momen S. Enhanced OCR Techniques for Recognizing Mathematical Expressions in Scanned Documents. *Ibn AL-Haitham Journal For Pure and Applied Sciences*. 2025;38(4):295–306. https://doi.org/10.30526/38.4.3640
13. Wang Z., Liu M., Liu K. Utilizing Machine Learning Techniques for Classifying Translated and Non-Translated Corporate Annual Reports. *Applied Artificial Intelligence*. 2024;38(1):e2340393. https://doi.org/10.1080/08839514.2024.2340393
14. Dong M., Gagnon M-A. Unveiling chemical industry secrets: Insights gleaned from scientific literatures that examine internal chemical corporate documents—A scoping review. *PLoS ONE*. 2025;20(1):e0310116. https://doi.org/10.1371/journal.pone.0310116

ИНФОРМАЦИЯ ОБ АВТОРАХ

Иван Владимирович Перлов, РТУ МИРЭА, Москва, Российская Федерация;
ORCID: https://orcid.org/0009-0001-1436-9621; e-mail: perlovivan@yandex.ru
Сергей Александрович Селиванов, канд. техн. наук, доцент, РТУ МИРЭА Институт информационных технологий, Москва, Российская Федерация;
ORCID: https://orcid.org/0000–0002–1229–9025; e-mail: selivanov@inevm.ru
Александр Владимирович Синицын, канд. физ.-мат. наук, РТУ МИРЭА Институт информационных технологий, Москва, Российская Федерация;
ORCID: https://orcid.org/0000-0001-7392-1837; e-mail: a@sinitsyn.info
Шамхал Мехти оглы Шахгусейнов, РТУ МИРЭА, Москва, Российская Федерация;
ORCID: https://orcid.org/0009-0002-0805-0742; e-mail: boss.shamhal@mail.ru

INFORMATION ABOUT THE AUTHORS

Ivan V. Perlov, RTU MIREA, Moscow, Russian Federation;
ORCID: https://orcid.org/0009-0001-1436-9621; e-mail: perlovivan@yandex.ru
Sergey A. Selivanov, Cand. Sci. (Eng.), Associate Professor, RTU MIREA Institute of Information Technology, Moscow, Russian Federation;
ORCID: https://orcid.org/0000–0002–1229–9025; e-mail: selivanov@inevm.ru

Alexander V. Sinitsyn, PhD of Physico-Mathematical Sciences, Associate Professor, RTU MIREA Institute of Information Technology, Moscow, Russian Federation;
ORCID: https://orcid.org/0000-0001-7392-1837; e-mail: a@sinitsyn.info
Shamhal M. Shakhguseynov, RTU MIREA, Moscow, Russian Federation;
ORCID: https://orcid.org/0009-0002-0805-0742; e-mail: boss.shamhal@mail.ru

Поступила / Received 30.05.2025
Принята / Accepted 27.06.2925