# Сопроводительная документация по задаче:

## 09. Автоматизированный алгоритм обезличивания данных.

**Разрабатываемое решение** — Автоматизированный алгоритм обезличивания данных.

**Целевой аудиторией решения** будут сотрудники органов исполнительной власти города Москвы и подведомственных учреждений.

**Конечным пользователем** обезличенных документов станут компании-разработчики решений в сфере искусственного интеллекта, участники эксперимента. Также сервисом смогут пользоваться сотрудники,которым в рамках исполнения служебных обязанностей необходимо передать документы,содержащие персональные данные сторонним организациям.

Так как задача реализации решения является сложной задачей, то в срок до 23.10.2021 г. (1 этап конкурса) были выявлены следующие этапы реализации решения:

1. Анализ требований к разработке решения согласно требований конкурса.
2. Выбор методов и средств разработки решения.
3. Анализ существующих решений в этой области.
4. Проработка способов распознавания и обезличивания ПДн в документах.
5. Разработка теоретической модели прототипа автоматизированного алгоритма обезличивания данных.
6. Проработка тестового варианта обезличивания документов на примере ФИО.
7. Проработка тестового варианта интерфейса веб-сервиса по обезличиванию документов.
8. Разработка тестовой физической реализации модели прототипа автоматизированного алгоритма обезличивания данных в виде приложения на языке программирования Python.
9. Разработка тестовой физической реализации веб-сервиса по обезличиванию документов.
10. Размещение тестовой версии приложения и его веб-интерфейса на хостинг с получением статического адреса.
11. Проверка качества работы реализованных в п. 7,8,9 тестовой версии приложения и его веб-интерфейса, включая возможность загрузки не обезличенных JPG/PDF-файлов и корректность скачивания их обезличенной версии.
12. Размещение исходного кода тестовой версии приложения на GitHub.
13. Разработка предварительной версии сопроводительной документации к решению.

Поскольку по состоянию на 24.10.2021 г. весь перечень работ, заявленный для 1 этапа конкурса, выполнен, то в данный момент команда сосредоточена на выполнении нижеизложенных работ с целью соблюдения сроков разработки решения.

Поскольку текущая версия веб-сервиса находится в стадии предварительного тестирования, то в дальнейшем планируется проведение следующего вида работ:

1. Провести анализ эффективности распознавания ПДн алгоритмом и провести корректировку с целью снизить число его ложных срабатываний. Критерий эффективности — число успешных распознаваний приблизить к 95% случаев. По этой причине параллельно с выполнением указанной задачи планируется проведения работ по изучению фреймворка Deeppavlov с целью обучения модели на поиск ПДн в неструктурированном тексте. Считается, что наиболее успешное решение,реализующее эффективный подход ляжет в основу финальной версии решения.
2. Добавить распознавание других ПДн (таких как дата рождения, паспортные данные, телефон и почта)в соответствии с ст. 3 Федерального закона от 27.07.2006 г. «О персональных данных» №152-ФЗ.
3. Реализовать замену обезличивания данных тестовым способом на обезличивание данных согласно требований Приказа Роскомнадзора от 05.09.2013 N 996 «Об утверждении требований и методов по обезличиванию персональных данных».
4. Разработать прототип, сверстать и внедрить веб-форму сервиса для комфортного взаимодействия его целевой аудитории вместо тестовой формы, расположенной сейчас по веб-адресу: http://193.32.219.30:5000/upload.
5. Провести анализ эффективности алгоритма на предмет его оптимизации с целью снижения времени на обработку файлов.
6. В соотвествии с изменениями доработать сопроводительную документацию к решению.

Гипотеза:

Проанализировав визуально базовый объем документов к данной задаче и имея экпертный опыт работы (создание, исполнение документов) с электронным документооборотом в государственной организации пришли к выводу: что документы в основном состоят из таблиц и текста, печатей, оттисков и др артефактов документооборота в организации. Текс размещен в таблицах, в шапке документа, в теле и в подвале документа, а также в таблицах без напечатанных границ.

При анализе поставленной проблемы деперсонификации управленческих, распорядительных и др. документов, решение задачи видится нам в решении двух подзадач:

1. Распознавание текста документа.

2. Лингвистический анализ получено текста с определением персональных данных.

Использовав анализ, пришли к возможности решения данных двух подзадач имеющимися средствами.

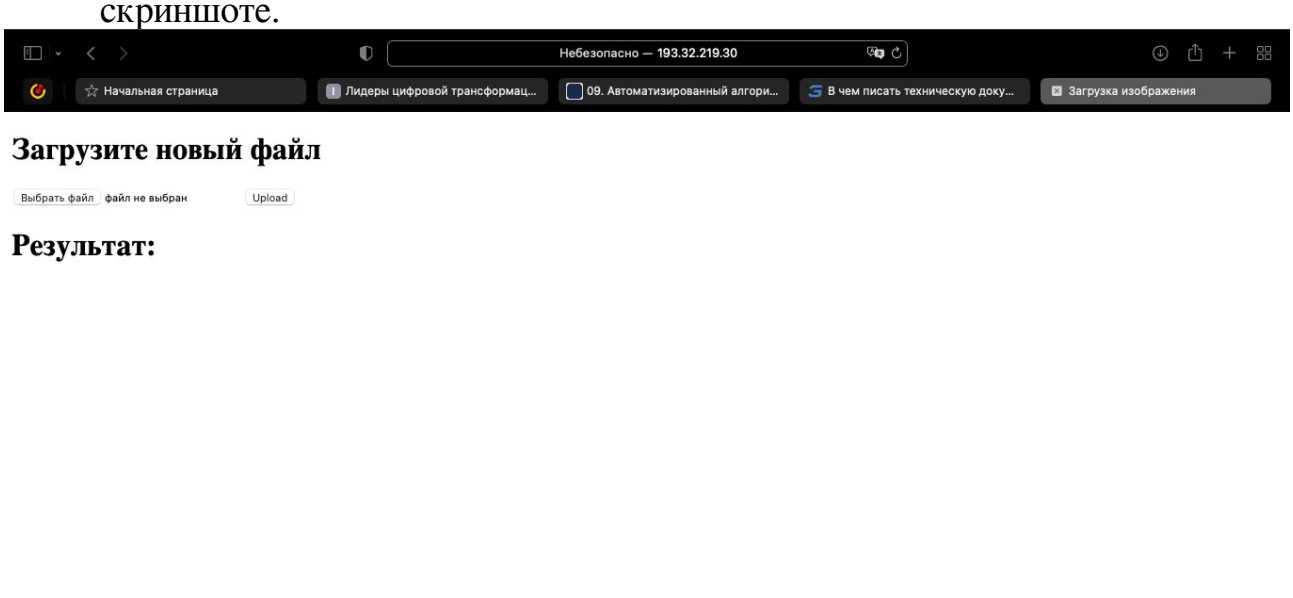
Распознавание текста возможно при использовании библиотеки с открытым исходным кодом Tesseract, EasyOCR (https://github.com/jaidedai/easyocr), сервис визуального распознавания Yandex.Cloud, ABBYY Cloud OCR SDK(публичный API распознавания в облаке Windows Azure).

Лингвистический анализ полученного текста можно проводить с помощью DaData, Pullenti, Abbyy Infoextractor, Dictum, Eureka, Promt, RCO, AOT, Ahunter, и открытыми решениями Natasha (https://github.com/natasha/natasha) и DeepPavlov (https://github.com/deepmipt/DeepPavlov).

В данной работе используется Tesseract (запасной вариант EasyOCR) как слой распознавания, в лингвистическом анализе используется библиотеки Natasha, DeepPavlov (нейроная модель NER_RUS, NER_RUS_BIRT) и синтез их совместной попарной реализации с оценкой результатов деперсонификации.

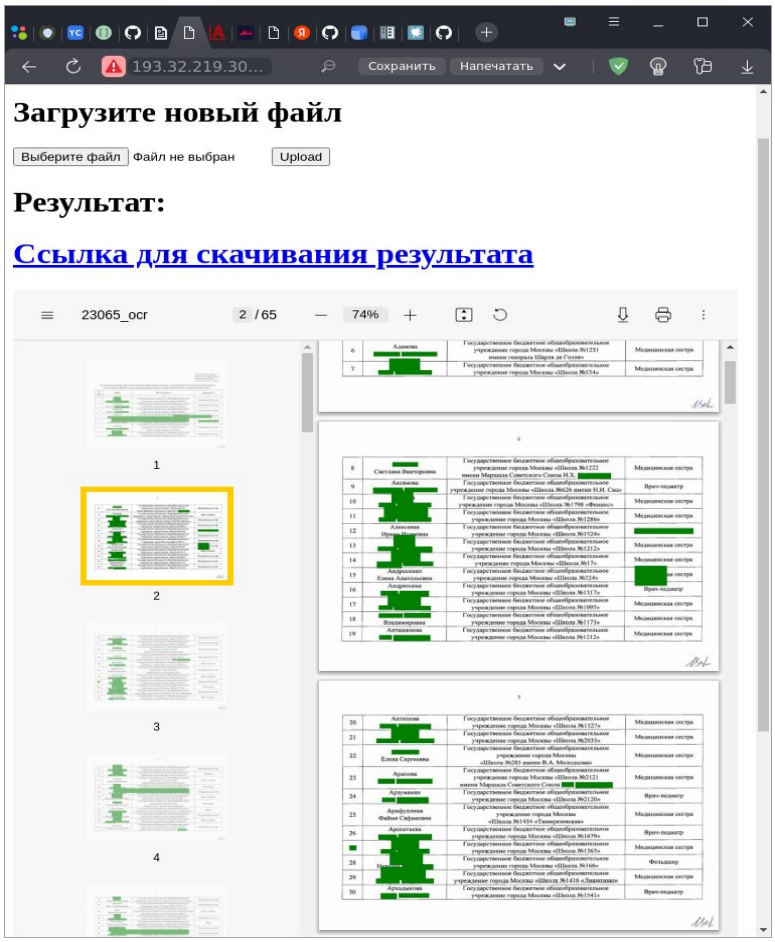
Алгоритм работы:

1. Пользователь заходит на ресурс обезличивания по веб-адресу: http://193.32.219.30:5000/upload и попадает в интерфейс сервиса по обезличиванию. Скриншот интерфейса пользователя изображен на скриншоте.




2. Далее пользователь нажав на кнопку «Выбрать файл» указывает путь к файлу, который нужно обезличить.
3. После выбора файла пользователю необходимо нажать кнопку «Upload» и произойдет выгрузка указанного пользователем файла на сервер распознавания и обезличивания. Поддерживаются файлы PDF/JPG/JPEG.
4. После этого посредством функции def conv_pdf(file_pdf): происходит конвертация PDF в массив файлов JPG. В своей работе модуль использует библиотеки fitz, PyMuPDF, а также их зависимости. Указанные библиотеки используют открытые лицензии. Если загружаемый на ресурс файл является JPG документом, то данный шаг пропускается.
5. После конвертации происходит передача одного изображения на распознавание текста из него. В своей работе модуль использует библиотеку Tesseract. Библиотека используют открытые лицензии.
6. Результат распознавания передается на вход библиотеки Natasha (реализация на тестовом сервере) или DeepPavlov (при наличии на сервере GPU). Указанные библиотеки используют открытые лицензии.
7. Далее затираем ФИО по известным координатам блоков. И, используя библиотеку Pillow мы затираем зеленым цветом ФИО. Закрашивание реализовано в тестовом режиме. Указанные библиотеки используют открытые лицензии. Выход модуля — изображение с обезличенными данными.

8. Если загруженный пользователем файл на ресурс являлся PDF-документом, то происходит сборка полученных в п.7 изображений в результирующий PDF-файл. Если пользователь загружал для обезличивания изображение(jpg/jpeg), то данный шаг пропускается.
9. Результат обезличивания отображается пользователю на ресурсе. Также пользователю доступна ссылка для скачивания обезличенного файла.




Как видно из описанного выше алгоритма, его работа строится на использовании методов:

1. Распознавания текста Tesseract OCR
2. NLP проекта Natasha.

В качестве примера работы был выбран файл documents_files_3650_PR-4-21-OFI.pdf. Файл состоит из 9 страниц. Его время обезличивания с момента загрузки до получения ссылки для скачивания составило 2 минуты. Указанный файл был предоставлен рамках выполнения задания.

Параметры тестового стенда:

**Ресурсы: Yandex.Cloud(Compute Cloud)**

Платформа: Intel Ice Lake
Гарантированная доля vCPU 100%
vCPU: 6
RAM: 18 ГБ
Объем дискового пространства: 20 ГБ
Прерываемая: да
GPU: нет

Работа алгоритма была проверена на всем наборе предоставленных в рамках конкурса документов. Набор содержал более 1300 открытых служебных документов.