Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика и управление

КАФЕДРА Системы обработки информации и управления (ИУ5)

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

НА ТЕМУ:

Студент ИУ5-61Б _____ В. С. Ноздрова
(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель _____ _____
(Подпись, дата) (И.О.Фамилия)

Консультант _____ _____
(Подпись, дата) (И.О.Фамилия)

2022 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ

Заведующий кафедрой _______
(Индекс)

_____ _____
(И.О.Фамилия)

« _____ » _____ 20 ____ г.

З А Д А Н И Е
на выполнение научно-исследовательской работы

по теме _____Оптическое распознавание текстов (OCR) с помощью методов_____

_____машинного обучения_____

_____

Студент группы __ИУ5-61Б__

_____Ноздрова Валентина Сергеевна_____
(Фамилия, имя, отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.)

_____

Источник тематики (кафедра, предприятие, НИР) _____

График выполнения НИР: 25% к ___ нед., 50% к ___ нед., 75% к __ нед., 100% к ___ нед.

Техническое задание _____

_____

_____

_____

_____

Оформление научно-исследовательской работы:

Расчетно-пояснительная записка на ____ листах формата А4.
Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

_____

_____

_____

_____

Дата выдачи задания « ___ » _____ 2022 г.

Руководитель НИР _____ _____
(Подпись, дата) (И.О.Фамилия)

Студент _____ **В. С. Ноздрова**
(Подпись, дата) (И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Оглавление

Введение

Оптическое распознавание символов (OCR) – технология компьютерного зрения, применяемая в таких областях, как автоматизация систем учета в бизнесе, распознавание спама в социальных сетях и почтовых сервисах, обеспечение безопасности дорожного движения, машинный перевод. В области OCR активно используются алгоритмы машинного обучения и нейронные сети. Целью данной работы является изучение принципов работы OCR-инструментов Tesseract и OCRopus.

1. Постановка задачи

В ходе выполнения работы были поставлены следующие задачи:

— исследование принципов работы Tesseract и OCRopus;

— исследование и визуализация данных;

— определение метрик для оценки моделей;

— сравнение качества распознавания на основе выбранных метрик.

2. Исследование принципов работы моделей Tesseract и OCRopus

2.1. Tesseract

Tesseract – OCR-движок, в настоящее время принадлежащий компании Google. Обработка изображений Tesseract состоит из нескольких этапов. Первым шагом является анализ связанных компонентов, в ходе которого сохраняются контуры компонентов, из которых далее собираются фрагменты, а из них, в свою очередь, текстовые строки. Строки и текстовые регионы обрабатываются по-разному в зависимости от ширины шрифта: моноширинный текст сразу разбивается на отдельные символьные ячейки, пропорциональный текст разбивается на слова с помощью четких и нечетких пробелов.

Распознавание проходит в два этапа. При первом проходе предпринимается попытка распознать каждое слово по очереди. Каждое успешно распознанное слово передается адаптивному классификатору в качестве обучающих данных, таким образом текст, расположенный ниже на странице, распознается более точно. Далее выполняется второй проход по странице, во время которого распознаются слова, которые не были достаточно хорошо распознаны в первый раз. В заключительной фазе распознаются области с нечеткими пробелами и проверяются альтернативные гипотезы высоты знаков, чтобы распознавать текст из малых заглавных букв.

2.1.1. Поиск строк

Алгоритм поиска строк разработан таким образом, что изогнутая страница может быть распознана без необходимости выравнивания, что помогает сохранить качество изображения. Предполагая, что анализ макета страницы уже определил регионы с примерно одинаковым размером текста, простой фильтр по высоте убирает буквицы и вертикально соприкасающиеся символы. Средняя высота приблизительно соответствует размеру текста в регионе, поэтому можно отфильтровать фрагменты меньше некоторой доли средней высоты, которые, скорее всего, являются пунктуацией, диактрическими знаками или шумом.

Отфильтрованные фрагменты с большой вероятностью ложатся на непересекающиеся параллельные косые линии. Сортировка и обработка фрагментов по горизонтальной координате позволяет присвоить фрагменты

уникальной строке. Отслеживание наклона по всей странице позволяет значительно уменьшить риск присвоения фрагментов неправильной строке в случае перекоса страницы. После распределения фрагментов по строкам базовая линия текста подбирается методом наименьших средних квадратов, и отфильтрованные фрагменты возвращаются в соответствующие строки.

На последнем этапе поиска строк происходит слияние фрагментов, которые перекрываются по горизонтали как минимум наполовину, таким образом диактрические знаки расставляются над правильными базами, соединяются части разделенных символов.

Начиная с версии 4.0 Tesseract использует для поиска строк рекуррентную нейронную сеть с долгой краткосрочной памятью.

2.1.2. Подбор базовой линии

После того, как строки найдены, происходит более точный подбор базовой линии с помощью квадратичного сплайна. Это позволяет Tesseract обрабатывать страницы с изогнутым текстом, что характерно для сканированных документов.

Базовые линии устанавливаются путем разбиения фрагментов на группы с примерно одинаковым смещением относительно исходной базовой линии. Квадратичный сплайн подбирается к самой многочисленной группе методом наименьших квадратов.

2.1.3. Обнаружение моноширинного и пропорционального текста

Tesseract проверяет текстовые строки на соответствие моноширинному шрифту. Там, где Tesseract определяет такой тип шрифта, слова разбиваются на символы равной ширины. В случае пропорционального текста пробелы между рамками символов могут быть различной ширины. Tesseract решает проблему измерением пробела в ограниченном вертикальном диапазоне между базовой линией и верхней линией строчных букв. Пробелы, близкие к пороговому значению, признаются нечеткими, таким образом окончательное решение принимается после распознавания слов.

2.1.4. Распознавание слов

Частью процесса распознавания любой OCR-системы является определение правильного разделения слова на символы. Полученные в результате поиска строк данные классифицируются в первую очередь. Остальная часть алгоритма распознавания слов применяется к тексту без фиксированной ширины символов.

2.1.5. Разделение слитных символов

В случае, когда результат распознавания слова неудовлетворителен, Tesseract пытается улучшить результат путем разделения фрагментов с низкой уверенностью в распознавании. К контурам фрагментов применяется многоугольная аппроксимация, и из вершин вогнутых элементов выбираются предполагаемые точки разделения. Затем варианты разделения рассматриваются по очереди. Разделение, которое не улучшает результат, отменяется, но может использоваться ассоциатором.

2.1.6. Соединение разделенных символов

Если варианты разделения фрагментов не дают улучшения результата, слово передается на обработку ассоциатору. Ассоциатор производит поиск возможных комбинаций соединения фрагментов.

2.1.7. Классификация символов

Классификация символов проходит в два этапа. На первом этапе выбираются классы символов, к которым может относиться распознанный символ. Для каждого отдельного сегмента полигональной аппроксимации контура вычисляется вектор классов, которым он может соответствовать. Векторы сегментов суммируются, и классы с наибольшим весом выбираются для следующего шага. На втором шаге сегменты сравниваются с прототипами классов, которым они могут соответствовать, и вычисляется расстояние между соответствующими векторами.

2.1.8. Лингвистический анализ

Каждый раз, когда модуль распознавания слов рассматривает новую сегментацию, лингвистический модуль выбирает наиболее подходящее слово из следующих категорий: самое часто встречающееся слово, лучшее словарное слово, лучшая числовая строка, лучшее слово в верхнем регистре, лучшее слово в нижнем регистре (с возможной первой заглавной буквой), лучший выбор классификатора. Окончательный выбор делается на основе наименьшего расстояния между распознанным сегментом и словом из каждой категории, причем каждое расстояние умножается на определенную для каждой категории константу.

2.2. OCRopus

OCRopus является набором инструментов для анализа документов и OCR и состоит из отдельных модулей для бинаризации, сегментации и т.д.

2.2.1. Анализ макета страницы

Для выделения текстовых регионов на странице используются пять различных алгоритмов. Морфологический сегментор выделяет отдельные текстовые блоки, затем сегментор на основе проекции исследует профили горизонтальной проекции для разделения текстовых строк на символы. Алгоритм рекурсивного разреза по осям x и y исследует страницу на наличие столбцов и строк белых пикселей и по ним «разрезает» страницу на блоки, после чего алгоритм применяется рекурсивно к каждому блоку. Метод Вороного строит диаграмму Вороного по центрам компонент связности, после чего объединяются те области, расстояние между центрами которых меньше заданного порога. Алгоритм RAST определяет максимальные или минимальные белые прямоугольники (покрытия) в зависимости от величины связных компонентов, после чего покрытия объединяются таким образом, чтобы в них попадало определенное количество компонентов. После того, как найдены все разделители, компоненты классифицируются и определяется порядок их чтения.

2.2.2. Распознавание текста

Для распознавания текста OCRopus использует рекуррентную нейронную сеть с долгой краткосрочной памятью. Это сильно нелинейная рекуррентная сеть с «фильтрами» внутри ячейки, позволяющими пропускать информацию на основании некоторых условий. OCRopus использует одномерную двунаправленную архитектуру LSTM для доступа к контексту в прямом и обратном направлениях. Оба слоя затем подключаются к одному выходному слою.

3. Исследование и визуализация данных

Исходный датасет состоит из 425 текстовых строк в виде изображений формата .png и расшифровок к ним.

Распределение количества различных символов (без учета пробелов) в данных приведено на рис. 1.

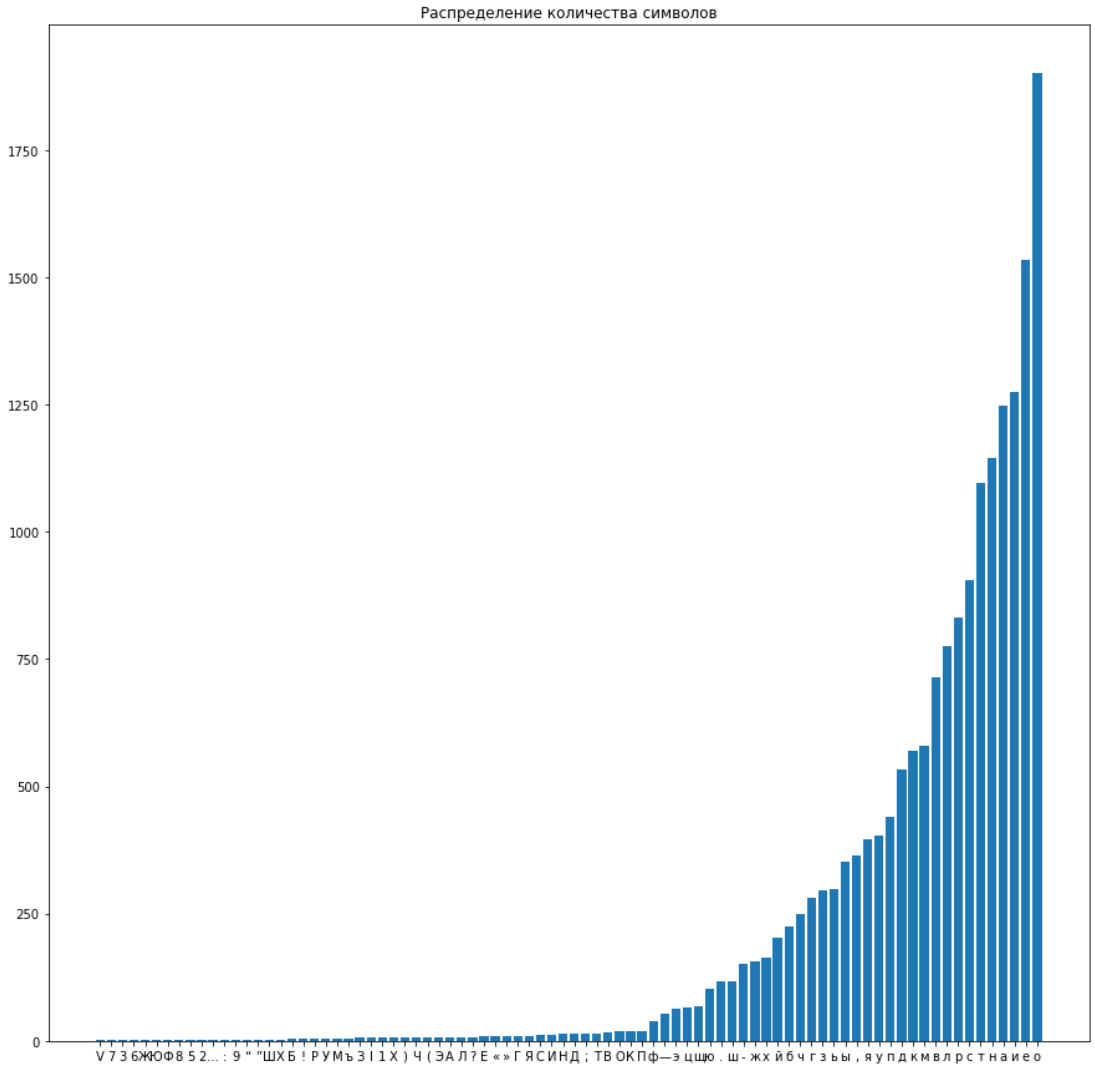

Рис.1. Диаграмма распределения символов

Распределение длины текстовых строк (расшифровок) с учетом пробелов приведено на рис. 2.

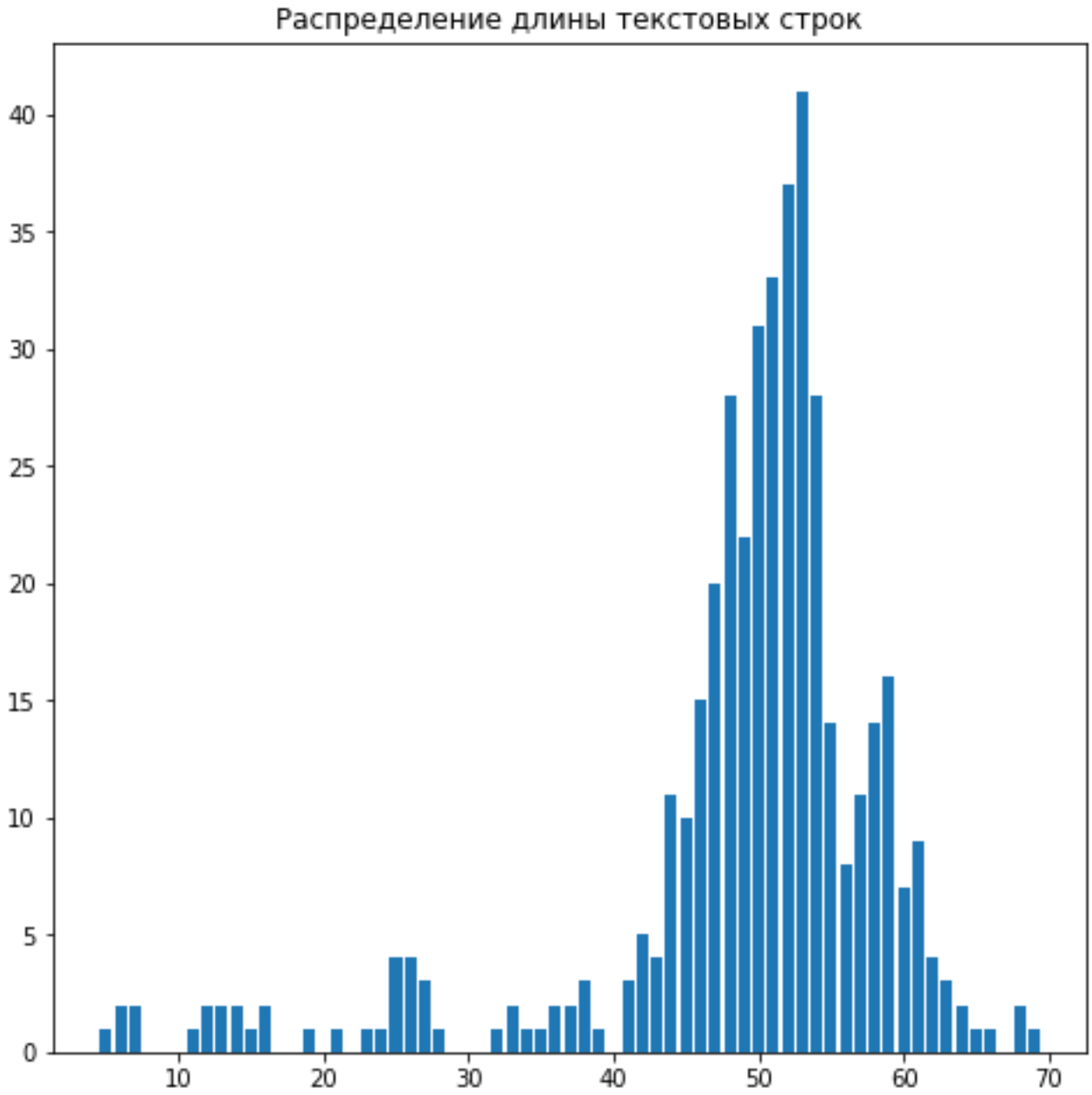

Рис. 2. Диаграмма распределения длины текстовых строк

4. Определение метрик для оценки моделей

В качестве метрики для оценки качества распознавания будем использовать метрики CER (англ. Character Error Rate) и WER (англ. Word Error Rate). Метрика CER показывает частоту неверно распознанных символов.

$$CER = (S + D + I)/N$$

где S - количество замен символов,

D - количество удалений символов,

I - количество вставок символов, которые необходимы, чтобы получить из предсказанной строки истинную строку;

N - общее количество символов.

WER является аналогичной метрикой для оценки частоты неверно распознанных слов.

5. Сравнение качества распознавания на основе выбранных метрик

Полученные для двух моделей значения CER и WER приведены на рис.3 и рис. 4.

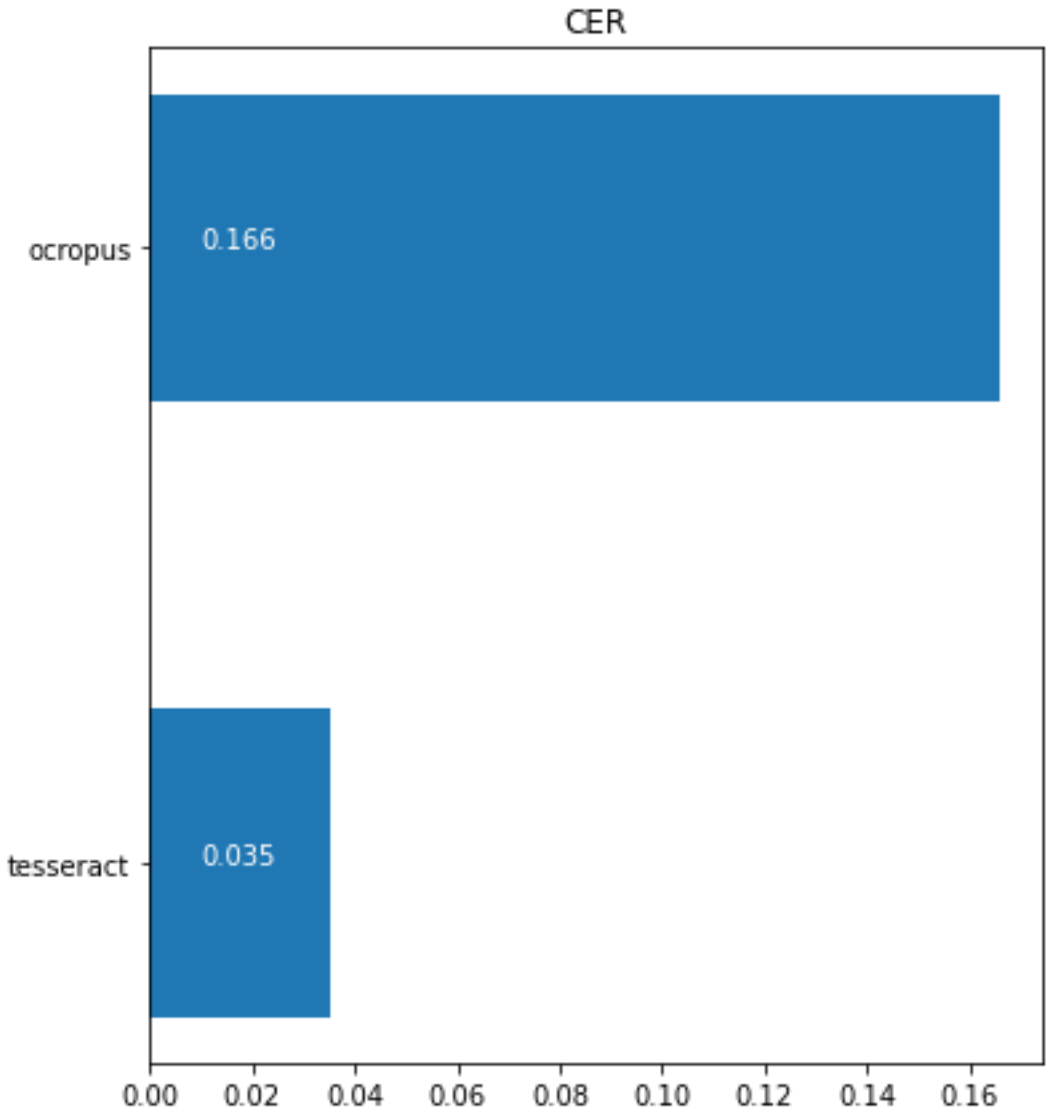

Рис. 3. Сравнение моделей на основании метрики CER

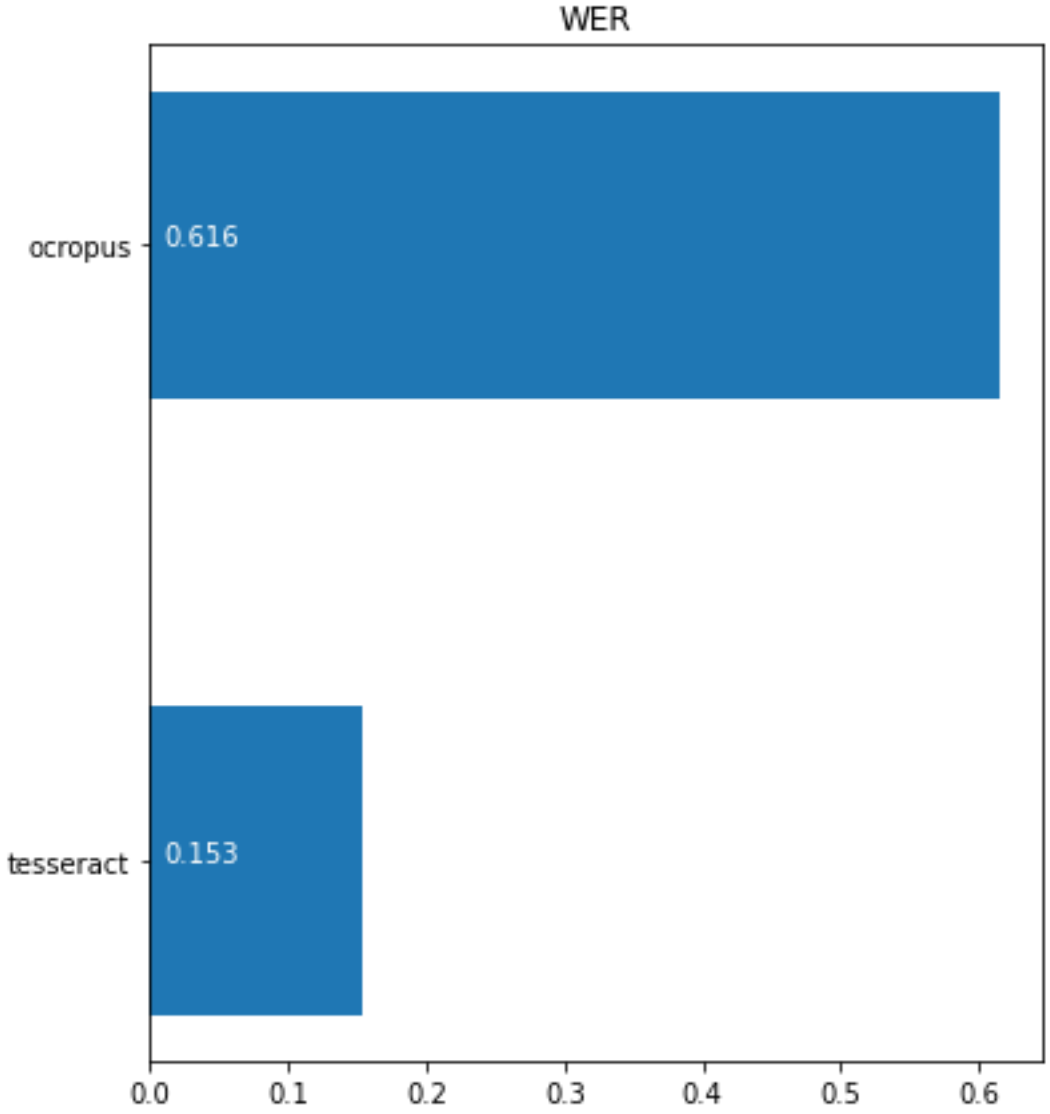

Рис. 4. Сравнение моделей на основании метрики CER

Из графиков видно, что качество распознавания OCRopus в намного ниже, что может быть объяснено малым количеством использованных при обучении данных.

Заключение

В результате работы были рассмотрены принципы работы OCR инструментов Tesseract и OCRopus. При сравнении качества распознавания текстов на русском языке модель Tesseract показала более хороший результат, чем OCRopus. Тем не менее, ошибки в распознавании присутствуют у обоих моделей. Результаты могут быть улучшены с помощью дообучения моделей и более качественной предобработки входных изображений.

Использованная литература

1. R. Smith, «An Overview of the Tesseract OCR Engine» Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), 2007
2. Ul-Hasan, Adnan & Breuel, Thomas. (2013). Can we build language-independent OCR using LSTM networks?
3. Winder, Amy & Andersen, Tim & Barney Smith, Elisa. (2011). Extending Page Segmentation Algorithms for Mixed-Layout Document Processing.
4. Blando, Luis & Kanai, J. & Nartker, Thomas. (1995). Prediction of OCR accuracy using simple image features.