УДК 004.9, 519.6

РАСПОЗНАВАНИЕ ИСТОРИЧЕСКИХ РУКОПИСНЫХ ДОКУМЕНТОВ С ПРИМЕНЕНИЕМ МЕТОДОВ ГЛУБОКОГО МАШИННОГО ОБУЧЕНИЯ

Мамедов Тимур Русланович[1], Ершов Николай Михайлович[2]

[1]Студент;
Московский государственный университет им. М. В. Ломоносова;
Россия, 119991, г. Москва, ул. Ленинские горы, д. 1;
e-mail: mtimur_2011@mail.ru.

[2]Старший научный сотрудник;
Московский государственный университет им. М. В. Ломоносова;
Россия, 119991, г. Москва, ул. Ленинские горы, д. 1;
e-mail: ershovnm@gmail.ru.

Настоящая работа посвящена разработке и анализу нейросетевых подходов и методов к решению задачи распознавания рукописных документов. Для решения данной задачи в работе предлагается использовать модели глубоких нейронных сетей. Рассматриваются вопросы конфигурации и обучения рассматриваемых моделей, также описываются и анализируются возможные их усовершенствования. Приводятся результаты численного исследования всех предложенных подходов и сравнение их эффективности в решении поставленной задачи.

Ключевые слова: распознавание образов, глубокое обучение, рекуррентные нейронные сети, трансформеры.

Для цитирования:

Мамедов Т. Р., Ершов Н. М. Распознавание исторических рукописных документов с применением методов глубокого машинного обучения // Системный анализ в науке и образовании: сетевое научное издание. 2024. № 3. С. 46-53. EDN: WZSVZO. URL: https://sanse.ru/index.php/sanse/article/view/625.

RECOGNIZING HISTORICAL HANDWRITTEN DOCUMENTS USING DEEP MACHINE LEARNING METHODS

Mamedov Timur R.[1], Ershov Nikolay M.[2]

[1]Student;
Lomonosov Moscow State University;
1 Leninskiye Gory, Moscow, 119991, Russia;
e-mail: mtimur_2011@mail.ru.

[1]Senior Researcher;
Lomonosov Moscow State University;
1 Leninskiye Gory, Moscow, 119991, Russia;
e-mail: ershovnm@gmail.ru.

The paper is devoted to the development and analysis of neural network approaches and methods for solving the problem of handwritten document recognition. To solve this problem, we proposes to use deep neural network models. The issues of configuration and training of the suggested models are considered, and their possible improvements are described and analyzed. The results of a numerical study of all proposed approaches and a comparison of their effectiveness in solving the problem are presented.

Keywords: pattern recognition, deep learning, recurrent neural networks, transformers.

For citation:
Mamedov T. R., Ershov N. M. Recognizing historical handwritten documents using deep machine learning methods. *System analysis in science and education*, 2024;(3):46-53 (in Russ). EDN: WZSVZO. Available from: https://sanse.ru/index.php/sanse/article/view/625

Введение

Задача распознавания рукописного текста (*handwritten text recognition* или *HTR*) является классической задачей распознавания образов. Данная проблема возникает во многих областях жизнедеятельности человека, например, большая часть данных о пациентах в больницах велась, и до сих пор ведётся, на бумажных носителях, но такой подход совершенно непрактичен, поэтому внедрение систем по распознаванию рукописного текста позволило бы перевести уже имеющиеся данные о посетителях в электронный формат. Другая актуальная область возможного применения систем распознавания рукописных текстов – работа с архивными документами, которая до сих пор ведётся в ручном режиме, когда для поиска нужной информации требуется обращаться к оригиналам документов или их отсканированным копиям. Использование системы автоматического распознавания текста позволило бы перевести весь накопленный материал в текстовое представление, для работы с которым существует немало удобных утилит и приложений.

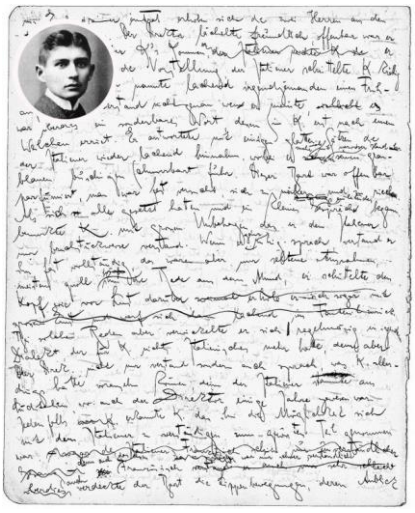
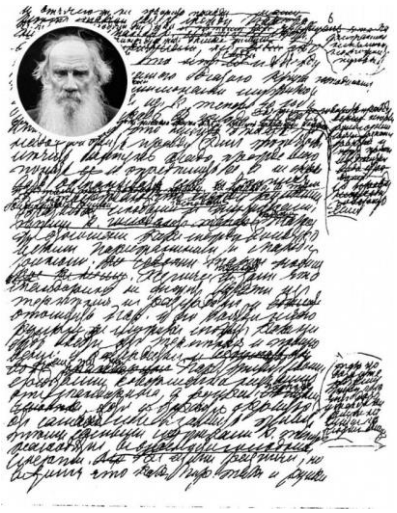
а)  б)  в) 

Рис. 1. Примеры рукописных документов: а) Ф. Кафка, б) А. С. Пушкин, в) Л. Н. Толстой

Несмотря на то что рассматриваемая проблема актуальна на протяжении многих десятков лет и существует множество подходов и алгоритмов её решения, отнести эту задачу к разряду решённых всё ещё нельзя (в отличие, например, от задачи распознавания печатных текстов, где прогресс является существенно более значительным). Сложности с распознаванием рукописного текста связаны с несколькими факторами (см. рис. 1, иллюстрирующий данные факторы), такими как:

- бесчисленное разнообразие почерков различных людей;
- почерк любого человека непостоянен и может меняться в рамках одного документа (и даже одного слова);
- низкое качество изображений исторических рукописных документов;
- текст может быть расположен в произвольном месте документа: в отличие от «печатных» текстов с более-менее жёстко заданной структурой, рукописные блоки текста могут располагаться в любом месте документа, в любой ориентации и обладать любой формой;
- наличие наклона почерка осложняет процесс сегментации или разделения текста на отдельные самостоятельные части — предложения, слова, символы.

1. Существующие подходы к распознаванию рукописных текстов

Методы распознавания рукописных текстов можно разделить на два больших класса: онлайн методы (распознавание производится в процессе написания текста) и офлайн методы (распознавание выполняется по полному изображению текстового документа). Так как предметной областью настоящей работы являются именно исторические тексты, дальнейшее обсуждение методов решения будет опираться на методы офлайн распознавания.

Исторически, первые подходы к решению задачи распознавания рукописного текста основывались на математических моделях и статистических алгоритмах. Так, одним из первых способов было сопоставление символов текста с базой шаблонов по определённому критерию (минимуму различных точек). Несмотря на то что такой подход был прост в реализации, он требовал составления огромного числа примеров и обладал довольно низким качеством.

Следующим этапом стало применение признаковых классификаторов. Из предварительно обработанного изображения автоматически или в ручном режиме выделяются различные признаки, которые затем используются классическими методами машинного обучения (скрытые марковские модели, опорные векторы). Однако набор применяемых признаков определяется индивидуально для каждого языка или даже документа, что делает такой подход абсолютно не масштабируемым.

Наиболее успешным оказался метод, основанный на глубоком машинном обучении (нейронных сетях), поскольку отбор необходимых признаков теперь осуществляет сама модель и отсутствует необходимость в составлении огромной базы примеров и шаблонов, хотя, конечно, для обучения нейронной сети требуется сбор тренировочных наборов, однако число таких примеров зачастую оказывается меньше необходимого для работы шаблонных классификаторов, и гарантирует более высокое качество.

Современные нейросетевые подходы к задаче распознавания рукописных текстов можно разделить на три группы:

- свёрточно-рекуррентные нейронные сети, *CRNN* [1]: с помощью свёрточных нейронных сетей выделяются признаки, которые затем обрабатываются рекуррентной частью;
- многомерные рекуррентные нейронные сети, *MDRNN* [2]: подобные архитектуры используют иерархию многомерных рекуррентных слоёв, которые обрабатывают изображение в различных направлениях;
- энкодер-декодер модели [3], например, модели трансформеров.

Отметим, что на сегодняшний день лучшие результаты в области распознавания рукописных текстов достигаются именно с использованием трансформеров, например, с архитектурой *TrOCR* [4].

2. Подготовка обучающего набора данных

В качестве обучающей, валидационной и тестовой выборок был выбран обучающий набор *IAM* [5], содержащий 13353 изображений современного английского языка от 657 авторов (рис. 2). В наборе имеется готовое разбиение на выборки, что позволяет проводить обучение различных моделей и проводить их сравнение без учёта фактора возможного дисбаланса данных в обучении и валидации. Представлены изображения текста в трёх различных вариациях: изображения каждого слова, строки или целой страницы текста.

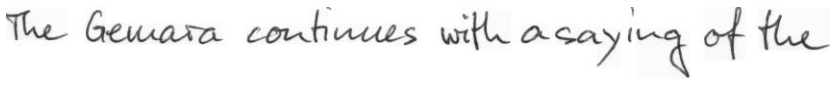


Рис. 2. Пример изображения из обучающего набора IAM

При использовании предобученных моделей, мы вынуждены применять алгоритмы предобработки, с которыми первоначально обучалась модель, однако при реализации собственных подходов может быть выбран произвольный алгоритм предобработки данных. Целью этапа предобработки является повышение качества изображения за счёт фильтрации, шумоподавления и других подходов, имеющих своей целью повысить качество изображения. В настоящей работе был предложен и реали-

зован механизм предобработки изображений на языке *Python* с использованием библиотеки *OpenCV*, включающий следующие шаги (см. рис. 3):

1. размытие изображения с ядром Гаусса: используется для удаления мелкого шума;
2. бинаризация: используется для отделения фона от интересующего текста, в зависимости от передаваемого на вход изображения может быть выбран алгоритм глобальной бинаризации (метод Оцу) либо локальной (с ядром Гаусса);
3. морфологические преобразования: полученное на предыдущем этапе изображение складывается с морфологическим градиентом, что позволяет избавиться от шума среднего размера, затем применяется морфологическое закрытие для удаления шума уже крупного размера, и морфологическое размытие для повышения чёткости контуров;
4. удаление наклона символов на изображении.

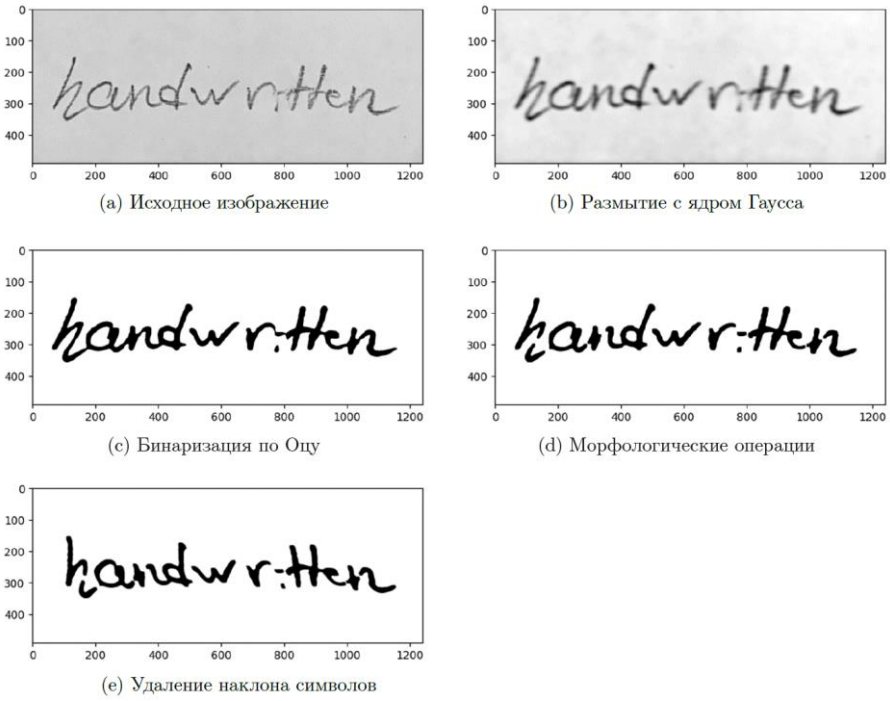


Рис. 3. Пример применения алгоритма предобработки изображений

3. Реализация и исследование нейросетевых моделей

В процессе выполнения настоящего исследования были проведены настройка, реализация и исследование четырёх нейросетевых моделей, предназначенных для обработки исторических рукописных текстов.

Модель SegLineHTR

В качестве первого подхода была реализована модель *SegLineHTR* (рис. 4), способная распознавать изображения, содержащие не более одного слова. Данное решение обусловлено тем, что подобная нейронная сеть содержит малое число параметров, при этом данных для обучения наоборот гораздо больше по сравнению с моделями для распознавания изображений с целыми строками текста (тот же набор *IAM* содержит порядка 130 тысяч подобных изображений).

Для каждой области изображения (несколько расположенных рядом столбцов) строится вектор признаков размерности 256, своего рода эмбеддинг этой области, с помощью композиции свёрточных слоев. Последовательность векторов-признаков обрабатывается двунаправленной двухслойной сетью *LSTM*, которая возвращает результаты работы всех ячеек (так как используем двунаправленный вариант, то размерность векторов признаков возрастает до 512). Для дальнейшей работы используется только первый выход слоя, после чего с помощью обучаемой матрицы перехода происходит отобра-

жение полученных векторов признаков в пространство размерности 80: 79 символов распознаваемого алфавита и один служебный символ, необходимый для корректной работы функции потерь *CTCLoss*, посредством которой и обучается нейронная сеть. Во время тестирования полученная матрица декодируется с помощью лучевого поиска с числом лучей равным 15. Для тестирования полученной модели на данных из *IAM* датасета с изображениями строк, необходим механизм сегментации на изображения, содержащие отдельные слова. Подобный алгоритм был реализован в данной работе с помощью библиотеки *OpenCV*.

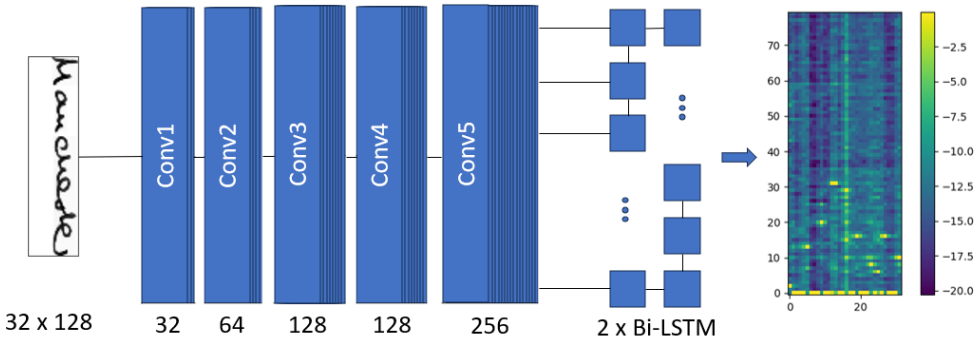


Рис. 4. Архитектура модели SegLineHTR

Модель *LineHTR*

Рассмотренный алгоритм сегментации для задачи распознавания рукописного текста не является устойчивым, поскольку во многих случаях расстояние между символами одного слова не сильно отличаются от расстояния между соседними словами. В связи с этим была реализована модель *LineHTR*, которая принимает на вход изображение, содержащее несколько слов (строку текста). Архитектура модели не отличается кардинально от предыдущей модели, а получается её модификацией – добавлением нормализации и увеличением числа слоёв. Так, количество свёрточных слоёв увеличено до семи, после каждого слоя применяется пакетная нормализация [6]. Размерность вектора признаков для области изображения возросло до 512, при этом число рекуррентных слоёв увеличено до трёх, и после них добавлена сеть прямого распространения с размерностью внутреннего состояния равной 2048. Описанная комбинация гиперпараметров, как будет показано дальше, позволила добиться наиболее высокого качества.

Модель LineHTR+T5

Как будет показано ниже, модель *LineHTR* справляется с разделением текста на изображении на отдельные слова, но нередко допускает грамматические и орфографические ошибки. Для исправления подобных ошибок был дообучен трансформер *T5* от *Google* [7]. В качестве тренировочного набора данных для дообучения модели выступал параллельный корпус, содержащий текст, полученный в ходе работы модели *LineHTR*, и оригинальный текст, содержащийся на изображении. Дообучение проводилось с помощью библиотеки *Transformers*.

Модель *TrOCR*

Последней рассмотренной в данной работе моделью является модификация наилучшего на сегодняшний день решения – модели *TrOCR* [4]. Механизм предобработки для этой модели был изменён таким образом, чтобы поступающие на вход изображения приводились к размеру 1536 пикселей по ширине и 96 пикселей по высоте, что не изменяет общего числа пикселей изображения, но лучше соответствует используемому набору данных. Так как модель обучалась на данных квадратной формы, патчи предобработанного новым подходом изображения требовалось переставить специальным образом, чтобы снова сформировать матрицу пикселей размером 384 на 384. Подобный «трюк» не требует переобучения всех слоёв сети, поскольку перед попаданием в кодировщик для каждого патча уже будет получен вектор признаков, и вся их совокупность будет представлена в виде последовательности. Поэтому для корректной работы модели сначала была обучена матрица абсолютных позицион-

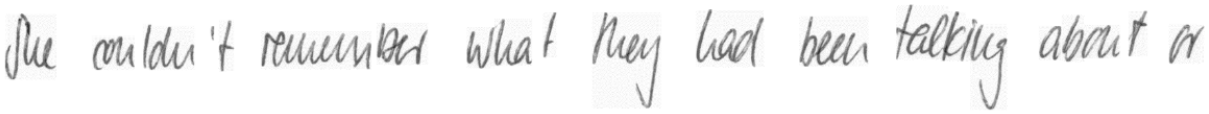
ных эмбеддингов (при «замороженных» остальных 25 слоях), которые добавляются к векторам признаков патчей, и уже на второй итерации все веса сети были «размерожены» и дообучены.

4. Результаты численных экспериментов

Два примера работы всех описанных выше нейросетевых моделей приведены на рисунках 5 и 6. Красным цветом выделены некорректно распознанные слова. Видно, что лучшее качество распознавание показывает модель *TrOCR*.

Модель	Декодированный текст
Target	So he put up for the night at The Admiral's Head, that
SegLineHTR	So he put up for the sightat The Admiral's Head, that
LineHTR	So he put up for the night at The Amialt Nead, that
LineHTR+T5	So he put up for the night at The Amialt Nead, that
TrOCR	So he put up for the night at The Admiral's Head, that

Рис. 5. Первый пример работы описанных моделей



Модель	Декодированный текст
Target	She couldn't remember what they had been talking about or
SegLineHTR	the couldner if remembdr Wha they had been talthy about a
LineHTR	She couldn't remember what they had been talhnyg abart it
LineHTR+T5	She couldn't remember what they had been talking about it
TrOCR	She couldn't remember what they had been talking about or

Рис. 6. Второй пример работы описанных моделей

Для формальной оценки качества моделей применялись две стандартные метрики: *CER* (*character error rate*) и *WER* (*word error rate*). Первая метрика численно равна расстоянию Левенштейна между истинной и распознанной моделями текстовым строками, нормированному на длину исходной строки. Вторая метрика является аналогичной первой, но вычисляется на уровне слов, а не отдельных символов.

На рис. 7 показаны распределения метрики *CER* для четырёх рассмотренных моделей. Как видно из представленных графиков, лучшее качество достигается при использовании дообученной модели *TrOCR*. Наибольшее количество ошибок совершает архитектура *SegLineHTR*, однако, связано это в большей степени с некорректной сегментацией изображения на отдельные «слова», чем собственно с распознаванием символов. Более отчётливо проявление данной проблемы можно заметить на гистограмме критерия *WER*, показанной на рис. 8, где показатели качества модели *SegLineHTR* гораздо хуже остальных моделей. Кроме того, следует отметить, что хотя применение трансформера *T5* в

среднем и улучшает результат работы модели *LineHTR*, но во многих случаях «исправление» ошибок трансформером приводит к корректным с точки зрения грамматики фразам, которые, однако, не соответствуют тексту на оригинальном изображении.

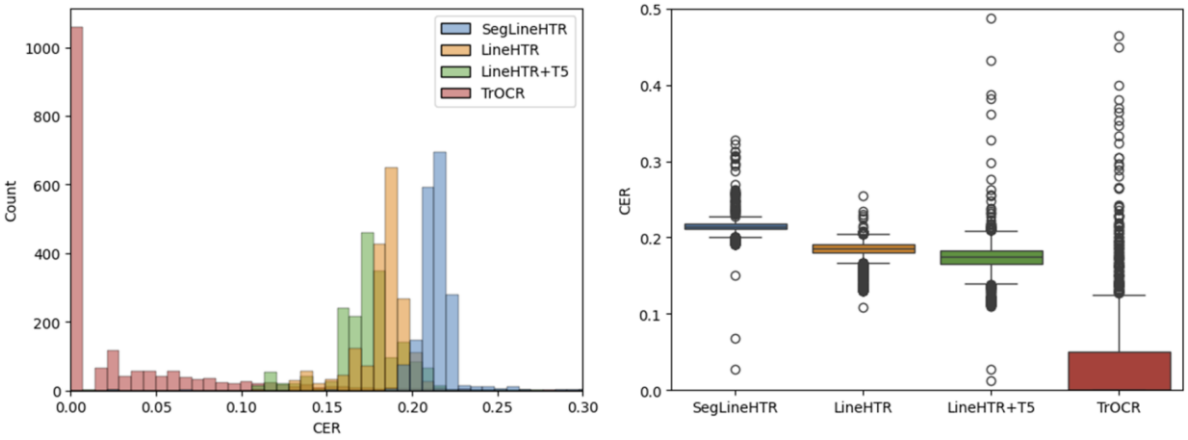


Рис. 7. Гистограмма и усиковая диаграмма критерия CER для разных моделей

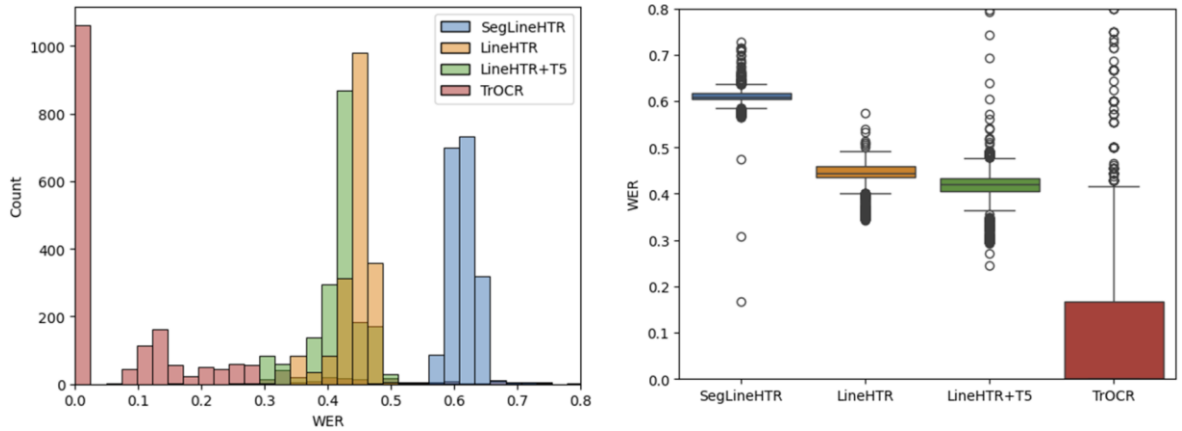


Рис. 8. Гистограмма и усиковая диаграмма критерия WER для разных моделей

Лучшее полученное решение – дообученный трансформер *TrOCR* с измененным алгоритмом предобработки изображений – был протестирован на исторических рукописных текстах, взятых из обучающего набора *Bentham*, содержащего 11473 изображения рукописей английского философа Джереми Бентама [8]. На рис. 9 показаны результаты сравнения оригинальной модели *TrOCR* и описанной в настоящей работе дообученной модели *TrOCR*.

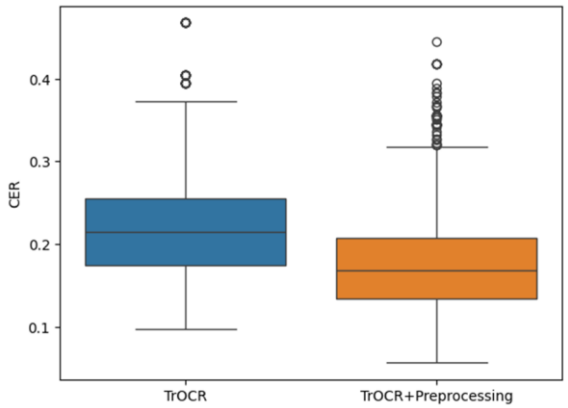


Рис. 9. Сравнение двух моделей TrOCR на данных из набора Bentham

Из представленного на рис. 9 графика можно сделать вывод, что изменения, внесённые в алгоритм предобработки данных для архитектуры *TrOCR* повышают качество работы данной модели. В

итоге, предложенное решение оказывается наиболее эффективным среди всех моделей, рассмотренных в данной работе.

Заключение

В рамках выполнения настоящей работы был проведён обзор существующих методов глубокого машинного обучения для задачи распознавания рукописных текстов. На основе результатов данного обзора был предложен и реализован алгоритм предобработки изображений средствами языка *Python* и библиотеки *OpenCV*. Разработана пользовательская реализация четырёх различных нейросетевых подходов, в том числе с использованием механизма сегментации изображений. Проведено обучение полученных моделей, в том числе дообучение модели *TrOCR*. Для сети с архитектурой трансформера был предложен и реализован новый подход к предобработке изображений и проведено её дообучение. Для удобства использования моделей разработано консольное приложение, позволяющее обучать и тестировать архитектуры на произвольных данных, переданных в качестве опций командной строки. Проведённый обзор и численные эксперименты показали работоспособность всех выбранных моделей для задачи распознавания рукописного текста. Лучшая из рассмотренных моделей *TrOCR* была дообучена на исторических рукописных текстах, где показала качество лучшее, чем у базовой модели.

Список источников

1. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks / A. Graves, S. Fernandez, F. Gomez, J. Schmidhuber //Proceedings of the 23rd international conference on Machine learning. – 2006. – С. 369-376.
2. Graves, A. Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks / A. Graves, J. Schmidhuber // Proceedings of the 21st International Conference on Neural Information Processing Systems (NIPS'08). – 2008. – С. 545-552.
3. Convolve, attend and spell: An attention-based sequence-to-sequence model for handwritten word recognition / L. Kang [et al.] //Pattern Recognition: 40th German Conference, GCPR 2018, Stuttgart, Germany, October 9-12, 2018, Proceedings 40. – Springer International Publishing, 2019. – С. 459-472.
4. TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models / Minghao Li [et al.] // Proceedings of the AAAI Conference on Artificial Intelligence. – 2023. – Т. 37. – №. 11. – С. 13094-13102.
5. IAM: A Comprehensive and Large-Scale Dataset for Integrated Argument Mining Tasks / L. Cheng et al. // Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. – 2022. – Vol. 1. – С. 2277–2287.
6. Ba J. Layer Normalization / J. Ba, J. Kiros, G. Hinton // arXive.org e-Print archive. – 2016. – DOI: 10.48550/arXiv.1607.06450.
7. Exploring the limits of transfer learning with a unified text-to-text transformer / C. Raffel [et al.] //Journal of machine learning research. – 2020. – Т. 21. – №. 140. – С. 1-67. – URL: http://jmlr.org/papers/v21/20-074.html.
8. ICFHR 2014 competition on handwritten keyword spotting (H-KWS 2014) / I. Pratikakis [et al.] // 2014 14th International Conference on Frontiers in Handwriting Recognition. – IEEE, 2014. – С. 814-819. – DOI: 10.1109/ICFHR.2014.142.