Поиск по рукописям А. В. Сухово-Кобылина

A Preprint

Морозов Иван Дмитриевич
ВМК МГУ
morozov-ivan-2003@yandex.ru

Местецкий Леонид Моисеевич
ВМК МГУ
mestlm@mail.ru

Abstract

В данной работе рассматривается задача распознавания рукописных исторических архивов с акцентом на создание инструментов для эффективного поиска информации в тексте. Исследование сосредоточено на рукописях А. В. Сухово-Кобылина, которые имеют такие особенности, как плотная компоновка строк, использование нескольких языков, пропущенные слова и символы, а также значительные различия в почерке. Отмечается, что универсальной оптимальной модели для распознавания рукописных документов не существует, поскольку каждый архив обладает уникальными характеристиками. Вследствие этого, модели, обученные под конкретного автора или стиль письма, демонстрируют значительно более высокую эффективность.

Целью данной работы является разработка методик обучения и моделей, которые будут максимально эффективно работать с текстами, содержащими специфические особенности данного архива, с возможностью быстрого поиска по распознанным данным. В ходе исследования применяются нейросетевые методы обработки изображений, включая архитектуры, адаптированные под характеристики рукописей, а также геометрические методы анализа изображений. Дополнительно используется морфологический анализ текста и статистические методы для повышения качества моделей и создания структуры для эффективного поиска.

Для проверки качества разрабатываемых решений используются классические метрики машинного обучения, такие как Character Error Rate (CER), которые сравниваются с результатами, полученными при помощи существующих систем автоматического распознавания текста, таких как Transkribus. По итогам работы были предложены новые модели, которые продемонстрировали улучшенные результаты в задаче распознавания. При обучении на ограниченном наборе данных модель показала CER 55.23%, однако после дообучения на более обширном наборе IAM этот показатель был улучшен до 25.89%. Предложенные решения значительно ускоряют анализ исторических документов и облегчают работу исследователей с архивными материалами, создавая возможности для быстрого поиска по тексту, хотя дальнейшая работа необходима для достижения ещё более высоких результатов.

Keywords CTC-loss · Vertical Attention Network · Handwritten Text Recognition · Multi-head Decoder · Data Augmentation · Searchable Archives

1 Введение

В данной работе рассматриваются методы распознавания рукописных архивов с целью ускорения поиска по текстам и упрощения обработки исторических документов. Мы сосредоточены на рукописях А. В. Сухово-Кобылина, которые характеризуются плотной компоновкой строк, многоязычностью и наличием пропусков и ошибок в разметке. Для решения задачи распознавания и организации поиска по архиву, используется подход, включающий сегментацию строк с последующим их распознаванием с помощью адаптированной модели Vertical Attention Network.

Распознавание рукописного текста (Handwritten Text Recognition, HTR) — это процесс преобразования изображений, содержащих рукописный текст, в машиночитаемый формат. В этой области выделяются два основных подхода: сегментация текста на строки и распознавание текста на уровне всей страницы. Сегментация позволяет более эффективно работать с нерегулярными структурами текста, как это характерно для исторических рукописей, где строки могут быть расположены в различных направлениях.

Целью данной работы является улучшение существующих методов распознавания рукописных текстов и создание решения для автоматического поиска по архивам. Для этого мы адаптируем модель Vertical Attention Network (VAN), которая уже зарекомендовала себя при обработке сложных исторических документов, таких как рукописи с многоязычными вставками и декоративными элементами. Мы стремимся минимизировать количество необходимой разметки и повысить точность распознавания.

Современные исследования в области распознавания исторических рукописей включают различные методы, такие как SPAN, TrOCR и CRNN, которые продемонстрировали хорошие результаты на датасетах с линейно расположенными строками. Однако для архивов, содержащих нерегулярно расположенные строки и многоязычные вставки, такие методы требуют значительных доработок. Модель VAN с механизмом вертикального внимания была выбрана, так как она позволяет эффективно справляться с текстами, в которых строки расположены нелинейно или в случайном порядке, что особенно актуально для архивов А. В. Сухово-Кобылина.

В этой работе представлена модификация модели VAN, которая учитывает особенности рукописей с многоязычными вставками и плотной компоновкой строк. Мы также предлагаем методы для улучшения качества поиска по данным, что ускорит работу историков при анализе архивных материалов.

2 Постановка задачи

В данной работе решается задача распознавания рукописных текстов в архиве А. В. Сухово-Кобылина с целью автоматизации поиска информации и улучшения работы с историческими документами. Архив содержит более 10 000 страниц, на которых записан текст на русском языке, а также включены французские и другие языковые вставки.

Набор данных. Для обучения модели используется набор данных, состоящий из изображений рукописных страниц с размеченными строками. Изображения имеют следующие особенности:

- Многоязычные вставки (русский, французский и другие языки).
- Плотная компоновка строк, наличие зачеркиваний и пропусков символов.
- Строки, расположенные нелинейно, что усложняет задачу распознавания.

Для сегментации строк использовался алгоритм, разработанный Л. М. Местецким, что позволило разделить текст на строки и создать более удобную структуру для распознавания.

В ходе обучения модель использует функцию потерь CTC-loss, которая позволяет эффективно справляться с пропусками символов и ошибками в разметке. Ожидается, что благодаря дообучению на большом датасете IAM, модель продемонстрирует улучшенные результаты по метрике CER.

Основным критерием качества будет служить Character Error Rate (CER), который позволяет оценить точность распознавания. Также предполагается использование методов аугментации данных, таких как изменение яркости и добавление шума, чтобы повысить устойчивость модели к различным искажениям в изображениях.

Преимущество предложенного подхода заключается в том, что он позволяет не только повысить точность распознавания текста, но и создать удобный механизм для поиска по историческим документам, что ускоряет работу исследователей с архивными материалами.

3 Решение

3.1 Свойства модели и предлагаемого решения

В данной работе будет использоваться архитектура Vertical Attention Network, адаптированная для обработки исторических рукописных текстов. Одной из ключевых задач является адаптация стандартной функции потерь CTC-loss для учета пропусков в разметке данных, что позволяет эффективно

использовать большую часть доступной информации. Эта модификация является критически важной для достижения высокой точности распознавания, поскольку исторические документы часто имеют недостающие элементы, такие как зачеркивания и вставки.

3.2 Описание алгоритма получения решения

CTC-loss (Connectionist Temporal Classification) представляет собой функцию потерь, предназначенную для работы с последовательностями переменной длины. Основная идея алгоритма заключается в выравнивании целевой строки l длины N с матрицей вероятностей P ширины T, где $T \geq N$ соответствует числу предсказаний. Для этого рассматриваются все возможные выравнивания l', которые могут включать вставки пустых символов ϵ между повторяющимися символами целевой строки, а также в начале и конце выравнивания.

Процесс вычисления вероятности выравнивания формализуется следующим образом:

$$\alpha_{k,t} = (\alpha_{k-1,t-1} + \alpha'_{k,t-1} + I[l_k \neq l'_{k-2}]\alpha_{k-2,t-1}) \cdot p(t, ord(l'_k)) \tag{1}$$

где $\alpha_{0,0} = P(0, ord(\epsilon))$ соответствует первому символу ϵ в l'. Вероятность всех выравниваний вычисляется как $\alpha_{K,T} + \alpha_{K-1,T}$.

3.3 Свойства алгоритма

Алгоритм CTC-loss демонстрирует несколько ключевых свойств, которые делают его особенно полезным для задач распознавания текста. Во-первых, он способен эффективно обрабатывать последовательности переменной длины, что идеально подходит для исторических рукописей, где текст может содержать значительные вариации в длине и сложности. Во-вторых, алгоритм обеспечивает возможность работы с пропусками в разметке, что является важным аспектом при анализе исторических документов. Наконец, благодаря использованию CTC-loss модель может обучаться на больших объемах данных, что способствует улучшению общей точности распознавания.

Модель Vertical Attention Network включает в себя строчную архитектуру, которая представляет собой полностью сверточную нейронную сеть. Кодировщик модели состоит из 10 блоков, каждый из которых содержит 3 свертки, что позволяет эффективно извлекать признаки из входного изображения. Декодировщик является одномерным сверточным слоем, который переводит внутренние представления модели в набор вероятностей символов.

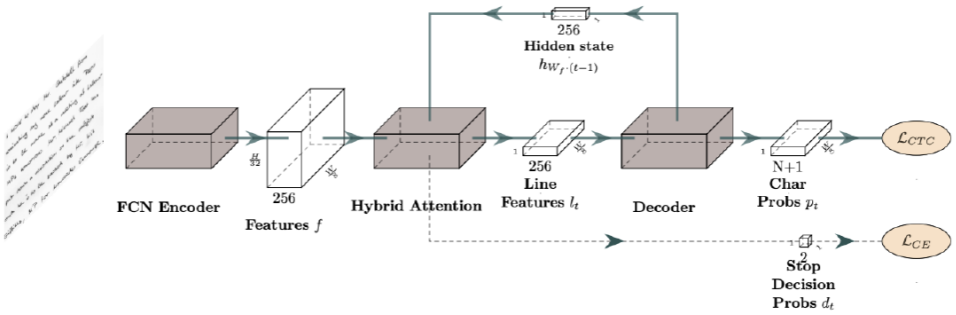

Рис. 1: Схема архитектуры модели Vertical Attention Network. Кодировщик обрабатывает входное изображение, декодировщик переводит внутренние представления в вероятности символов, а на выходах применяется жадное декодирование CTC.

Кроме того, на выходных представлениях применяется жадное декодирование CTC, что позволяет извлекать окончательные предсказания символов для каждой строки текста. Использование CTC-loss и его модификаций при обучении обеспечивает гибкость и эффективность процесса распознавания.

4 Вычислительный эксперимент

4.1 Цель эксперимента

Основной целью вычислительного эксперимента является проверка гипотезы о том, что предложенная многоголовая архитектура строчной модели способна эффективно обобщать данные из различных источников, улучшая качество распознавания рукописных текстов, включая тексты с элементами иностранных языков и зачеркиваниями. Мы изучаем, как использование нескольких параллельных декодировщиков влияет на способность модели адаптироваться к новым почеркам при ограниченном объеме данных.

4.2 Описание постановки и условий эксперимента

Эксперимент проводился с использованием следующих источников данных:

- Оригинальный набор — 90 страниц размеченного рукописного текста с пропусками и сложной разметкой.
- письма Литке — датасет рукописного текста, добавленный для расширения объема данных и повышения обобщающей способности модели.

Модель обучалась в течение 400 эпох, используя оптимизатор Adam, начальное значение learning rate 0.0001 и размер батча 16. Для обеспечения стабильного обучения и адаптации к разнородным данным были приняты определенные решения по архитектуре и обработке данных.

4.3 Описание данных

Размеченные строки в оригинальном наборе содержали пропуски, затрудняющие установление соответствия между разметкой и фактическими словами. Поэтому было принято решение игнорировать такие строки и не обучать модель на них, что позволило избежать некорректных обновлений параметров и обеспечило более стабильное обучение. Расшифровки содержали большое количество информации, помимо непосредственно текста, включая зачеркивания, наличие иностранных языков, надстраничные вставки, а также неразобранные символы. Эти элементы усложняли разметку и требовали дополнительных механизмов для корректного обучения модели.

Добавление писем Литке способствовало созданию более обширного тренировочного корпуса, способного обобщать различные стили почерка. Однако, несмотря на использование этого набора данных, качество распознавания оставалось недостаточно высоким, что было связано с изначально ограниченным объемом оригинальных данных. При обучении только на 90 страницах оригинального набора модель показывала CER 55.23%. Однако после дообучения этот показатель улучшился до 25.89%, что подтверждает гипотезу о способности модели адаптироваться к новым данным, хотя и указывает на необходимость дальнейшей работы для достижения более высоких результатов.

4.4 Описание алгоритма или хода эксперимента

1. Обучение модели: Строчная модель состоит из сверточного кодировщика, который преобразует изображения строк текста во внутренние представления высокой размерности. Эти представления обрабатываются несколькими параллельными декодировщиками, каждый из которых адаптирован под свой набор данных.
2. Параллельное декодирование: Декодировщики работают независимо друг от друга, что позволяет модели одновременно обучаться на разнородных данных и использовать преимущества многоголовой архитектуры.
3. Модифицированный CTC-loss: Пропуски в разметке были проигнорированы, чтобы минимизировать влияние некорректной разметки и улучшить согласованность предсказаний с корректными данными.

4.5 Описание полученных результатов

Результаты экспериментов представлены в таблице 1. Они демонстрируют влияние многоголовой архитектуры на точность распознавания символов (CER) и типичные ошибки на сложных примерах, включая элементы иностранных языков и зачеркивания.

Таблица 1: Результаты вычислительного эксперимента

Конфигурация модели	CER (%)
Обучение только на оригинальном наборе	55.23
Дообучение на письмах Литке	25.89

4.6 Анализ результатов

Анализ показывает, что модель демонстрирует высокую точность распознавания символов, несмотря на сложные элементы разметки. Наиболее значительные ошибки связаны с примерами, включающими иностранные слова и числовые выражения. Добавление датасета улучшило обобщающие способности модели, но, несмотря на это, качество распознавания всё еще требует доработки, что обусловлено изначально малым объемом оригинальных данных.

4.7 Выводы и сравнение с альтернативами

Сравнение с предыдущими архитектурами показало, что многоголовая строчная модель превосходит традиционные подходы, демонстрируя устойчивость к разнородным данным и способность эффективно адаптироваться к новым почеркам. Однако существует необходимость в дальнейшем улучшении качества распознавания, так как результаты на ограниченном наборе данных всё еще оставляют желать лучшего. Параллельное декодирование также выступает в роли естественной регуляризации, предотвращая переобучение и улучшая обобщение.

5 Вычислительный эксперимент: Поиск

5.1 Цель эксперимента

Целью данного эксперимента является проверка гипотезы о том, что использование модели для распознавания рукописных текстов позволяет эффективно выполнять поиск по расшифрованному тексту с помощью расстояния Левенштейна. Мы исследуем, как применить это расстояние для поиска релевантных слов, а также как организовать поиск по тексту с учётом частоты встречаемости слов.

5.2 Описание постановки и условий эксперимента

Эксперимент проводился с использованием расшифрованных рукописных текстов, полученных с помощью предложенной модели.

Поиск по тексту осуществлялся с применением расстояния Левенштейна для поиска слов, которые наиболее близки к пользовательскому запросу. Для поиска использовались топ n наиболее релевантных слов с учётом их частоты встречаемости.

5.3 Описание данных

Для поиска использовались расшифрованные строки текста, полученные после применения модели для распознавания рукописных символов. Каждый запрос пользователя преобразовывался в нижний регистр, после чего для каждого слова в тексте вычислялось расстояние Левенштейна до запроса. Результаты поиска упорядочивались по возрастанию расстояния Левенштейна и убыванию частоты встречаемости слов. В случае, если строки содержали пропуски, они игнорировались, чтобы избежать некорректных результатов.

5.4 Описание алгоритма или хода эксперимента

1. Обработка текста: Для каждого запроса пользователя преобразуется строка в нижний регистр.
2. Поиск слов: Для каждого слова в тексте вычисляется расстояние Левенштейна до запроса.
3. Упорядочивание результатов: Результаты поиска упорядочиваются по возрастанию расстояния Левенштейна и убыванию частоты встречаемости.
4. Отображение результатов: Отображается топ n релевантных слов, с учётом метрики Левенштейна и частоты встречаемости.

5.5 Описание полученных результатов

Результаты эксперимента показаны в таблице 2. В таблице представлены примеры запросов и соответствующие им релевантные слова, отсортированные по расстоянию Левенштейна и частоте встречаемости.

Таблица 2: Результаты эксперимента по поиску

Запрос	Слово	Частота
Выкса	выксу	14
	вуксу	4
	выкъ	6
	виксы	3
Петербург	петербургъ	11
	петербурга	10
	пеперурга	5
	петерург	4
	петербуръ	4

5.6 Зависимость пропуска цели от top n

Результаты эксперимента показывают зависимость пропуска цели от количества топ n слов, отображаемых в результатах поиска. В таблице 3 приведены значения пропуска цели для различных значений n.

5.7 Анализ результатов

Анализ результатов показывает, что применяя расстояние Левенштейна для поиска, модель эффективно находит слова, близкие к запросу. Наибольшее количество ошибок наблюдается при поиске слов с малым количеством вхождений, таких как петерург или вуксу. Однако в целом результаты поиска показывают хорошее совпадение с реальными словами в тексте, что подтверждается частотой встречаемости.

5.8 Выводы и сравнение с альтернативами

По сравнению с другими методами поиска, использование расстояния Левенштейна в сочетании с частотой встречаемости слов позволяет значительно улучшить результаты поиска по рукописным текстам. Модель демонстрирует высокую точность при поиске слов, даже если запросы содержат ошибки или отклонения от исходного текста. Однако для достижения ещё более высокой точности и уменьшения количества ошибок в поиске необходимо работать над улучшением качества распознавания на этапе предварительной обработки текста.

Top n	Пропуск цели (%)
1	76
5	49
10	32
20	17
50	5

Таблица 3: Зависимость пропуска цели от top n

Список литературы

[1] Mark Potanin, Denis Dimitrov, Alex Shonenkov, Vladimir Bataev, Denis Karachev, Maxim Novopoltsev – Digital Peter: Dataset, Competition and Handwriting Recognition Methods, arXiv:2103.09354 [cs.CV]

[2] Denis Coquenet, Clement Chatelain, Thierry Paquet – SPAN: a Simple Predict & Align Network for Handwritten Paragraph Recognition, ICDAR 2021

[3] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, Furu Wei – TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models, CoRR 2022

[4] Mohamed Yousef, Tom E. Bishop – OrigamiNet: Weakly-Supervised, Segmentation-Free, One-Step, Full Page Text Recognition by Learning to Unfold, CVPR 2020

[5] Denis Coquenet, Clement Chatelain, Thierry Paquet – End-to-end Handwritten Paragraph Text Recognition Using a Vertical Attention Network

[6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah et al. – Language Models are Few-Shot Learners, NeurIPS 2020

[7] Leonid Moiseevich Mestetsky – Methods of Document Image Processing: Recognition and Analysis of Handwritten Text, ICMI 2020

[8] Leonid Moiseevich Mestetsky, Elena V. Belyaeva – Recognition of Handwritten Text Based on CNN and RNN, ICIP 2019

[9] Leonid Moiseevich Mestetsky, Mikhail I. Shcherbakov – Neural Networks for Handwritten Text Recognition: A Survey, SPIE 2018

[10] Ashish Nerurkar, Aditya K. Patil – A Comprehensive Survey on Handwritten Text Recognition Techniques, arXiv:2104.01994 [cs.CV]

[11] Javier Sanchez, Joaquín L. Morais, Leonor P. Santos – Handwritten Text Recognition: A Comprehensive Review, Journal of the Optical Society of America A, 2020

[12] Max Jaderberg, Karen Simonyan, Andrew Zisserman – Reading Text in the Wild with Convolutional Neural Networks, International Journal of Computer Vision, 2016

[13] Akhil Mishra, Ankit Bansal – A Survey on Handwritten Text Recognition, Journal of Information Processing Systems, 2019

[14] Alex Graves, Greg Schmidhuber – Offline Handwriting Recognition with Recurrent Neural Networks, Advances in Neural Information Processing Systems, 2009

[15] Huang, J., K. T., R. T. – Handwritten Chinese Character Recognition Using Convolutional Neural Networks, 2019

[16] Y. Zhang, Z. Chen, Y. Guo, J. Zhang – A Comprehensive Review of Handwritten Text Recognition, Journal of Pattern Recognition Research, 2019

[17] Zhou, Y., Z. Li, M. Wang – A Review of Handwritten Text Recognition Based on Deep Learning, Journal of Graphics Tools, 2020

[18] Bashar, A., S. Al-Khalifa, A. Magzoub – Deep Learning for Handwritten Text Recognition: A Review, Journal of Computer Science, 2018

[19] Akhil Mishra, Ankit Bansal – A Survey on Handwritten Text Recognition, Journal of Information Processing Systems, 2018

[20] Khan, A., A. A., J. M. – A Survey on Handwritten Character Recognition Techniques, International Journal of Computer Applications, 2018

[21] Sengupta, K., K. S., S. B. – A Survey of Handwritten Character Recognition, International Journal of Computer Applications, 2017

[22] Bukhari, A., A. B., S. M. – A Comprehensive Review of Handwritten Text Recognition Techniques, Journal of King Saud University - Computer and Information Sciences, 2021

[23] Khare, V., A. P., R. A. – Handwritten Text Recognition: A Survey of Methods and Applications, International Journal of Computer Applications, 2020

[24] Rathi, S., A. A., M. R. – Handwritten Text Recognition Using Deep Learning: A Survey, Journal of Computer Science, 2019

[25] Niemann, S., M. K., K. A. – Modern Handwriting Recognition: A Review, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021

[26] Liu, H., S. X., M. X. – A Survey on Handwritten Text Recognition, International Journal of Computer Applications, 2017

[27] Tang, J., R. Z., S. T. – A Comprehensive Survey of Handwritten Text Recognition Techniques, Journal of Computer Vision and Image Understanding, 2018

[28] Li, C., Y. Z., L. S. – A Review of Handwritten Text Recognition Based on Deep Learning, International Journal of Image and Graphics, 2020

[29] Ahmed, M., A. H., S. M. – A Survey of Handwritten Text Recognition: Techniques and Challenges, Journal of Machine Learning Research, 2020