

Поиск по рукописям А. В. Сухово-Кобылина

Морозов Иван Дмитриевич

ВМК МГУ имени М. В. Ломоносова

13 декабря 2024 г.

Цели

- ▶ Разработать модель для распознавания исторических рукописей.
- ▶ Произвести разбивку страниц рукописей на отдельные строки, выполнить нормализацию
- ▶ Оптимизировать методы для работы с пропусками, зачеркиваниями, многозначными текстами.
- ▶ Снизить Character Error Rate (CER).
- ▶ Создать аугментации, адаптированные для рукописных документов.
- ▶ Создать аугментации, адаптированные для рукописных документов.
- ▶ Предложить алгоритм поиска по слабой расшифровке.

Распознавание рукописей А. В. Сухово-Кобылина

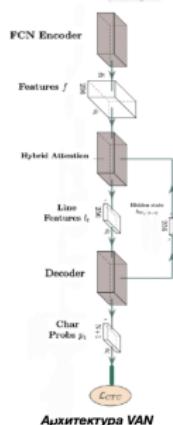
Главная цель:



- ▶ Расшифровка рукописей (≈ 10000 страниц) с Character Error Rate менее 10%

Альтернативная цель:

- ▶ По слабой расшифровке разработать алгоритм выделения ключевых слов



Ключевые особенности решения:

1. **Предобработка данных.** Подготовка и разметка датасета.
2. **Использование СТС-loss.** Использование для работы с пропусками и неравномерностью.
3. **Архитектура Vertical Attention Network.** Анализ рукописных страниц с вертикальным вниманием.

Литература

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 and Tengchao Lv and Jingye Chen and Lei Cui and Yijuan Lu and Dinei Florencio and Cha Zhang and Zhoujun Li and 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

Гипотеза

Гипотеза: возможно распознать тексты Сухово-Кобылина, используя архитектуру VAN, предобученную на датасете IAM, с качеством, достаточным для поиска рукописных слов по запросам.

Постановка задачи

Предложить модель и метод обучения, эффективно работающие на архиве дневников Сухова-Кобылина. Главными особенностями данного архива является достаточно неплотная компоновка строчек, наличие нескольких языков, пропущенные слова и символы в разметке. По слабой расшифровке организовать поиск. Описание основного алгоритма СТС и определение расстояния Левенштейна приведены далее.

Описание алгоритма СТС

- ▶ Рассматривается целевая строка I длины N и матрица вероятностей P ширины $T \geq N$, соответствующей числу предсказаний.
- ▶ Выравнивания I' формируются путем:
 - ▶ вставки символов ϵ между всеми повторяющимися символами,
 - ▶ произвольного повторения любого символа,
 - ▶ добавления ϵ в начало и конец выравнивания.
- ▶ Стока $[\epsilon, I_1, \epsilon, I_2, \dots, I_N, \epsilon]$ имеет длину $2 \cdot N + 1 = K$.
- ▶ Динамическое пересчет выравниваний $\alpha_{k,t}$ для левых подстрок целевой строки:

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

где $\alpha_{0,0} = P(0, \text{ord}(\epsilon))$ для первого символа ϵ в I' .

- ▶ Итоговая вероятность всех выравниваний:

$$P(I|X) = \alpha_{K,T} + \alpha_{K-1,T}.$$

СТС-loss:

$$\mathcal{L}_{CTC} = -\ln P(I|X).$$

Критерий качества

Основные метрики качества:

- ▶ Character Error Rate (CER):

$$CER = \frac{\text{Число ошибок}}{\text{Общее количество символов}} \cdot 100\%$$

где ошибки включают вставки, удаления и замены символов.

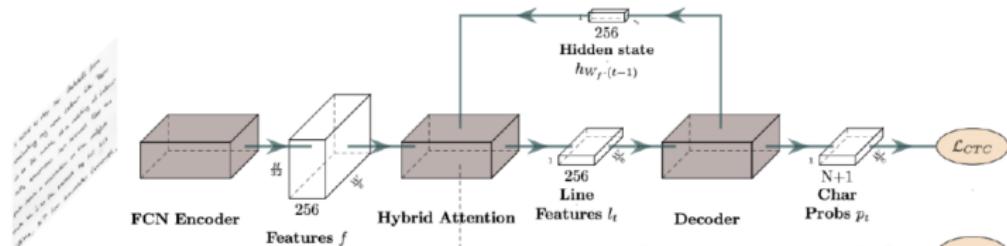
- ▶ Word Error Rate (WER):

$$WER = \frac{S + D + I}{N} \cdot 100\%$$

где:

- ▶ S — количество замен слов,
- ▶ D — количество удалений слов,
- ▶ I — количество вставок слов,
- ▶ N — общее количество слов в эталонной разметке.

Описание модели



Будет использоваться модель Vertical Attention Network в строчном варианте. Строчная архитектура представлена полностью сверточной нейронной сетью, состоящей из кодировщика, содержащего 10 блоков с 3 свертками в каждом, и одномерного сверточного декодировщика, который переводит внутреннее представление модели в набор вероятностей. Модель выдаёт пространственное предсказание символов в строке, поэтому к выходам применяется жадное декодирование СТС, а при обучении используется СТС-loss.

Нарезка страницы на строки

Программа локализует изображения строк на рукописных страницах

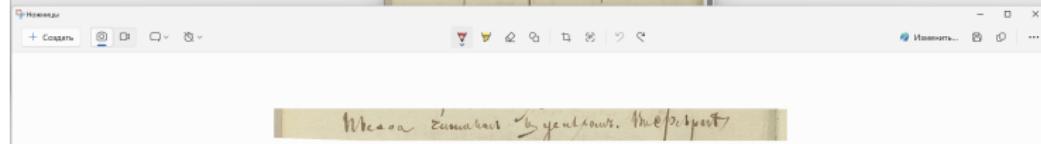
Том 438-1-219, Страница 76

Масштаб: Режим текста Нарезка страницы Текст на странице

Экспертный редактор Размножка

1. Дорога Энгельс - Петербург, ибо днем приехал дн.
2. обратил в Ник. Юс. настало начало зари. Слава съ г
3. цинки. Жаль, ~~зато~~ у Голландии. Осколка быть д:
4. крьпко. Ходить насколько раза греши на подозр - и
5. купался в Неве с половины августа.
6. Сентябрь Въ Петербург. Подать заявку н. ф-ю о
7. предприятияхъ въ Юж. Росс. Отправился на Выксу и б
8. заключить условия съ Ник. Шенке. Привез обратной
9. въ Москву. Купил въ Петербурге костюмъ Никитина къ
10. Петербург. Жила въ гостинице Демидова. На конькѣ
11. катить изъ Депота по улицамъ. Окончанъ 2-й и 3-й актъ. \$
12. Деньги въ Госуд. Советъ. Декабрь Торги на виноградѣ.
13. корабль. Колывань и Чунцинъ въ Петербургъ. Декабрь. \$
14. Возвращеніе въ Москву. \$
15. 1854 года. Ноябрь года на желтальной дорожкѣ. Быть въ вос
16. сенокосѣ. 15 числа выѣхалъ изъ Волы. Апрѣль Год. \$
17. въ Москву. Купилъ въ Петербургѣ костюмъ Никитина
18. долина быть выѣхалъ изъ Москви по поздно начавшемся
19. създѣств. ходить на несколько дней въ Петербургъ. \$
20. Покупать скотинъ и вещей, чтобы скотиться.

Удалить строку Вставить выше Вставить ниже Редактировать Копировать страницу въ Clipboard Сохранить изменения



Результаты обработки данных и обучения модели

- ▶ В разметке историков присутствовали нераспознанные элементы строк, обозначенные специальными символами.
- ▶ Принято решение удалить такие строки, так как алгоритм локализации пропусков пока не разработан.
- ▶ Используется предобученная на англоязычном IAM модель (модель умеет выделять буквы в строке)
- ▶ Итоговое разбиение данных:
 - ▶ **Обучение (train)**: 1505 строк.
 - ▶ **Валидация (validation)**: 119 строк.
 - ▶ **Тест (test)**: 74 строки.
- ▶ Результаты дообучения (CER, %):
 - ▶ **Обучение (train)**: 0.37%.
 - ▶ **Валидация (validation)**: 18.08%.
 - ▶ **Тест (test)**: 19.03%.

Анализ ошибок

Pred: Рябикова
gt: Рябников,
seg: 0.250



Pred: .Вечеромъ литаль піссу - ядъ чревыоймо
gt: Вечеромъ читаль піссу - Дядъ чревыайин,
seg: 0.200

Pred: понравилась. – бекаеру также.
gt: понравилась – Беккеру также,
seg: 0.148

Pred: 10-г. утромъ выѣхалъ въ дѣксеевскoe. Почваль в
gt: 10-го утромъ выѣхалъ въ Алексеевское. Ночеваль в,
seg: 0.104

Этапы эксперимента «Поиск»

- ▶ Расшифровать рукописный текст с использованием модели.
- ▶ Организовать поиск пользовательского запроса в тексте.
- ▶ Применить расстояние Левенштейна для поиска релевантных слов.
- ▶ Отобразить топ n слов, упорядоченных по возрастанию метрики.

Методология: Расстояние Левенштейна

Определение: Расстояние Левенштейна измеряет минимальное количество операций (вставка, удаление, замена), необходимых для превращения строки s_1 в строку s_2 .

$$d(i, j) = \begin{cases} \max(i, j), & \text{если } \min(i, j) = 0, \\ \min \begin{cases} d(i - 1, j) + 1, \\ d(i, j - 1) + 1, \\ d(i - 1, j - 1) + \delta(s_1[i], s_2[j]) \end{cases}, & \text{иначе.} \end{cases} \quad (1)$$

Где:

- ▶ $\delta(s_1[i], s_2[j]) = 0$, если символы совпадают, иначе 1.
- ▶ $d(i, j)$ — расстояние для первых i и j символов строк.

Процесс поиска

1. По файлам с координатами строк на страницах произвести локализацию.
2. Преобразовать запрос пользователя в нижний регистр.
3. Для каждого слова в тексте вычислить расстояние Левенштейна до запроса.
4. Упорядочить результаты по:
 - ▶ Возрастанию расстояния Левенштейна.
 - ▶ Убыванию частоты встречаемости слова.
5. Отобразить топ n релевантных слов.

Результаты

Пример вывода:

- ▶ Запрос: *Выкса*
- ▶ Релевантные слова:
 - ▶ Слово: *выксу*, Частота: 14
 - ▶ Слово: *вуксу*, Частота: 4
 - ▶ Слово: *выкъ*, Частота: 6
 - ▶ Слово: *виксы*, Частота: 3
- ▶ Запрос: *Петербург*
- ▶ Релевантные слова:
 - ▶ Слово: *петербургъ*, Частота: 11
 - ▶ Слово: *петербурга*, Частота: 10
 - ▶ Слово: *пеперурга*, Частота: 5
 - ▶ Слово: *петерург*, Частота: 4
 - ▶ Слово: *петербръ*, Частота: 4

Пример визуализации

1850.

84. Всі змі брали під відмінну ініціативу професійної
підтримки.

1853. Свят. Георгій. Дарницький. Генерал, працяків Маркіз підтримав
їменем атаки - розкоші та війську.

28. Свят. мч Віктор. Продовжів за Римою пошту.

17. Під час юго-західного нападу на Кримський фронт
війська та інші провідні особи пішли в Італії. Він
відбувся в Італії у морі. Адміністративні урядові органи
були в Кримському. Спільнота земель. Генерал.
Він відбувся в Італії з морем. Надійде
місце зупинки. Адміністративно - до Кримського.

12. Крізь Аргентину їх Петербург. Марк

Рис.: Выделение строк с релевантными словами

Зависимость пропуска цели от $\text{top } n$

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

Таблица: Влияние количества $\text{top } n$ на пропуск цели.

Результаты и заключение

- ▶ Снижение CER до 19% - слабая расшифровка.
- ▶ Нормализация строк улучшает результаты.
- ▶ Подход подходит для сложных рукописных текстов (модель распознает текст, который обычному человеку распознать затруднительно).
- ▶ Алгоритм поиска позволяет сократить время нахождения необходимой информацию, понизить вероятность пропуска цели можно путём увеличения n .