From image to LaTeX

Перевод письменных математических документов в формат LaTeX

Хрыкин Яромир Александрович

August 2025

Аннотация

В работе представлен конвейерный метод автоматизированного преобразования рукописных и отсканированных конспектов в форматированный код LaTeX. Подход сочетает детекцию и группировку строк текста, их обработку квантизированной vision–language моделью и пост-редактирование результатами языковых моделей. Такой метод позволяет повысить точность распознавания математических выражений и обеспечить баланс между качеством и вычислительной эффективностью, что делает его практически применимым в образовательной и научной среде.

Ключевые слова

OCR, LaTeX, Vision-Language Models, Handwritten Notes, Document Digitization

1 Introduction

Задача преобразования рукописных и отсканированных конспектов в структурированный формат LaTeX имеет высокую практическую значимость. Такие цифровые версии заметок повышают удобство восприятия и распространения материалов, а также открывают возможности их использования в автоматизированных системах обработки текста и формул.

Существующие решения обладают существенными ограничениями. Открытые OCR-инструменты показывают низкую точность при распознавании математических выражений, а коммерческие сервисы с приемлемым качеством часто оказываются дорогостоящими и малодоступными. Даже крупные языковые модели, например GPT, демонстрирующие хорошие результаты, ограничены по количеству обрабатываемых изображений и плохо подходят для массовой конвертации конспектов.

В данной работе предлагается собственный метод, который устраняет указанные недостатки. Подход основан на построчной сегментации изображения, объединении близко расположенных строк в блоки и их последующей обработке квантизированной vision–language моделью. Для повышения точности распознавания результаты дополнительно корректируются двумя последовательными языковыми моделями. Такой конвейер обеспечивает более точное, масштабируемое и доступное решение по сравнению с существующими альтернативами.

2 Обзор литературы

На сегодняшний день существует ограниченное количество решений для автоматизированного преобразования рукописных конспектов в формат LaTeX. Наиболее распространённые методы можно разделить на два направления: классические OCR-системы и современные мультимодальные модели.

Одним из наиболее заметных инструментов является PaddleOCR [6]. Данный фреймворк демонстрирует хорошие результаты в распознавании печатного текста на множестве языков и поддерживает работу с рукописными данными. Однако его точность при обработке математических выражений и сложных рукописных формул остаётся ограниченной. PaddleOCR в большей степени ориентирован на извлечение текста, а не на корректное восстановление структурированной разметки в LaTeX. Кроме того, в оригинальной работе [6] внимание уделялось в первую очередь задачам универсального OCR, а не специализированному восстановлению математического контента.

В последние годы всё больше внимания привлекают мультимодальные трансформерные модели, способные связывать изображения и текст. Примером являются модели семейства Qwen-VL [2], Donut [7], а также Nougat [3]. Эти подходы демонстрируют значительный прогресс в обработке сложных документов, включая математические формулы. Donut и Nougat изначально ориентированы на работу с научными статьями и рукописными документами, а Qwen-VL показывает универсальные возможности благодаря обучению на масштабных мультимодальных данных. Несмотря на высокое качество, такие модели остаются вычислительно затратными и часто требуют дообработки текста для получения корректного синтаксиса LaTeX. В частности, работы [7, 3] отмечают необходимость постобработки результатов и исправления ошибок синтаксиса. Также стоит выделить более узкоспециализированные исследования, такие как подход im2latex [5], в котором используется архитектура энкодер–декодер для преобразования изображений формул в код LaTeX. Модель демонстрирует высокую точность на печатных математических выражениях, но плохо обобщает на рукописные данные и в полной мере не решает задачу восстановления конспектов.

Отдельное направление исследований связано с анализом структуры документов. Здесь применяются специализированные парсеры макета страницы, например layout-parser [10] и dhSegment [1], которые используют методы сегментации и классификации областей для выделения заголовков, абзацев и других элементов. Подобные подходы ориентированы в первую очередь на печатные документы с регулярной структурой.

Другой класс решений основан на объектных детекторах с поддержкой текстовых запросов, таких как GroundingDINO [8] и OWLv2 [9]. Эти модели позволяют выделять отдельные фрагменты текста или графические элементы на изображении страницы, что открывает возможности для построения более гибких систем сегментации рукописных заметок.

Наконец, важным направлением является детекция отдельных строк текста. В частности, в PaddleOCR предусмотрен модуль для выделения строк, который может использоваться независимо от полной OCR-системы. Подобные методы дают возможность строить более модульные и масштабируемые конвейеры, где детекция текста отделена от этапа его распознавания.

3 Методология

Разработанный подход направлен на извлечение текстовой информации из изображения и её последующее преобразование в структурированный формат LaTeX. Пользователь загружает одно изображение через веб-интерфейс, реализованный на `Streamlit`, после чего система возвращает результат в виде текстового файла и текстовой строки.

Для выделения текстовых областей применяется детектор, реализованный в библиотеке `PaddleOCR` [4]. Его основой является модель **DBNet** (Differentiable Binarization), которая решает задачу детекции текста как задачу сегментации. Архитектура детектора включает сверточный бэкбон (например, ResNet) и пирамиду признаков (FPN) для извлечения многомасштабных представлений. Выходом сети является карта вероятностей для каждого пикселя, отражающая принадлежность к текстовой области. На этапе постобработки карта вероятностей преобразуется в бинарную маску, из которой извлекаются и аппроксимируются полигональные границы текстовых строк.

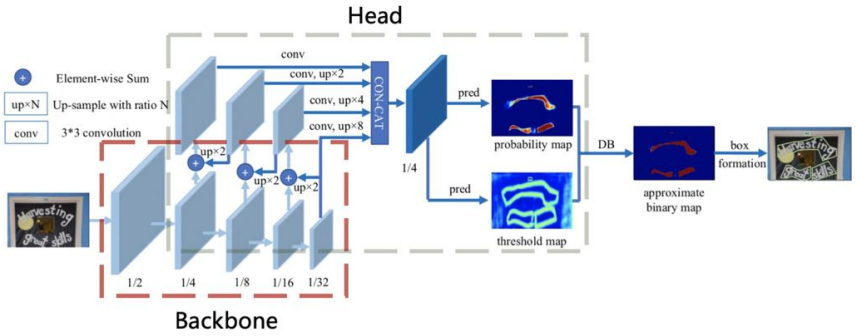


Figure 5: Architecture of the text detector DB. This figure comes from the paper of DB (Liao et al. 2020). The red and gray rectangles show the backbone and head of the text detector separately.

С помощью этой модели формируем множество полигонов $P = \{p_1, p_2, \ldots, p_n\}$, каждый из которых ограничивает отдельную строку текста. Для объединения близко расположенных строк в единые текстовые блоки применяется метрика перекрытия полигонов:

$$M(p_i, p_j) = \frac{S(p_i \cap p_j)}{\min(S(p_i), S(p_j))},$$

где $S(p_i)$ — площадь полигона p_i, а $S(p_i \cap p_j)$ — площадь пересечения полигонов p_i и p_j. Если $M(p_i, p_j) > 0.04$, строки p_i и p_j объединяются в один блок. Такой подход позволяет формировать структурированные сегменты текста и избежать чрезмерного дробления.

Каждый полученный блок передаётся в мультимодальную модель `Qwen2.5-VL-7B-Instruct-AWQ` [11]. Архитектура модели включает три основных компонента: визуальный энкодер, модуль проекции и языковую модель. В качестве визуального энкодера применяется модифицированный трансформер ViT (Vision Transformer), который преобразует изображение блока в набор эмбеддингов. Далее используется модуль проекции (linear projection), отображающий эмбеддинги в пространство, совместимое с токенами языковой модели.

Языковое ядро `Qwen2.5` построено по архитектуре **decoder-only Transformer**, в которой все блоки

состоят исключительно из декодеров без энкодерной части. Такая организация позволяет модели эффективно решать задачи условной генерации текста, опираясь как на текстовые токены, так и на мультимодальные эмбеддинги. Каждый декодерный блок включает механизмы *multi-head self-attention*, *feed-forward network* (FFN), нормализацию (LayerNorm) и остаточные соединения.

Механизм самовнимания реализуется в форме **каузального masked self-attention**, при котором каждая позиция последовательности может обращаться только к предыдущим токенам. Это гарантирует корректность авторегрессионной генерации. Внутри многоголового внимания вычисляется набор весовых матриц Q, K и V, которые формируют распределение внимания по ключам и позволяют учитывать как локальные, так и дальние зависимости в последовательности. FFN-компоненты включают две линейные проекции с нелинейностью типа GELU, что повышает выразительность модели и устойчивость к градиентному затуханию.

Для объединения визуальных и текстовых данных применяются специальные токены-маркеры, которые позволяют проецировать визуальные эмбеддинги в то же пространство, что и текстовые токены. Таким образом, модель обрабатывает единую последовательность, где визуальная информация интегрирована в общий поток внимания, что обеспечивает согласованное мультимодальное представление.

В используемой версии применяется метод **AWQ (Activation-aware Weight Quantization)**, при котором веса трансформера квантованы в низкий разряд (например, 4 бита), но с учётом распределения активаций в разных слоях. Такая стратегия позволяет минимизировать погрешность квантизации на критичных путях распространения сигнала и сохранить качество генерации при существенном снижении вычислительных затрат и памяти.

После обработки всех блоков результаты объединяются в один текст, который затем проходит этап пост-редактирования. Для этого применяются две последовательные итерации использующие только языковую часть модели `Qwen2.5-7B-Instruct-AWQ` [12]. Эти шаги включают исправление ошибок распознавания, нормализацию пунктуации и преобразование математических выражений в корректный синтаксис LaTeX.

4 Эксперименты

Одной из ключевых сложностей проекта стало отсутствие специализированного датасета для задачи конвертации рукописных заметок в LaTeX. Существуют наборы данных, содержащие изображения формул и их представления в LaTeX, однако они не охватывают случай рукописных текстов. Это обстоятельство предопределило необходимость комбинирования различных подходов и разработки собственной схемы обработки. Дополнительным ограничением являлись вычислительные ресурсы: все эксперименты проводились на одной видеокарте NVIDIA RTX 3060 Ti с 8GB видеопамяти, что не позволяло напрямую использовать крупные модели для обработки изображений целиком.

На первом этапе была проверена гипотеза о возможности непосредственного применения больших мультимодальных моделей (например, `Qwen-7B-AWQ`) [12] ко всей странице рукописного текста. Однако эксперимент показал, что такой подход неработоспособен: модель допускала ошибки в формулах, пропускала отдельные слова и целые фрагменты текста. Кроме того, передача полного изображения страницы выходила

за пределы доступной видеопамяти, что делало данный метод неприменимым. Это послужило основанием для перехода к идее последовательного разбиения документа на более мелкие части.

Первыми были опробованы специализированные парсеры структуры документа, такие как `layout-parser` [10] и `dhSegment` [1]. Эти методы показали низкую эффективность: их алгоритмы ориентированы на печатные документы с регулярной структурой, но не на свободные рукописные заметки.

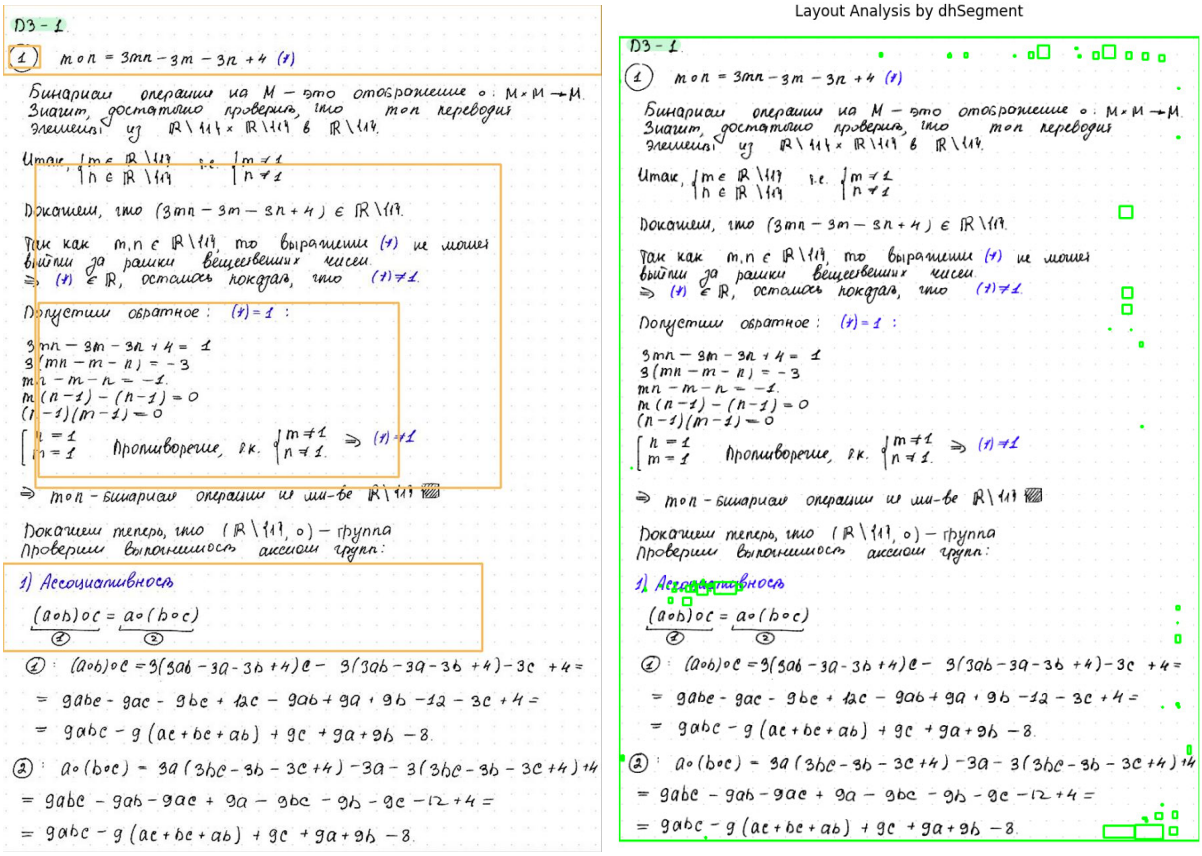


Figure 1: Примеры работы парсеров структуры документа (`layout-parser` и `dhSegment`).

В качестве следующего шага были протестированы методы объектного детектирования с поддержкой текстовых запросов, такие как `GroundingDINO` [8] и `OWLv2` [9]. Они показали лучшие результаты по сравнению с предыдущим подходом, однако сохранялись проблемы: детекторы создавали множество пересекающихся прямоугольников и часто не выделяли фрагменты текста в целом.

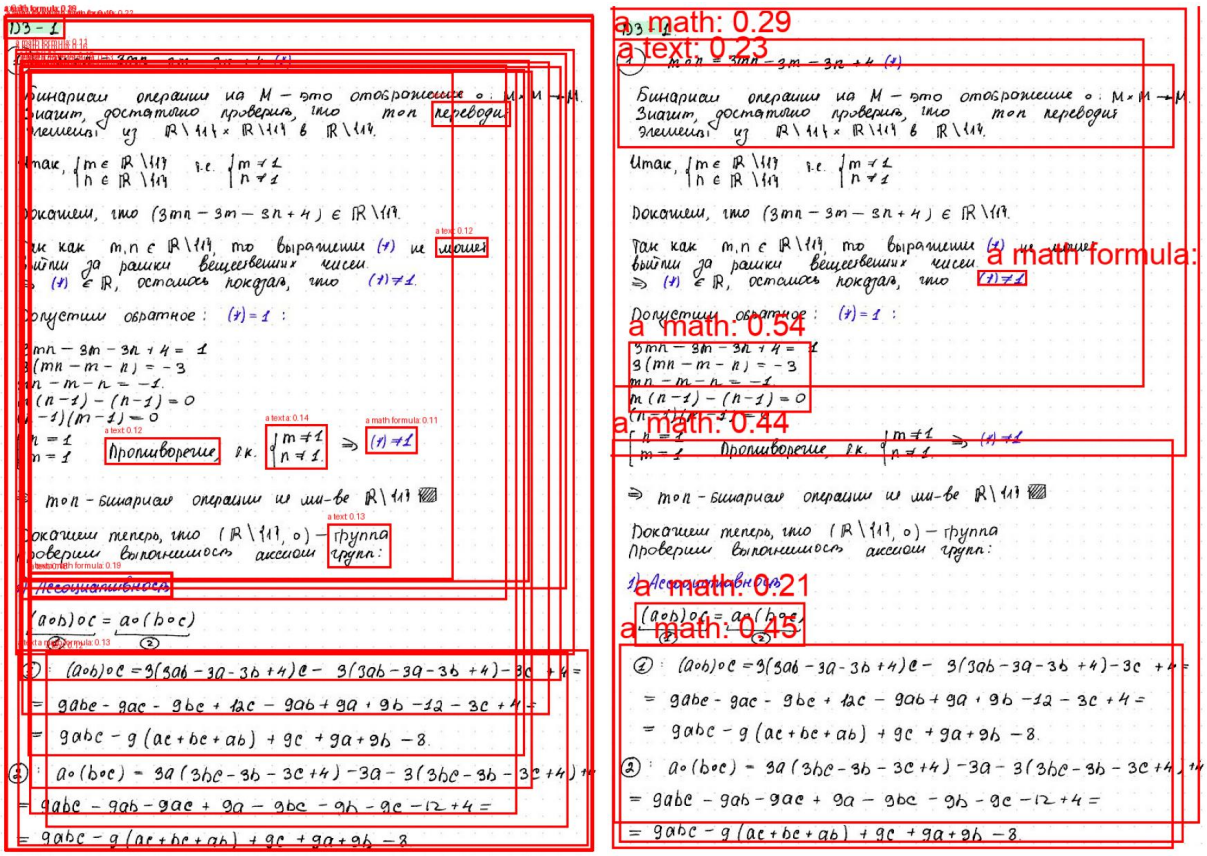


Figure 2: Примеры результатов работы детекторов объектов (`GroundingDINO` и `OWLv2`).

Учитывая указанные недостатки, было решено перейти на более низкий уровень разбиения — детекцию отдельных строк текста. Здесь перспективным оказался детектор строк, реализованный в библиотеке `PaddleOCR`. В отличие от полной end-to-end системы `PaddleOCR`, которая показала неудовлетворительные результаты для русскоязычных рукописных заметок, именно модуль детектирования строк оказался достаточно точным и устойчивым.

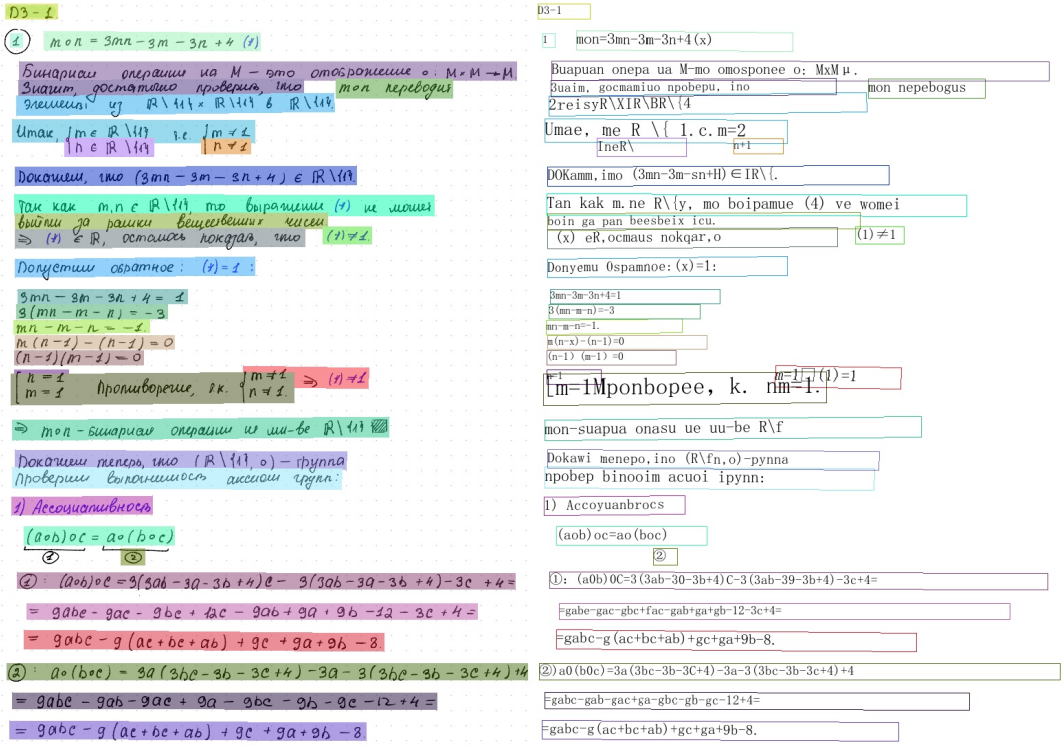


Figure 3: Визуализация детекции строк и последующей обработки с помощью `PaddleOCR`.

Тем не менее, одной лишь детекции строк оказалось недостаточно: требовалось объединять их в более крупные логические блоки текста. Для этого была предложена специальная метрика, основанная на вычислении относительной площади пересечения:

$$\text{metric} = \frac{\text{area}(B_{\text{base}} \cap B_{\text{other}})}{\min(\text{area}(B_{\text{base}}), \text{area}(B_{\text{other}}))}.$$

Если выполняется условие metric > 0.04, то соответствующие прямоугольники объединяются в один блок. Данный критерий показал себя лучше классического Intersection-over-Union (IoU), который часто некорректно работал в случаях сильно различающихся размеров строк.

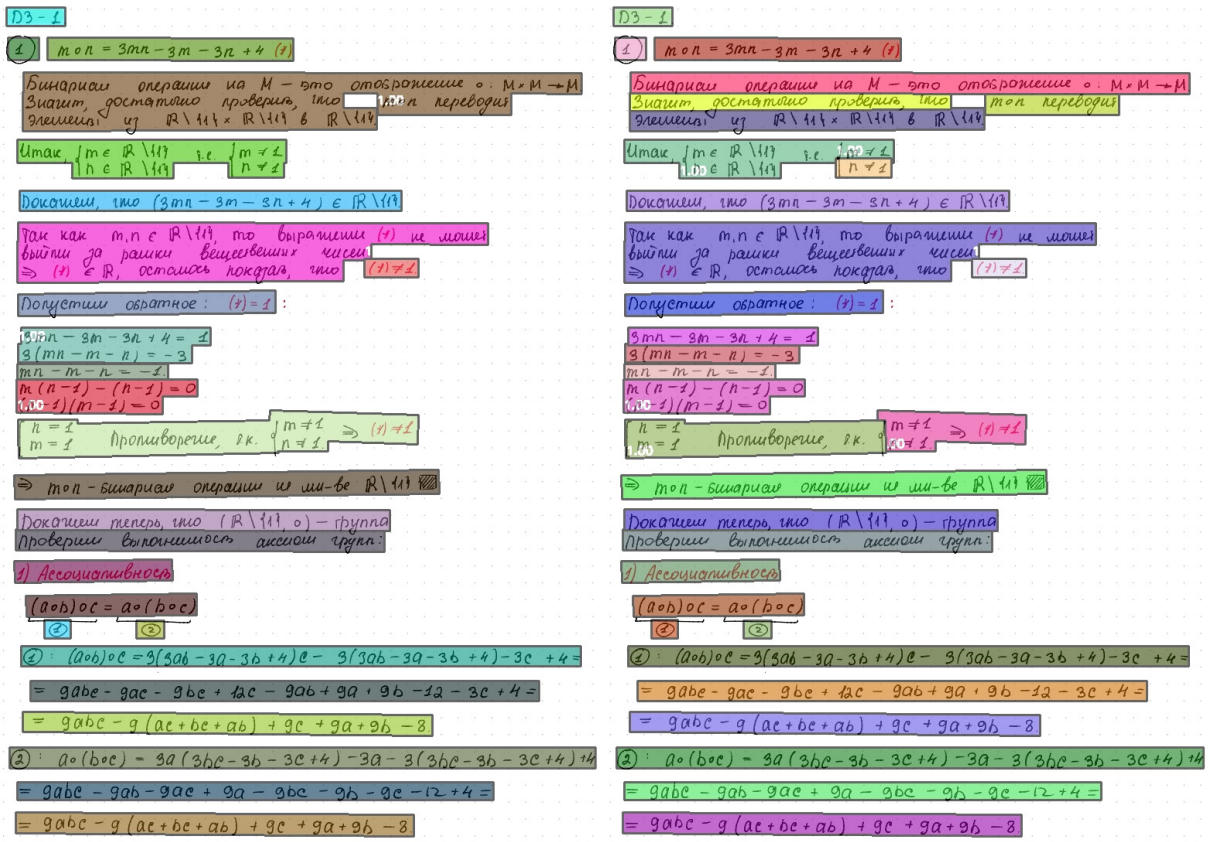
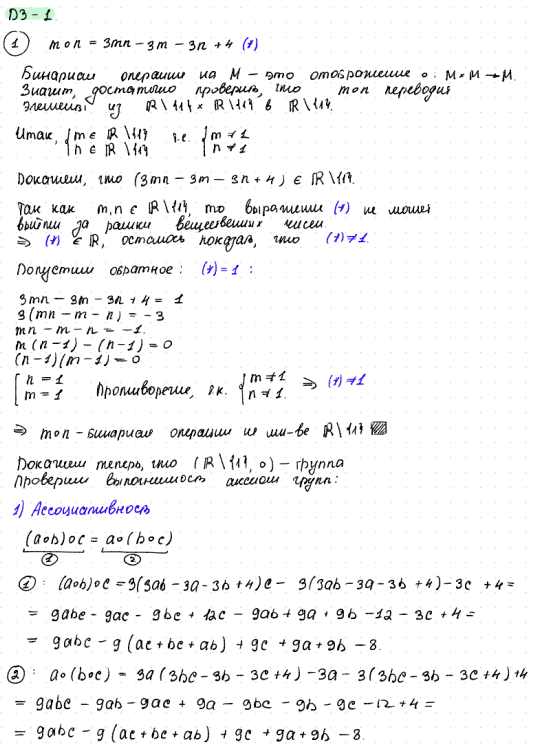


Figure 4: Сравнение объединения строк с использованием предложенной метрики и стандартного IoU.

После разбивания текст на блоки была также проверена гипотеза о отдельном выделении математических формул от рукописного текста. Каждая из них частей обрабатывалась специализированной моделью. Однако данный подход оказался неудачным, из-за очень низкой точности модели по распознаванию рукописного русского языка, например фразу 'Получим обратное' модель видела, как 'Полустии ображое'. Поэтому от этого подхода пришлось отказаться.

Окончательное решение строилось на последовательной обработке каждого блока текста с помощью квантизованной мультимодальной модели `Qwen2.5-VL-7B-Instruct-AWQ` [11]. Полученные результаты объединялись в единый текст, который затем последовательно дообрабатывался двумя языковыми моделями (`Qwen2.5-7B-Instruct-AWQ` [12]). Это обеспечивало как исправление ошибок, так и приведение результата к корректному LaTeX-формату.



D3 - 1.

① $m \circ n = 3mn - 3m - 3n + 4$ (*)

Бинарная операция на M — это отображение $\circ : M \times M \to M$. Значит, достаточно проверить, что $m \circ n$ переводит элементы из $\mathbb{R} \setminus \{1\} \times \mathbb{R} \setminus \{1\}$ в $\mathbb{R} \setminus \{1\}$.

Итак, $\begin{cases} m \in \mathbb{R} \setminus \{1\} \\ n \in \mathbb{R} \setminus \{1\} \end{cases}$ т.е. $\begin{cases} m \neq 1 \\ n \neq 1 \end{cases}$

Докажем, что $(3mn - 3m - 3n + 4) \in \mathbb{R} \setminus \{1\}$.

Так как $m, n \in \mathbb{R} \setminus \{1\}$, то выражение (*) не может выйти за рамки вещественных чисел.
$\Rightarrow$ (*) $\in \mathbb{R}$, остаётся показать, что $(*) \neq 1$.

Допустим обратное: $(*) = 1$:

$3mn - 3m - 3n + 4 = 1$
$3(mn - m - n) = -3$
$mn - m - n = -1$.
$m(n-1) - (n-1) = 0$
$(n-1)(m-1) = 0$
$\left[\begin{array}{l} n = 1 \\ m = 1 \end{array} \right.$ Противоречие, т.к. $\begin{cases} m \neq 1 \\ n \neq 1. \end{cases}$ $\Rightarrow$ (*) $\neq 1$

$\Rightarrow$ $m \circ n$ - бинарная операция на мн-ве $\mathbb{R} \setminus \{1\}$

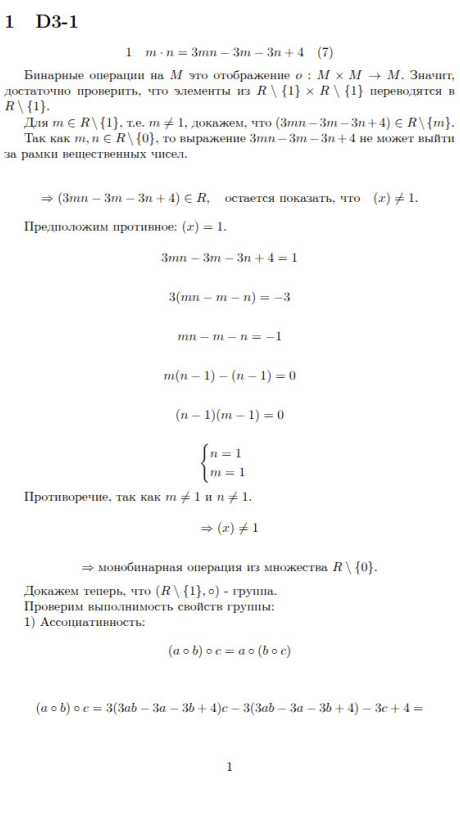
Докажем теперь, что $(\mathbb{R} \setminus \{1\}, \circ)$ — группа
Проверим выполнимость аксиом группы:

1) Ассоциативность

$\underbrace{(a \circ b) \circ c}_{①} = \underbrace{a \circ (b \circ c)}_{②}$

① : $(a \circ b) \circ c = 3(3ab - 3a - 3b + 4)c - 3(3ab - 3a - 3b + 4) - 3c + 4 =$
$= 9abc - 9ac - 9bc + 12c - 9ab + 9a + 9b - 12 - 3c + 4 =$
$= 9abc - 9(ac + bc + ab) + 9c + 9a + 9b - 8.$

② : $a \circ (b \circ c) = 3a(3bc - 3b - 3c + 4) - 3a - 3(3bc - 3b - 3c + 4) + 4$
$= 9abc - 9ab - 9ac + 9a - 9bc - 9b - 9c - 12 + 4 =$
$= 9abc - 9(ac + bc + ab) + 9c + 9a + 9b - 8.$



1 D3-1

1 $m \cdot n = 3mn - 3m - 3n + 4$ (7)

Бинарные операции на M это отображение $o : M \times M \to M$. Значит, достаточно проверить, что элементы из $R \setminus \{1\} \times R \setminus \{1\}$ переводятся в $R \setminus \{1\}$.

Для $m \in R \setminus \{1\}$, т.е. $m \neq 1$, докажем, что $(3mn - 3m - 3n + 4) \in R \setminus \{m\}$.

Так как $m, n \in R \setminus \{0\}$, то выражение $3mn - 3m - 3n + 4$ не может выйти за рамки вещественных чисел.

$\Rightarrow (3mn - 3m - 3n + 4) \in R$, остается показать, что $(x) \neq 1$.

Предположим противное: $(x) = 1$.

$$3mn - 3m - 3n + 4 = 1$$

$$3(mn - m - n) = -3$$

$$mn - m - n = -1$$

$$m(n-1) - (n-1) = 0$$

$$(n-1)(m-1) = 0$$

$$\begin{cases} n = 1 \\ m = 1 \end{cases}$$

Противоречие, так как $m \neq 1$ и $n \neq 1$.

$$\Rightarrow (x) \neq 1$$

$\Rightarrow$ монобинарная операция из множества $R \setminus \{0\}$.

Докажем теперь, что $(R \setminus \{1\}, \circ)$ - группа.
Проверим выполнимость свойств группы:
1) Ассоциативность:

$$(a \circ b) \circ c = a \circ (b \circ c)$$

$$(a \circ b) \circ c = 3(3ab - 3a - 3b + 4)c - 3(3ab - 3a - 3b + 4) - 3c + 4 =$$

1

Figure 5: Примеры работы финальной системы. Дополнительные результаты представлены в репозитории проекта.

Таким образом, проведённые эксперименты показали, что для успешного решения задачи необходимо сочетать детекцию на уровне строк с последующим объединением блоков и использованием каскада мультимодальных и языковых моделей. Прямое применение существующих парсеров или детекторов оказалось недостаточным, а разработанная схема позволила достичь наилучшего качества.

References

[1] Sofia Ares Oliveira, Benoit Seguin, and Frederic Kaplan. *dhSegment: A generic deep-learning approach for document segmentation*. IEEE, 2018.

[2] J. Bai et al. *Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond*. 2023.

[3] Lukas Blecher et al. *Nougat: Neural Optical Understanding for Academic Documents*. 2023.

[4] Cheng Cui et al. *PaddleOCR 3.0 Technical Report*. 2025. arXiv: 2507.05595 [cs.CV]. URL: https://arxiv.org/abs/2507.05595.

[5] Yuntian Deng, Anssi Kanervisto, and Alexander M. Rush. *Image-to-Markup Generation with Coarse-to-Fine Attention*. 2017.

[6] Yuning Du et al. *PP-OCR: A Practical Ultra Lightweight OCR System*. 2020.

[7] Geewook Kim, Teakgyu Hong, and et al. *Donut: Document Understanding Transformer without OCR*. 2022.

[8] Shilong Liu et al. *Grounding dino: Marrying dino with grounded pre-training for open-set object detection*. 2023.

[9] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. *Scaling Open-Vocabulary Object Detection*. 2023. arXiv: 2306.09683 [cs.CV].

[10] Zejiang Shen et al. *LayoutParser: A Unified Toolkit for Deep Learning Based Document Image Analysis*. 2021.

[11] Qwen Team. *Qwen2.5-VL*. Jan. 2025. URL: https://qwenlm.github.io/blog/qwen2.5-vl/.

[12] Qwen Team. *Qwen2.5: A Party of Foundation Models*. Sept. 2024. URL: https://qwenlm.github.io/blog/qwen2.5/.