Санкт-Петербургский государственный университет

Кафедра информационно-аналитических систем

Группа 22.Б11-мм

Исследование, анализ и разработка методов обнаружения и классификации структуры документов с применением моделей машинного обучения

ШМАКОВ Александр Александрович

Отчёт по учебной практике
в форме «Производственное задание»

Научный руководитель:
ассистент кафедры информационно-аналитических систем Чернышев Г. А.

Санкт-Петербург
2024

Оглавление

Введение

Написание выпускной квалификационной работы для студентов университетов имеет исключительное значение в их учебном процессе. Этот процесс предоставляет студентам возможность продемонстрировать усвоенные знания и умения, глубже погрузиться в конкретную область исследования, сформулировать новые идеи в предметной области, а также развить навыки самостоятельной работы и аналитического мышления. Кроме того, умение довести исследование до завершения в виде выпускной работы является важным опытом, который способствует развитию студентов как будущих специалистов в своей области.

Корректно выполненная академическая работа требует тщательной проверки на отсутствие грамматических и пунктуационных ошибок, а также соблюдение установленных стандартов оформления. Однако достижение такого результата требует значительных ресурсов и предполагает выполнение проверки текста в несколько этапов как студентом, так и его научным руководителем. Это требует затрат времени и усилий, которые могли бы быть направлены на улучшение содержания работы.

Приложение Mundane Assignment Police (MAP) [8], реализованное на языке программирования Kotlin [6], применяется для автоматизированной проверки курсовых и дипломных работ, и на данный момент MAP уже способен выявлять различные нарушения правил в тексте PDF-документов. Однако в настоящий момент работа приложения сопровождается большим количеством ложных срабатываний по причине неспособности обнаружить такие сущности в документах курсовых работ и ВКР, как листинги (изображения, таблицы, примеры кода, графики), и исключить их из зоны применения правил валидации текста. В ходе данной работы предлагается исследовать и разработать альтернативный способ обнаружения и классификации сущностей в структуре документов при помощи моделей машинного обучения.

1 Постановка задачи

Целью работы является разработка модели машинного обучения для выделения в структуре документов зон интереса, а также их классификации. Для достижения данной цели были поставлены следующие задачи:

1. Исследовать существующие подходы разработки моделей для анализа документов.
2. Провести обзор готовых наборов данных, подходящих для разработки моделей машинного обучения с использованием выбранных подходов.
3. Разработать модели машинного обучения с использованием выбранных наборов данных.
4. Сравнить результаты работы полученных моделей.

2 Обзор

В предыдущей [10] курсовой работе автором была предпринята попытка реализовать систему сегментирования листингов, содержащих строки кода, при помощи простых эвристик, однако такой метод не показал достаточного уровня эффективности в силу невозможности задания гибких правил обнаружения строк кода. На данный момент наиболее эффективным способом для сегментации докуметов на зоны интереса и последующей их классификации является разработка специализированных моделей машинного обучения.

Существует два основных подхода: с использованием моделей на базе LayoutLMv3 или с использованием моделей, натренированных с помощью библиотеки Detectron2.

2.1 LayoutLMv3

Компанией Microsoft было разработано три поколения моделей LayoutLM, из которых наиболее продвинутая — LayoutLMv3 [7]. Эта модель является одним из наиболее распространенных подходов в области Document AI/Document Understanding. Однако модели LayoutLM известны своей значительной вычислительной сложностью и ресурсозатратностью. Это связано с особенностями работы LayoutLMv3: сначала весь документ разбивается на сущности (токены), затем проходит через OCR-движок[1] для распознавания текста, после чего все токены классифицируются с использованием алгоритмов машинного обучения.

LayoutLMv3 использует архитектуру трансформер, что позволяет эффективно обрабатывать пространственные и текстовые данные одновременно. Это особенно важно для задач, связанных с анализом документов, таких как извлечение информации, классификация документов и распознавание форм. Модель также поддерживает мультимодальные входные данные, что позволяет ей учитывать как визуальные, так и текстовые признаки документа.

[1] Optical Character Recognition

Несмотря на высокую точность и производительность, модели LayoutLM требуют значительных вычислительных ресурсов для обучения и запуска на наборе тестовых данных. Это делает их применение более подходящим для крупных организаций с доступом к мощным вычислительным кластерам. Однако для проектов с ограниченным вычислительными ресурсами данный подход может быть не подходящим.

2.2 Detectron2

Библиотека Detectron2 [1] представляет собой удобную и качественную альтернативу моделям на базе LayoutLMv3. Процесс работы моделей, созданных с помощью Detectron2, сводится к обнаружению и классификации сущностей, найденных на изображении документа, без необходимости токенизации каждого слова и использования OCR-движка. Благодаря этому модели Detectron2 являются более легковесными и эффективными в использовании вычислительных ресурсов.

Detectron2 применяет передовые методы компьютерного зрения и глубокого обучения для выполнения задач, таких как детекция объектов, сегментация изображений и распознавание ключевых точек. Это делает библиотеку особенно полезной для анализа визуальных данных, где требуется высокая точность и производительность.

В отличие от LayoutLMv3, который использует токенизацию и OCR для обработки текстовых данных, Detectron2 фокусируется на анализе визуальных данных и применяет методы компьютерного зрения для обнаружения и классификации объектов. LayoutLMv3 использует архитектуру трансформеров для обработки пространственных и текстовых данных одновременно, что позволяет ей эффективно решать задачи, связанные с анализом документов, такие как извлечение информации и распознавание форм. Detectron2, в свою очередь, более подходит для задач, где требуется быстрая и эффективная обработка изображений.

Библиотека предоставляет свои реализации моделей Faster R-CNN и Mask R-CNN, являющихся основными инструментами для анализа структуры документов с выделением и классификацией зон интереса и

объектов, находящихся в них.

Для Faster/Mask R-CNN предоставлены базовые модели, основанные на трех различных комбинациях нейронных сетей:

- FPN (Feature Pyramid Network): Используется основа ResNet+FPN со стандартными сверточными[2] и полносвязными[3] слоями для предсказания масок и областей соответственно. Эта комбинация обеспечивает наилучшее соотношение скорости и точности.

- C4: Используется основа ResNet conv4 с внешним слоем (head) conv5[4]. Это оригинальная базовая модель, представленная в статье о Faster R-CNN.

- DC5 (Dilated-C5): Используется основа ResNet conv5 с дилатацией (расширением) в conv5[5] и стандартными сверточными и полносвязными слоями для предсказания масок и областей соответственно.

В дальшей работе будут использоваться реализации, основанные на FPN.

[2]Слои Convolutional Neural Netwrok. Эти слои применяют фильтры (или ядра) к входному изображению, чтобы извлечь различные признаки, такие как края, текстуры и другие детали.

[3]В таких слоях каждый нейрон связан со всеми нейронами предыдущего слоя.

[4]Это архитектура, где используется четвертый блок сверточных слоев ResNet (conv4) для извлечения признаков, а пятый блок (conv5) для окончательной обработки.

[5]Здесь используется пятый блок сверточных слоев ResNet (conv5) с дилатацией, что позволяет увеличить область восприятия без увеличения количества параметров.

3 Набор данных

Для тренировки моделей, предназначенных для классификации содержимого документов, существуют готовые наборы данных. В этой главе будут представлены наиболее значимые из них.

3.1 FUNSD

FUNSD [5] (Form Understanding in Noisy Scanned Documents) — это небольшой набор данных (199 изображений), содержащий множество классифицируемых сущностей структуры документа. Однако его размер слишком мал для того, чтобы модели машинного обучения могли быть эффективными на новых данных. Этот набор данных часто используется для задач, связанных с распознаванием форм и структурированных данных в отсканированных документах. Он включает в себя аннотации для текстовых блоков, заголовков, списков и других элементов, что делает его полезным для начального обучения моделей.

3.2 DocBank

DocBank [2] — это один из крупнейших наборов данных (500000 изображений) в области Document AI и Document Understanding, содержащий множество распознаваемых сущностей. Он включает в себя большое количество аннотированных документов, что делает его ценным ресурсом для обучения моделей. DocBank охватывает широкий спектр типов документов, включая научные статьи, отчеты и другие текстовые материалы, что способствует созданию универсальных моделей. Однако стоит отметить, что на данный момент указанный набор данных предоставлен авторами в непригодном для использования состоянии: нет значительной части тренировочных изображений, а также аннотаций.

3.3 DocLayNet

DocLayNet [3] — это набор данных среднего размера (80863 изображений), содержащий множество распознаваемых сущностей. Этот набор данных является хорошим компромиссом между размером, количеством распознаваемых сущностей и качеством аннотаций.

3.4 PubLayNet

PubLayNet [11] — это очень большой набор данных (360000 изображений, использовано более миллиона PDF-документов статей, находящихся в открытом доступе), содержащий все необходимые для распознавания сущности. Благодаря своему размеру, позволяет создавать более сложные и точные модели для автоматического анализа документов.

3.5 Вывод

В дальнейшей работе будут использованы наборы данных FUNSD, DocLayNet и PubLayNet. Модели, натренированные на вышеуказанных наборах данных, будут оценены и будет выбрана комбинация наиболее подходящего подхода тренировки и наиболее репрезентативного набора данных.

4 Метод

В данной части работы представлен обзор и описание разработанных моделей машинного обучения. Для всех перечисленных в данном разделе моделей были написаны коды для тренировки и тестирования.

4.1 LayoutLMv3

Для тонкой настройки (fine-tuning) модели LayoutLMv3 был использован набор данных FUNSD. Комбинация LayoutLMv3 и FUNSD не является случайной. FUNSD является набором данных наименьшего размера среди перечисленных в разделе 3 — "Набор данных", а модель LayoutLMv3, которая сочетает в себе возможности обработки естественного языка и компьютерного зрения, является более ресурсозатратной по сравнению с моделями на базе Detectron2. LayoutLMv3 использует трансформеры для анализа текстовых и визуальных данных, что позволяет модели эффективно распознавать и классифицировать сложные документы, однако это также сильно повышает количество необходимых вычислительных мощностей. Такая комбинация позволит приблизительно оценить потенциальные затраты ресурсов при масштабировании используемого набора данных.

4.2 Detectron2

Наборы данных DocLayNet и PubLayNet были использованы для настройки моделей предоставляемых библиотекой Detectron2. Изначально существовала и существует модель Fast R-CNN, выполняющая распознавание объектов и выделяющая границы зоны интереса (bounding boxes). Однако на данный момент эта архитектура является устаревшей. Есть два варианта усовершенствованной архитектуры данной модели:

- Faster R-CNN [4]
- Mask R-CNN [9]

4.2.1 Faster R-CNN

Усовершенствованная версия Fast R-CNN. При работе использует два больших модуля: Fast R-CNN и RPN[6]. Сильно ускоряет работу, благодаря замене модуля Selective Search в Fast R-CNN[7]. В Faster R-CNN его роль выполняет как раз модуль RPN, позволяющий избежать множественных запусков CNN на одном изображении.

4.2.2 Mask R-CNN

Расширение Faster R-CNN — предсказывает положения маски, покрывающей найденный объект внутри регионов интереса. Может быть полезно для дальнейшего более точного определения положения листингов в документах курсовых работ и ВКР, чтобы исключить их из зоны действия правил валидации текста.

[6]Region Proposal Network

[7]Selective Search в Fast R-CNN отвечает за наложение границ зон интереса на изображение, путем итерационного запуска CNN и масштабирования размеров границ зон интереса в соответствии с размером обработанных изображений

5 Эксперимент

Были проведены замеры результатов работы всех натренированных моделей. Для оценки качества готовых моделей были использованы метрики Accuracy, Precision и Recall:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

- $TP\,(TruePositive)$ — количество верно классифицированных положительных примеров
- $TN\,(TrueNegative)$ — количество верно классифицированных отрицательных примеров
- $FP\,(FalsePositive)$ — количество ложно классифицированных положительных примеров
- $FN\,(FalseNegative)$ — количество ложно классифицированных отрицательных примеров.

Для поставленной задачи наиболее важной метрикой в данном наборе является метрика Recall. Высокий показатель данной метрики означает, что модель способна обнаружить большое количество необходимых для обнаружения сущностей классов. Результаты измерения метрик моделей будут приведены в конце раздела.

Для тренировки и тестирования моделей был использован ноутбук со следующими параметрами:

- Процессор AMD Ryzen 7 5800H
- Видеокарта NVIDIA GeForce RTX 3070 Laptop GPU
- Windows Subsystem for Linux — Ubuntu jammy 22.04 x86_64

5.1 LayoutLMv3 + FUNSD

Соотношение тестового набора данных к тренировочному составляет 50:149, то есть примерно 25% и 75% соответственно. Несмотря на небольшой размер набора данных, модель продемонстрировала высокие показатели метрик благодаря принципу работы модели LayoutLMv3. Однако, из-за этого модель получилась большого размера (500 МБ при размере набора данных в 18 МБ) и очень ресурсозатратной, чего проект не может себе позволить. Поэтому было принято решение отказаться от тренировки основной модели на более подходящем наборе данных с использованием LayoutLMv3. Найти готовую модель можно по ссылке[8] на репозиторий в Hugging Face.

5.2 Detectron2/Faster R-CNN + DocLayNet

Соотношение тестового набора данных к тренировочному составляет 11488:69375, то есть примерно 14% и 86% соответственно. Для тренировки модели потребовалось 4 часа. Модель, натренированная с использованием Detectron2 и набора данных DocLayNet, показала довольно хорошие результаты. Однако, несмотря на эти результаты, модель значительно уступает по эффективности и качеству работы на новых данных моделям, натренированным с использованием Detectron2 и датасета PubLayNet. Это связано с тем, что PubLayNet является более крупным и разнообразным датасетом с меньшим количеством распознаваемых классов объектов. По этой причине модель с использованием данного датасета также не будет использоваться в дальнейшей работе. Готовая модель доступна по ссылке[9] на репозиторий в Hugging Face.

5.3 Detectron2/Faster R-CNN + PubLayNet

Соотношение тестового набора данных к тренировочному составляет 11488:69375, то есть примерно 14% и 86% соответственно. Для тре-

[8] https://huggingface.co/AlexShmak/LayoutLMv3_FUNSD
[9] https://huggingface.co/AlexShmak/Detectron2_DocLayNet

нировки модели потребовалось 3,45 часа. Данная модель превзошла результаты работы на новых данных модели, основанной комбинации Detectron2/Faster R-CNN и DocLayNet, при одинаковых тренировочных конфигурациях.

5.4 Detectron2/Mask R-CNN + PubLayNet

Использованы те же разделы тренировочного и тестового наборов данных. Для тренировки модели потребовалось 6 часов. Показала результаты сравнимые с моделью "Detectron2/Faster R-CNN + PubLayNet". Однако потребовала больше времени для тренировки и тестирования из-за дополнительного обнаружения масок зон интереса.

5.5 Результаты измерения метрик моделей

Model	Dataset	Average Accuracy	Average Precision	Average Recall
LayoutLMv3	FUNSD	0.91	0.87	0.89
Faster R-CNN	DocLayNet	0.86	0.82	0.79
Faster R-CNN	PubLayNet	0.93	0.91	0.88
Mask R-CNN	PubLayNet	0.92	0.9	0.86

По итогам эксперимента модели, натренированные с использованием набора данных PubLayNet, являются наиболее продвинутыми моделями, основанными на моделях из библиотеки Detectron2. Модели, основанные на датасете PubLayNet, а также моделях Faster R-CNN и Mask R-CNN показали сопоставимое друг другу качество работы по результатам вычисления метрик. С данными моделями будет проводиться вся дальнейшая работа.

Заключение

В рамках работы были достигнуты следующие результаты:

- Проведен обзор существующих подходов для решения проблемы сегментации документов
- Выделены наиболее подходящие наборы данных, предоставляющие данные для тренировки эффективной модели
- Разработано несколько моделей машинного обучения с использованием разных комбинаций походов и наборов данных, а также выделены модели наиболее подходящие для выполнения поставленной задачи
- Произведено сравнение результатов полученных моделей

В дальнейшем планируется произвести повторную тренировку наиболее эффективной модели на наборе данных, собранном из курсовых работ и выпускных квалификационных работ, доступных на сайте кафедры Системного Программирования. Код для тренировки основных моделей, а также сами модели доступны по ссылке[10] на репозиторий в Hugging Face.

[10] https://huggingface.co/AlexShmak/document_understanding

Список литературы

[1] Wu Yuxin, Kirillov Alexander, Massa Francisco et al. Detectron2. — https://github.com/facebookresearch/detectron2. — 2019.

[2] Li Minghao, Xu Yiheng, Cui Lei et al. DocBank: A Benchmark Dataset for Document Layout Analysis. — 2020. — 2006.01038.

[3] DocLayNet: A Large Human-Annotated Dataset for Document-Layout Segmentation / Birgit Pfitzmann, Christoph Auer, Michele Dolfi et al. // Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. — KDD '22. — ACM, 2022. — . — P. 3743–3751. — URL: http://dx.doi.org/10.1145/3534678.3539043.

[4] Ren Shaoqing, He Kaiming, Girshick Ross, Sun Jian. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. — 2016. — 1506.01497.

[5] Jaume Guillaume, Ekenel Hazim Kemal, Thiran Jean-Philippe. FUNSD: A Dataset for Form Understanding in Noisy Scanned Documents. — 2019. — 1905.13538.

[6] Kotlin. — URL: https://kotlinlang.org/.

[7] Huang Yupan, Lv Tengchao, Cui Lei et al. LayoutLMv3: Pre-training for Document AI with Unified Text and Image Masking. — 2022. — 2204.08387.

[8] Makeev Vladislav. Map: Web-app that assists in checking students' assignments. — https://github.com/Darderion/map. — URL: https://github.com/Darderion/map.

[9] He Kaiming, Gkioxari Georgia, Dollár Piotr, Girshick Ross. Mask R-CNN. — 2018. — 1703.06870.

[10] Shmakov Alexander. CodeDetector for MAP. — https://github.com/Darderion/map/pull/68. — 2023. — URL: https://github.com/Darderion/map/pull/68.

[11] Zhong Xu, Tang Jianbin, Yepes Antonio Jimeno. PubLayNet: largest dataset ever for document layout analysis. — 2019. — 1908.07836.