УДК 004.89

РАЗРАБОТКА МЕТОДИКИ ИЗВЛЕЧЕНИЯ ИНФОРМАЦИИ ОБ УСЛОВИЯХ ХИМИЧЕСКОЙ РЕАКЦИИ ИЗ ТЕКСТА НА ИЗОБРАЖЕНИИ

Скреденас Д.А., Новикова О.А.

Федеральное государственное бюджетное образовательное учреждение высшего образования «МИРЭА — Российский технологический университет», 119454, Российская Федерация, г. Москва, пр-т Вернадского, 78, e-mail: tehnik145@yandex.ru, ol-novikova@bk.ru

В работе рассматривается методика извлечения ключевой информации об условиях химической реакции из неструктурированного текста, расположенного на иллюстрациях к научным статьям. Данная методика позволяет ускорить процесс получения и структуризации данных о синтезе веществ, приводимых в научных статьях. Для решения поставленной задачи был разработан модуль, выполняющий распознавание текста на изображении, а также выявление и классификацию параметров реакции в распознанном тексте при помощи нейронных сетей. Для сокращения объёма целевых данных, требуемого для обучения модели распознавания текста, был создан генератор синтетических изображений и меток к ним. В этих же целях был применён подход предобучения модели распознавания сущностей на большом наборе размеченных химических патентов, размещённом в открытом доступе. При обучении модели распознавания текста были использованы аугментации входных изображений для моделирования различных особенностей в целевых данных, увеличения объёма обучающего набора данных, повышения его разнородности, а также улучшения обобщающей способности модели. Предложен модифицированный алгоритм получения векторного представления текста в модели BERT для учёта словесной информации при использовании символьных токенов. После обучения моделей было проведено развёртывание и тестирование модуля, выполнены замеры производительности и объёма потребляемых ресурсов.

Ключевые слова: машинное обучение, глубокое обучение, извлечение ключевой информации, оптическое распознавание символов, распознавание именованных сущностей, предобучение, синтетические данные.

DEVELOPMENT OF A METHODOLOGY FOR EXCTRACTING CHEMICAL REACTION CONDITIONS FROM TEXT WITHIN IMAGE

Skredenas D.A., Novikova O.A.

Federal State Budget Educational Institution of Higher Education «MIREA – Russian Technological University», 119454, Russian Federation, Moscow, Vernadsky pr., 78, e-mail: tehnik145@yandex.ru, ol-novikova@bk.ru

This paper discusses a methodology for extracting key information regarding the conditions of a chemical reaction from unstructured textual data present in illustrations within scientific articles. The proposed methodology aims to expedite the process of acquiring and organizing data on the synthesis of compounds presented in scientific literature. To solve this task, a module was developed that performs text recognition in an image, as well as identification and classification of reaction parameters in the recognized text using neural networks. In order to reduce the amount of labeled data necessary for training a robust text recognition model, an application for generating synthetic images and corresponding labels has been created. For the same purpose a pre-training strategy has been applied for named entities recognition model by utilizing a large publicly available dataset of chemical patents. During training of the text recognition model, input image augmentations were used to simulate various features in the target data, increase the size of the training set, increase its variety, and enhance generalizability of the model. A modified algorithm of BERT embeddings extraction was proposed to incorporate verbal information when using character-level tokenization. After training the models, the module was deployed and tested, performance and resource consumption measurements were performed.

Keywords: machine learning, deep learning, key information extraction, optical character recognition, named entities recognition, pre-training, synthetic data.

Введение

Химическая отрасль является одной из самых важных и быстроразвивающихся областей науки и промышленности. Открытия, сделанные учёными-химиками, приносят пользу в фармацевтической, косметической, промышленной, сельскохозяйственной и многих других областях.

При разработке новых химических веществ исследователи часто обращаются к научным статьям в изучаемой области. Исследовательский этап является одним из самых важных и трудоёмких: среди огромного количества страниц и рассуждений сложно найти точную эмпирическую информацию, поэтому статьи приходится долго изучать, вчитываться в них. Не последнюю роль в этом процессе играет человеческий фактор: многочасовой, каждодневный процесс изучения может сильно утомить человека, понизить его внимательность и концентрацию, что в конечном итоге приводит не только к замедлению процесса исследования, но и к ошибкам в извлечённой информации.

Для ускорения и упрощения работы исследователей в области химии могла бы помочь аналитическая платформа, выполняющая извлечение основной информации из научных статей и её агрегацию. К такой информации, в частности, относится описание условий прохождения химической реакции, без знания которых практически невозможно воспроизведение результатов синтеза веществ: реакция может не пройти вовсе, либо дать результаты, не имеющие ничего общего с полученными в статье.

Стоит учесть следующую особенность: зачастую условия реакций изображены на иллюстрациях, размещённых в тексте статьи. В случае человеческой обработки это усложняет извлечение необходимой информации: исследователю придётся вручную перепечатывать нужную информацию с иллюстрации, что дополнительно повышает риск опечаток и ошибок. Однако для компьютерной программы такая особенность не является проблемой: современные достижения в области распознавания текста позволяют работать с изображениями напрямую.

Целью данной работы является разработка модуля, который из изображения текстового описания условий химической реакции извлечёт числовые и категориальные показатели, характеризующие эту реакцию. При этом необходимо подготовить инфраструктуру модуля и выполнить его развёртывание для последующего встраивания во внешнюю аналитическую платформу, отвечающую за извлечение и агрегацию информации из загружаемых в неё научных статей.

Постановка задачи

В научных публикациях химические реакции описываются при помощи иллюстраций. На этих иллюстрациях реагенты изображены в виде структурных формул, стрелкой обозначено направление протекания реакции, а рядом со стрелкой указываются условия, при которых проводилась реакция. На рис. 1 представлен пример такой иллюстрации.

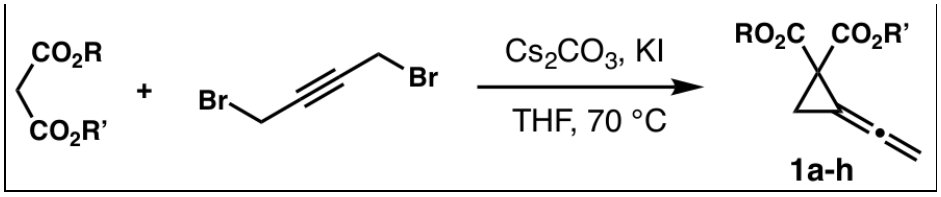


Рисунок 1. Пример иллюстрации, описывающей химическую реакцию

Детектор, входящий в состав ранее упомянутой аналитической платформы, извлекает изображение условий над и под стрелкой и передаёт рассматриваемому в данной работе модулю в качестве входных данных. Пример поступающего с детектора изображения представлен на рис. 2.

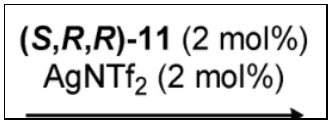


Рисунок 2. Пример входного изображения

Ответ модуля должен представлять собой набор пар, состоящих из текстового фрагмента (параметр реакции) и его класса (тип параметра). На рис. 3 приведён пример ответа для изображения на рис. 2.

```
[{'word': '(S,R,R){\\text{-}}11', 'label': 'reactant_link'},
 {'word': '2\\;mol\\%', 'label': 'molarity'},
 {'word': 'AgNTf{}_2', 'label': 'catalyst'},
 {'word': '2\\;mol\\%', 'label': 'molarity'}]
```

Рисунок 3. Пример извлечённой модулем информации об условиях реакции

Процесс извлечения информации был разделён на два этапа: распознавание текста на входном изображении и выявление параметров реакции в распознанном тексте. Под выявлением параметров понимается извлечение текстовых фрагментов, описывающих реакцию, и их классификация на одну из заранее заданных категорий (примеры категорий: "solvent" — растворитель, "catalyst" — катализатор, "molarity" — количество молей вещества, "time" — время проведения реакции). Отметим, что условия записываются в статьях при помощи системы визуализации математических формул "LaTeX", поэтому, для сохранения верхних и нижних индексов и учёта специальных математических символов, текст также распознаётся в нотации LaTeX.

В области машинного обучения существуют направления, соответствующие двум озвученным выше этапам: задача оптического распознавания символов (Optical Character Recognition, OCR) и задача распознавания именованных сущностей (Named Entity Recognition, NER). Поскольку для решения каждой из этих задач используются различные архитектуры и методы, разработанный нами модуль разделён на две отдельные модели OCR и NER, соединённые последовательно.

Для того, чтобы определиться с архитектурой моделей, входящих в состав модуля, а также с подходами к их обучению, рассмотрим подробнее решения, которые существуют в областях OCR и NER.

Обзор существующих решений

В области OCR широкое распространение получила нейросетевая архитектура CNN-RNN [1]. Наличие свёрточной нейронной сети в составе модели позволяет ей самостоятельно сформировать пространство репрезентативных визуальных признаков. А благодаря задействованию рекуррентной нейросети модель может обрабатывать изображения любой длины (при фиксированной высоте).

В [2] авторы отказались от использования рекуррентных нейросетей, поскольку они замедляют обучение модели, а наличие обратной связи между нейронами приводит к увеличению глубины графа вычислений и повышенному использованию видеопамяти соответственно. Предложенная архитектура Gated CNN использует свёрточную нейронную сеть и блок фильтрации информативных признаков.

Также авторы используют аугментации входных изображений. Аугментации — это некоторые действия, которые изменяют входные данные (например, поворот и сдвиг изображения), но при этом сохраняют истинность метки. Это даёт возможность искусственно увеличить количество данных и повысить их разнородность. Существующие исследования демонстрируют, что использование аугментаций позволяет улучшить обобщающую способность модели и повысить качество предсказаний на данных, не встречавшихся в обучающей выборке [3].

Оба предыдущих подхода не позволяют распознавать многострочный текст без предварительной сегментации на отдельные строки. Для решения этой проблемы в [4] предлагается архитектура CNN-Transformer: вместе со свёрточной нейросетью применяется трансформерная модель, которая при распознавании текста использует информацию обо всех участках изображения сразу.

В [5] авторы предлагают архитектуру TrOCR. В ней трансформер используется как для предсказания символов, так и для извлечения визуальных признаков, поскольку визуальным трансформерам удаётся получить более высокие результаты в сравнении с глубокими свёрточными нейросетями в задачах обработки изображений. Это также даёт возможность инициализировать модель размещёнными в открытом доступе весами визуальных трансформеров и языковых моделей.

На основе проведённого обзора OCR моделей было решено использовать для распознавания текста архитектуру TrOCR. Благодаря наличию в открытом доступе большого числа обученных трансформерных моделей, появляется возможность выполнить инициализацию параметров нейросети, что сократит время обучения и упростит его из-за наличия у модели начальных знаний об изображениях и естественном языке. Также при обучении выбранной архитектуры следует воспользоваться операцией аугментации входных изображений для повышения обобщающей способности модели.

Теперь перейдём к рассмотрению существующих моделей распознавания сущностей в тексте. Поскольку текст представляет собой связную последовательность переменной длины, в задаче NER широко применяются рекуррентные нейронные сети. Так, в [6] авторы используют архитектуру BiLSTM-CRF. Входной слой модели

состоит из обучаемых репрезентаций слов и символов, благодаря чему она самостоятельно формирует информативные признаковые представления входных текстов. Также авторы используют слой "attention", который при помощи механизма внимания позволяет учитывать взаимосвязи между словами и контекст их употребления.

В [7] авторы подтверждают состоятельность механизмов внимания в задачах обработки текста и предлагают использовать трансформерную модель BERT [8]. Однако авторы установили, что BERT показывает невысокие результаты в узкоспециализированных задачах биомедицины и хемоинформатики, поскольку она была обучена на текстах общей тематики. Поэтому авторы проводили калибровку модели в три этапа: инициализация обученными весами BERT (получение общих знаний о языке), предобучение на текстах предметной области, дообучение на целевых данных.

В [9] трансформерная языковая модель BERT дополнена рекуррентной нейросетью для извлечения «локальных» признаков — информации о взаимосвязях между близкорасположенными словами. В полученной архитектуре HGN также используются блок внимания, который объединяет контекстные и локальные признаки, и однослойный перцептрон, выполняющий предсказание на основе признаковых векторов.

В результате для распознавания сущностей было решено использовать архитектуру HGN, поскольку она учитывает, как контекст данных на глобальном уровне, так и взаимосвязи между словами и их взаимное расположение на локальном уровне. При этом появляется возможность снабдить трансформерный кодировщик базовыми знаниями о языке, инициализировав его обученными весами BERT, что должно ускорить и упростить обучение всей модели.

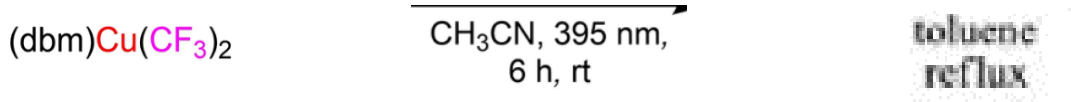
Также стоит проводить калибровку модели в два этапа: предобучение на большом корпусе текстов про химические реакции (что позволит сформировать базовое представление о предметной области и о роли различных химических элементов в реакциях) и дообучение на целевых данных. При этом за счёт наличия фазы предобучения потребуется меньший объём целевых данных, что приведёт к сокращению времени на их разметку.

Реализация модели OCR

Перед применением модели TrOCR её необходимо обучить на парах «изображение, текст на изображении». Для калибровки глубокой нейронной сети требуется обучающий набор данных большого объёма, а ручной сбор такого количества данных является трудоёмким процессом. Поэтому калибровка модели выполнялась в два этапа: предварительное формирование признакового пространства при помощи большого количества синтетических данных и последующее дообучение на небольшом, вручную подготовленном наборе целевых данных.

Для формирования синтетических данных была написана программа, которая генерирует случайную текстовую последовательность (включающую в себя LaTeX-конструкции, которые могут встретиться в реальных данных, такие как индексы, математические символы и пр.) и визуализирует её. Таким образом, полученные изображения становятся входными данными, а сгенерированный текст — метками к ним.

Во время обучения к входным изображениям применялись аугментации. Набор аугментаций подбирался с учётом особенностей реальных входных изображений (рис. 4).



а) б) в)

Рисунок 4. Примеры особенностей входных изображений: а) цветной текст; б) попадание в кадр посторонних объектов; в) низкое качество, зашумлённость

Для обработки случаев, представленных на рис. 5, применялись аугментации обесцвечивания изображения, добавления постороннего содержимого, сжатия картинки и добавления шума. Пример синтетического изображения, полученного в результате применения аугментаций, представлен на рис. 5:

Рисунок 5. Пример работы аугментаций

Перед обучением трансформерный кодировщик модели был инициализирован весами визуального трансформера Swin Transformer V2 [10]. В качестве декодировщика используется модель BERT. Реализация модели и загрузка весов осуществлялись при помощи Python-библиотеки Transformerms [11]. Обучение модели проводилось по алгоритму градиентного спуска с использованием кросс-энтропии в качестве функции ошибки (1):

$$\mathrm{CE} = -\sum_{i=1}^{N}\sum_{j=1}^{V} y_{ij}\log\left(\hat{y}_{ij}\right), \tag{1}$$

где N — количество символов во входном тексте;

V — размер словаря символов;

y_{ij} — истинная вероятность того, что на i-ой позиции в тексте расположен j-ый символ из словаря (для истинного символа вероятность равна единице, для остальных — нулю);

$\hat{y}_{ij}$ — предсказанная моделью вероятность того, что на i-ой позиции в тексте расположен j-ый символ из словаря.

Для оптимизации функции ошибки использовался алгоритм Adam [12], обновляющий параметры модели по следующей формуле (2):

$$\begin{gathered} w_t = w_{t-1} - \eta\frac{m_t}{\sqrt{v_t}+\varepsilon}, \\ m_t = \alpha m_{t-1} + (1-\alpha)g_t, (m_0 = g_0), \\ v_t = \beta v_{t-1} + (1-\beta)g_t^2, (v_0 = g_0^2), \end{gathered} \tag{2}$$

где g_t — значение градиента функции ошибки по параметрам модели в момент времени t;

m_t — экспоненциальное скользящее среднее градиента;

v_t — экспоненциальное скользящее среднее квадрата градиента;

α и β — гиперпараметры, регулирующие влияние предыдущих значений скользящего среднего на последующие;

ε — некоторая малая положительная величина, введённая во избежание деления на ноль в случае, когда значение v_t равно нулю;

η — скорость обучения.

В алгоритме Adam выполняется нормировка вектора градиента, благодаря чему скорость перемещения в пространстве весов стабилизируется, а использование скользящих средних делает изменение параметров инерционным, что позволяет избежать попадания весов в точку локального минимума функции ошибки.

Оценка качества работы модели выполнялась при помощи метрики Accuracy на уровне всего текста: она равна отношению количества полностью верно распознанных текстов к количеству изображений в проверочной выборке. Итоговое значение метрики Accuracy составило 0,933. Также в качестве вспомогательной метрики использовалась метрика Macro accuracy (3) на уровне символов, равная усреднённой точности распознавания каждого уникального символа в проверочном наборе данных:

$$\mathrm{MA} = \frac{1}{V}\cdot\sum_{i=1}^{N} 1_{x=y_i}(p_i), \tag{3}$$

где V — количество символов в словаре модели;

N — общее количество символов в тексте;

p_i — символ на i-ой позиции, предсказанный моделью;

y_i — истинный символ на i-ой позиции;

$1_{x=y_i}(p_i)$ — функция-индикатор, равная единице, если p_i равен y_i, и нулю в противном случае.

Итоговое значение метрики Macro accuracy составило 0,949.

На рис.6 представлен пример работы полностью обученной модели OCR. В верхнем левом углу расположено входное изображение, в нижнем левом углу— его метка, в нижнем правом — распознанный моделью текст в нотации LaTeX, а в верхнем правом — визуализация распознанного текста (для удобства сравнения с входным изображением).

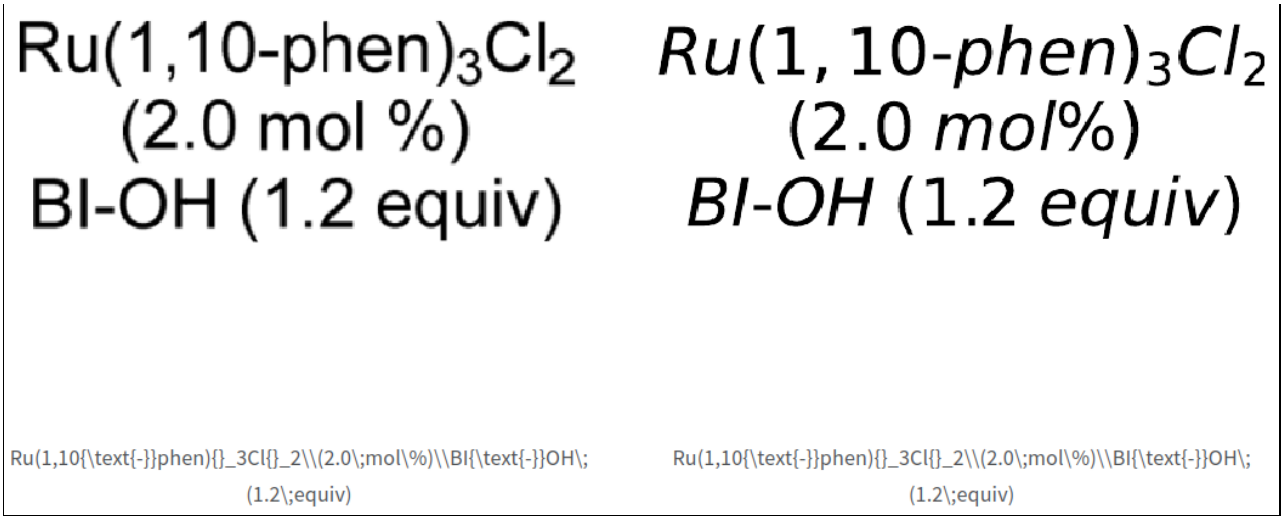


Рисунок 6. Пример распознавания текста на входном изображении

Как видно из рисунка 6, модель OCR успешно справляется с распознаванием последовательностей, состоящих из большого количества символов и содержащих LaTeX-конструкции. Теперь перейдём к рассмотрению процесса разработки модели распознавания сущностей.

Реализация модели NER

Классический подход к задаче NER предполагает предварительное разбиение входного текста по пробелам, поскольку это позволяет корректно определить границы именованных сущностей. Полученные в результате слова разбиваются на токены. Токен — это текстовый фрагмент (символ, часть слова или слово целиком), которому сопоставляется некоторое число. Разбиение входного текста на токены необходимо, поскольку нейронные сети работают именно с числами, а не с текстовыми данными. Токены проходят через обучаемые слои, ответственные за получение векторного представления текста (т.н. «слои эмбеддингов»): в них числа отображаются в векторы фиксированной размерности, описывающие токен во внутреннем представлении модели. Оперируя этими векторами, модель может выучиваться смыслу слов и взаимосвязям между ними в рамках текста.

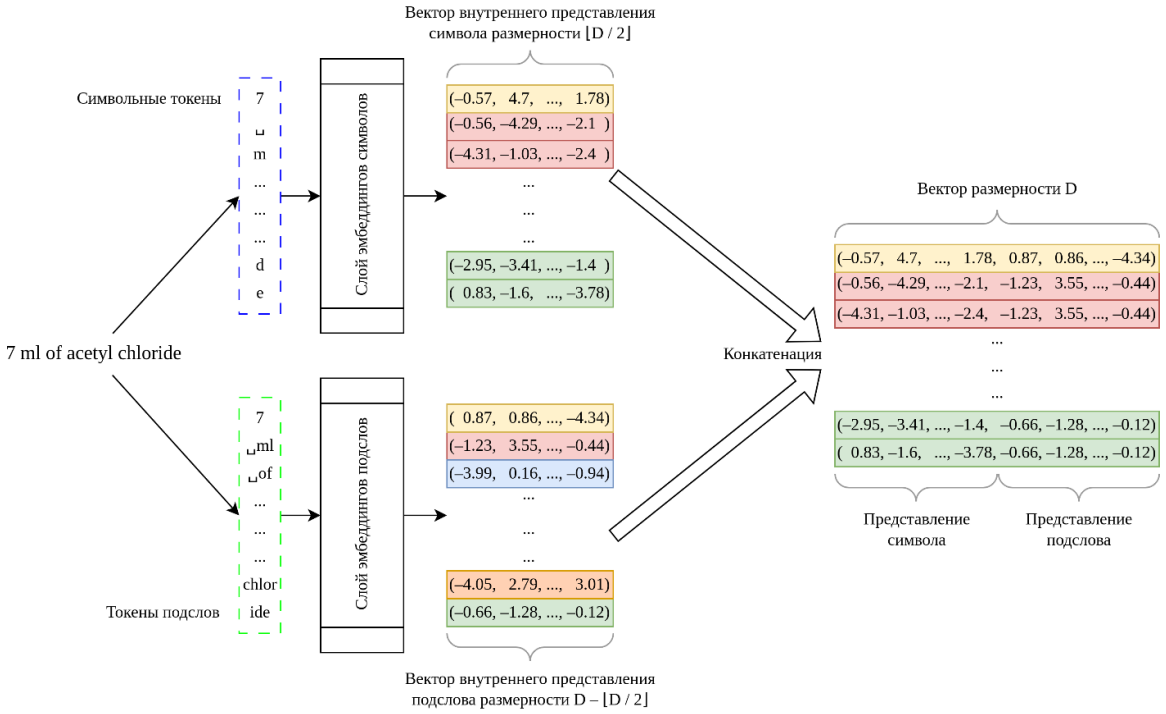


Рисунок 7. Иллюстрация модифицированного процесса получения внутреннего представления входного текста

Однако модель OCR не обладает абсолютной точностью и может не распознать некоторые пробелы, что приведёт к наличию «граничных» токенов, принадлежащих к двум сущностям одновременно, и невозможности однозначного определения границ сущностей. Для решения этой проблемы входные тексты разбивались на отдельные символы (символьный токен не может принадлежать к двум словам одновременно), и определение типа параметра условий выполнялось при помощи классификации каждого из полученных символьных токенов на одну из заранее заданных категорий именованных сущностей.

При таком подходе модели трудно выработать универсальное смысловое представление отдельного символа, а не слова целиком или его фрагмента, теряется часть информации. Решением этой проблемы может стать сопоставление токенам символов токенов подслов, к которым они относились. Для этого архитектура модели BERT, входящей в состав HGN, была модифицирована: один слой извлечения векторных представлений был заменён на два — слой, предназначенный для символов, и слой, предназначенный для подслов. Получаемые в результате векторы внутренних представлений конкатенируются. Описанная модификация представлена на рис. 7, где D — размерность пространства внутреннего представления модели.

Так же, как и в случае с моделью TrOCR, калибровка параметров модифицированной модели NGH выполнялась в два этапа: предобучение на большом наборе химических патентов USPTO [13] для формирования представления о предметной области и химических терминах и последующее дообучение на целевых данных. Тексты в обучающем наборе данных размечались следующим образом.

Первым символам незначащих сущностей (не являющихся параметрами реакции) присваивался класс "O", затем первым символам сущностей, являющихся параметрами реакции, присваивался класс, соответствующий типу параметра (всего рассматривается 29 типов), а остальным (внутренним) символам присваивался класс "I". Итого задействован 31 класс. Такой способ разметки отличается от широко применяемой в задаче NER нотации BIO [14], при использовании которой дополнительно вводятся классы для обозначения внутренних символов каждого из типов параметров, что практически удваивает число классов и существенно затрудняет обучение модели.

Для имитации случая утери пробела моделью OCR, к входным данным применялась аугментация: каждый из пробелов во входной последовательности удалялся с вероятностью 10%.

Также, как и при обучении модели OCR, в качестве функции ошибки использовалась функция кросс-энтропии (1), однако с иными обозначениями: V — количество категорий сущностей (всего 31), y_{ij} — истинная вероятность того, что i-ый символ входного текста принадлежит к сущности j-ой категории (для истинной категории вероятность равна единице, для остальных — нулю), $\hat{y}_{ij}$ — предсказанная моделью вероятность того, что на i-ый символ входного текста принадлежит к сущности j-ой категории.

Качество распознавания сущностей оценивалось при помощи метрики Macro accuracy (3) на уровне классов сущностей, т.е. при помощи усреднённой точности распознавания каждого из 31 видов параметров реакции в проверочном наборе данных. Итоговое значение метрики составило 0,885. Для получения информации в виде, представленном на рис. 3, выполняется считывание символов входного текста с учётом предсказанных для них классов: при встрече символа, классифицированного как начало значащей сущности, считываем его и все последующие символы до тех пор, пока не встретится начало другой сущности или конец строки. Ответ модели NER для входного текста "Pd(PPh{}_3){}_2Cl{}_2\\;(5\\;mol\\%)\\\\2M\\;aq.\\;K{}_2CO{}_3,\\;reflux" представлен на рис. 8.

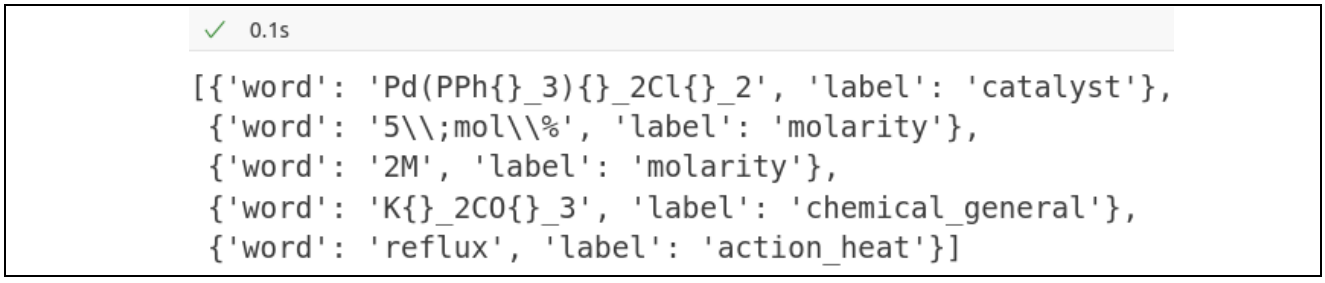


Рисунок 8. Последовательность предсказанных моделью именованных сущностей и их классов

Как видно из рис. 8, модель NER успешно справляется с выявлением и классификацией сущностей в тексте, записанном в LaTeX-нотации: действительно, $Pd(PPh_3)_2Cl_2$ является катализатором, K_2CO_3 — реагентом, "5 mol%" и "2M" — обозначениями молей, а "reflux" — это название процесса контролируемого нагрева вещества при постоянной температуре.

Теперь перейдём к рассмотрению процесса развёртывания описанных выше моделей в виде итогового модуля.

Создание инфраструктуры модуля

После обучения и оценки качества моделей было выполнено их развёртывание в виде двух изолированных подмодулей при помощи инструмента контейнерной виртуализации Docker [15]. Такое разделение моделей позволяет избежать конфликтов версий программных библиотек, а также обеспечивает возможность балансировки нагрузки: например, при высокой потребности в распознавании текста без выявления сущностей, инструменты балансировки могут увеличить число соответствующих контейнеров без увеличения количества моделей NER.

Тестирование работы модуля выполнялось следующим образом: программа-маршрутизатор при помощи Python-библиотеки Requests [16] отправляет в подмодуль OCR HTTP-запрос с полезной нагрузкой в виде изображения в кодировке Base64. Внутри контейнера работает веб-сервер Uvicorn [17], который принимает этот запрос. Программный интерфейс под управлением FastAPI [18] передаёт данные из запроса в функцию-обработчик, в которой изображение декодируется, и при помощи модели OCR, работающей внутри контейнера, выполняется распознавание текста. Распознанный текст в виде HTTP-ответа возвращается исходной программе-маршрутизатору, после чего она отправляет HTTP-запрос подмодулю NER с полезной нагрузкой в виде распознанного текста. Внутри контейнера с моделью NER запрос аналогичным образом принимается, обрабатывается, и формируется ответ с распознанными сущностями.

Таблица 1 — Производительность подмодулей OCR и NER

Подмодуль	**Занимаемый объём видеопамяти, ГБ**	**Время обработки одного входного примера, с**
OCR	2,1	0,24
NER	0,3	0,05

В табл. 1 приведены замеры быстродействия и ресурсоёмкости разработанных подмодулей в рамках описанного выше цикла тестирования с использованием Docker-контейнеров. Тестирование проводилось на пользовательской видеокарте NVIDIA GeForce RTX 3070 для входного изображения с разрешением 384×384.

Полученный в результате модуль состоит из подмодулей OCR и NER, а также из сервиса обмена данными между ними, используемого на стороне аналитической платформы, в которой выполнялась последующая интеграция подмодулей.

Заключение

В данной работе представлен модуль, выполняющий извлечение информации об условиях химической реакции из текста, расположенного на иллюстрациях к научным статьям. Процесс извлечения информации был разделён на два этапа: распознавание текста на изображении и обнаружение именованных сущностей в распознанном тексте. Были рассмотрены существующие решения в соответствующих областях машинного обучения. В целях снижения затрат на подготовку целевых данных, была разработана программа, генерирующая синтетические данные для модели распознавания текста. Для учёта различных особенностей в целевых данных, увеличения объёма обучающего набора данных и улучшения обобщающей способности модели, ко входным изображениям применялся набор аугментаций. Также был использован подход предобучения модели распознавания сущностей на размещённом в открытом доступе наборе размеченных химических патентов. Для решения проблемы утери пробелов в распознанном тексте, необходимых для корректного разграничения сущностей, были использованы символьные токены. Во избежание потери смысловой и контекстной информации о словах, к векторным представлениям символов было решено добавлять векторные представления соответствующих слов, для чего архитектура модели BERT была модифицирована.

После реализации и развёртывания модуля было проведено его тестирование. Результаты тестирования показали, что модуль способен решать поставленную задачу. Дальнейшие исследования будут направлены на повышение точности определения границ сущностей и их классификации.

Список литературы

1. Shi B., Bai X., Yao C. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition // IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015. — Vol. 39. — P. 2298-2304.

2. Yousef M., Hussain K.F., Mohammed U.S. Accurate, Data-Efficient, Unconstrained Text Recognition with Convolutional Neural Networks // Pattern Recognit., 2018. — Vol. 108. — P. 107482.

3. Taylor L., Nitschke G.S. Improving Deep Learning with Generic Data Augmentation // 2018 IEEE Symposium Series on Computational Intelligence (SSCI), 2018. — Vol. 1. — P. 1542-1547.

4. Singh S.S., Karayev S. Full Page Handwriting Recognition via Image to Sequence Extraction // Document Analysis and Recognition — ICDAR 2021, 2021. — Vol. 3. — P. 55-69.

5. Li M., Lv T., Cui L., Lu Y., Florencio D.A., Zhang C., Li Z., Wei F. TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models // Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence, 2023. — Vol. 37. — P. 13094–13102.

6. Luo L., Yang Z., Yang P., Zhang Y., Wang L., Lin H., Wang, J. An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition // Bioinformatics, 2018. — Vol. 34. — P. 1381-1388.

7. Lee J., Yoon W., Kim S., Kim D., Kim S., So C.H., Kang J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining // Bioinformatics, 2019. — Vol. 36. — P. 1234-1240.

8. Devlin J., Chang M., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // North American Chapter of the Association for Computational Linguistics, 2019. — Vol. 1. — P. 4171-4186.

9. Hu J., Shen Y., Liu Y., Wan X., Chang T. Hero-Gang Neural Model For Named Entity Recognition // Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2022. — Vol. 1. — P. 1924–1936.

10. Liu Z., Hu H., Lin Y., Yao Z., Xie Z., Wei Y., Ning J., Cao Y., Zhang Z., Dong L., Wei F., Guo B. Swin Transformer V2: Scaling Up Capacity and Resolution // 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. — Vol. 1. — P. 11999-12009.

11. Transformers Documentation / Hugging Face [Электронный ресурс]. — URL: https://huggingface.co/docs/transformers/en/index.

12. Kingma D.P., Ba J. Adam: A Method for Stochastic Optimization // 3rd International Conference on Learning Representations (ICLR 2015), 2015. — Vol. 1.

13. Chemical reactions from US patents (1976-Sep2016) / figshare [Электронный ресурс]. — URL: https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873.

14. Alshammari N.O., Alanazi S.A. The impact of using different annotation schemes on named entity recognition // Egyptian Informatics Journal, 2021. — Vol. 22. — P. 295-302.

15. Merkel, D. Docker: lightweight Linux containers for consistent development and deployment // Linux Journal 2014, 2014. — Vol. 1. — P. 2.

16. Requests User Guide / Requests: HTTP for Humans™ [Электронный ресурс]. — URL: https://requests.readthedocs.io/en/latest/.

17. Introduction / Uvicorn [Электронный ресурс]. — URL: https://www.uvicorn.org/.

18. FastAPI / FastAPI [Электронный ресурс]. — URL: https://fastapi.tiangolo.com/.

References

1. Shi B., Bai X., Yao C. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition // IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015. — Vol. 39. — P. 2298-2304.

2. Yousef M., Hussain K.F., Mohammed U.S. Accurate, Data-Efficient, Unconstrained Text Recognition with Convolutional Neural Networks // Pattern Recognit., 2018. — Vol. 108. — P. 107482.

3. Taylor L., Nitschke G.S. Improving Deep Learning with Generic Data Augmentation // 2018 IEEE Symposium Series on Computational Intelligence (SSCI), 2018. — Vol. 1. — P. 1542-1547.

4. Singh S.S., Karayev S. Full Page Handwriting Recognition via Image to Sequence Extraction // Document Analysis and Recognition — ICDAR 2021, 2021. — Vol. 3. — P. 55-69.

5. Li M., Lv T., Cui L., Lu Y., Florencio D.A., Zhang C., Li Z., Wei F. TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models // Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence, 2023. — Vol. 37. — P. 13094–13102.

6. Luo L., Yang Z., Yang P., Zhang Y., Wang L., Lin H., Wang, J. An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition // Bioinformatics, 2018. — Vol. 34. — P. 1381-1388.

7. Lee J., Yoon W., Kim S., Kim D., Kim S., So C.H., Kang J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining // Bioinformatics, 2019. — Vol. 36. — P. 1234-1240.

8. Devlin J., Chang M., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // North American Chapter of the Association for Computational Linguistics, 2019. — Vol. 1. — P. 4171-4186.

9. Hu J., Shen Y., Liu Y., Wan X., Chang T. Hero-Gang Neural Model For Named Entity Recognition // Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2022. — Vol. 1. — P. 1924–1936.

10. Liu Z., Hu H., Lin Y., Yao Z., Xie Z., Wei Y., Ning J., Cao Y., Zhang Z., Dong L., Wei F., Guo B. Swin Transformer V2: Scaling Up Capacity and Resolution // 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. — Vol. 1. — P. 11999-12009.

11. Transformers Documentation / Hugging Face [Website]. — URL: https://huggingface.co/docs/transformers/en/index.

12. Kingma D.P., Ba J. Adam: A Method for Stochastic Optimization // 3rd International Conference on Learning Representations (ICLR 2015), 2015. — Vol. 1.

13. Chemical reactions from US patents (1976-Sep2016) / figshare [Website]. — URL: https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873.

14. Alshammari N.O., Alanazi S.A. The impact of using different annotation schemes on named entity recognition // Egyptian Informatics Journal, 2021. — Vol. 22. — P. 295-302.

15. Merkel, D. Docker: lightweight Linux containers for consistent development and deployment // Linux Journal 2014, 2014. — Vol. 1. — P. 2.

16. Requests User Guide / Requests: HTTP for Humans™ [Website]. — URL: https://requests.readthedocs.io/en/latest/.

17. Introduction / Uvicorn [Website]. — URL: https://www.uvicorn.org/.

18. FastAPI / FastAPI [Website]. — URL: https://fastapi.tiangolo.com/.