*ВЕСТНИК ВОЕННОГО ИННОВАЦИОННОГО ТЕХНОПОЛИСА «ЭРА», 2022, том 3, № 1, с. 57–67*

## ЭЛЕКТРОННАЯ КОМПОНЕНТНАЯ БАЗА И ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ УПРАВЛЕНИЯ

*УДК 004.032.26*

# СОВРЕМЕННЫЕ МЕТОДЫ ЭКСТРАКЦИИ СВЯЗАННЫХ ИМЕНОВАННЫХ СУЩНОСТЕЙ НА ПРИМЕРЕ БИОМЕДИЦИНСКИХ ТЕКСТОВЫХ ДАННЫХ

**© 2022 г. А. Г. Сбоев[1, 2, *], А. А. Селиванов[1], Р. Б. Рыбка[1], И. А. Молошников[1], А. В. Наумов[1], Г. В. Рыльков[1]**

[1] *Национальный исследовательский центр "Курчатовский институт", Москва, Россия*

[2] *Национальный исследовательский ядерный университет "МИФИ", Москва, Россия*

**E-mail: Sboev_AG@nrcki.ru*

Поступила в редакцию 14.12.2021 г.
После доработки 14.01.2022 г.
Принята к публикации 17.01.2022 г.

Проанализированы современные методы решения актуальной задачи распознавания именованных сущностей и связей между ними, являющейся ключевой в спектре задач извлечения смысла из текста. Рассматривается ряд применяемых подходов с анализом текущих точностей на различных корпусах размеченных примеров из области биомедицины, в котором данное направление представлено наиболее полно. Показано, что наилучшие результаты достигаются с использованием единой модели для совместного решения обеих задач с оценкой общей ошибки в процессе обучения. Обоснована перспектива расширения указанного подхода на сложные случаи пересекающихся и разрывных сущностей.

**DOI:** 10.56304/S2782375X22010193

## ВВЕДЕНИЕ

В настоящее время открытые интернет-источники содержат огромные объемы текстовых данных в виде научных и публицистических статей, отзывов на товары и услуги, новостных записей, блогов и микроблогов, форумов и т.д. Объем данных и отсутствие строгой структуры ограничивают возможности ручного анализа такой информации, что обусловливает актуальность использования средств автоматической обработки естественного языка (*Natural language processing*, **NLP**), в том числе для выявления структуры исходных данных, например распознавания именованных сущностей, их упоминаний и связей между ними. Решение перечисленных задач позволяет выделять из текста смысловую информацию на основе упомянутых в тексте связанных сущностей.

В случае с биомедицинской тематикой решение задач выделения именованных сущностей и связей между ними позволяет извлечь дополнительную информацию о лекарственных средствах, их действии, побочных эффектах, взаимодействии лекарственных средств и их влиянии на организм, что может быть использовано в задачах фармаконадзора, социального мониторинга, маркетинга.

Таким образом, актуальным является рассмотрение методов автоматической обработки естественного языка в части распознавания именованных сущностей и связей между ними на примере текстов биомедицинской тематики.

В данной работе рассматриваются существующие подходы к извлечению связанных именованных сущностей; современные методы извлечения связанных именованных сущностей; открытые наборы (корпусы) размеченных биомедицинских текстовых данных; анализ результатов, полученных на основе методов, вошедших в обзор, на наиболее популярных корпусах.

## ПОДХОДЫ К ЭКСТРАКЦИИ СВЯЗАННЫХ ИМЕНОВАННЫХ СУЩНОСТЕЙ

Наибольшее развитие получили два подхода на базе глубоких нейросетевых моделей: каскадный и совместный. Первый предполагает последовательный анализ текста, когда сначала решается задача распознавания именованных сущностей (*named entity recognition*, **NER**), далее задача заключается в определении наличия и типов связей между выделенными сущностями (*relation extraction*, **RE**). Для каждой из задач настраивается отдельная модель. Это позволяет контролировать

процесс обучения каждой из частей и более гибко выбирать методы и релевантные гиперпараметры.

Второй подход предполагает решение двух задач в рамках единой модели, содержащей несколько блоков анализа, обучаемых совместно. Преимуществом такого способа является использование общей ошибки в процессе обучения всех блоков единой модели. Кроме того, обучение модели в рамках совместного подхода возможно без наличия в корпусе информации о местоположении экстрагируемых сущностей в анализируемом тексте.

## МЕТОДЫ НА ОСНОВЕ КАСКАДНОГО ПОДХОДА

*Методы распознавания именованных сущностей.* Методы NER по принципу работы с текстом можно разделить на две категории: методы на основе определения типов сущности для каждого “токена” (значимой для анализа последовательности символов, но не более одного слова) входного текста, методы на основе определения “спана” сущности (позиции начального и конечного символа сущности в тексте, но не менее одного слова).

Первый подход предполагает выбор схемы разметки токенов текста, наиболее популярными являются [1]:

– *IO* – наиболее простая схема, применимая к задаче разметки токенов. Тег ***I*** (*inside*) используется, чтобы отметить токены, которые относятся к сущностям, тег ***O*** (*outside*) проставляется токенам, которые не входят в сущности. Недостатком подхода является невозможность отделить разные сущности, сущности, которые идут одна за другой, а также пересекающиеся (overlapping) сущности;

– *IOB* – также известная как *BIO*, используется на конференции *Computational Natural Language Learning* (**CoNLL**) [2]. Каждому токену в тексте присваивается один из следующих тегов: тег начала именованной сущности (***B*** – *beginning*), тег включения в именованную сущность (***I***) или тег ***O***, который обозначает, что токен не относится ни к одной из выделенных именованных сущностей.

– *IOE* – принцип разметки, аналогичный принципу *IOB*, но вместо тега начала сущности (***B***) используется тег конца (***E***).

– *IOBES* – альтернатива схеме *IOB*, содержащая больше информации о границах сущностей: в дополнение к тегам начала (***B***), конца (***E***), включения (***I***) в сущность и отсутствия принадлежности к сущности (***O***) существует тег сущности из одного токена (***S*** – *single-token entity*).

Также существуют иные схемы разметки (*IB*, *IE*, *BIES*), представляющие собой иные комбинации уже упомянутых тегов.

Методы на основе разметки по токенам являются первыми, которые были использованы в задаче распознавания именованных сущностей. Такие методы предполагают присвоение каждому токену одной из возможных комбинаций из типа сущности и метки из схемы разметки токенов. Примерами работ, описывающих методы на основе этого подхода, являются [3–13].

Применение данного подхода к вложенным или разрывным сущностям предполагает усложнение разметки [9, 10], или ее многоуровневость [14–16]. В качестве альтернативного варианта решения проблемы появился подход на основе определения границ сущностей. Основное различие методов в рамках подхода заключается в способе получения этих границ. В [17] рассмотрены узлы графа в качестве основы для получения границ спанов. В [17–22] использован перебор доступных комбинаций слов, которые могут формировать сущности. В [23] методы на основе гиперграфов рассмотрены как эффективный способ представления экспоненциально растущего числа возможных вложенных сущностей в предложении, что изучается в [24–26].

В последнее время для определения именованных сущностей используется гибридный подход на основе совместного использования подхода по токенам и уточнения границ сущностей. Таким образом, с одной стороны, исчезает необходимость перебора всех возможных комбинаций токенов, с другой, это позволяет модели учитывать информацию о границах именованных сущностей [27–31]. Рассмотренный подход применялся в задачах выделения именованных сущностей из текстов биомедицинской тематики, где хорошо себя зарекомендовал в задаче выделения побочных эффектов лекарств из открытого корпуса текстов CADEC [32].

В [33] особое внимание уделено проблеме выделения разрывных именованных сущностей. Для их выделения применена модель, основанная на *shift-reduce*-разборщике [34, 35], который использует стек для хранения частично обработанных спанов (коротких фрагментов текста в несколько токенов) и буфер для хранения необработанных токенов. Основное отличие данной работы от других заключается в том, что был разработан новый набор действий перехода состояний парсера специально для распознавания прерывистых (*discontinuous*) и перекрывающихся (*overlapping*) структур сущностей:

– *SHIFT* – перемещает первый токен из буфера в стек, это подразумевает, что данный токен является частью сущности;

– *OUT* – вынимает первый токен из буфера, указывая, что он не принадлежит ни к одной сущности;

– *COMPLETE* – вынимает верхний спан стека, выводя его как упоминание сущности;

– *REDUCE* – вынимает два верхних спана из стека и объединяет их в новый спан, который затем подается обратно в стек;

– *LEFT-REDUCE* – аналогично действию *REDUCE*, за исключением того, что левый спан сохраняется в стеке. Это действие указывает на то, что данный спан участвует в нескольких сущностях.

– *RIGHT-REDUCE* – аналогично действию *LEFT-REDUCE*, за исключением того, что только правый спан сохраняется в стеке.

Входной текст для парсера кодируется с помощью посимвольной модели CNN. Получившиеся токены подаются в двунаправленную модель LSTM, образуя контекстное представление каждого токена. Данное представление объединяется с контекстно-зависимым представлением слов предобученной модели ELMo [36]. Чтобы отразить взаимодействие между спанами в стеке парсера и токенами в буфере парсера, используется мультипликативный механизм внимания [37]. Выбор следующего действия парсера проводится с помощью линейного слоя с функцией активации Softmax.

В [38] модель SpanBERT [39] использована для векторного представления токенов слов. Предварительное обучение данной модели проходит таким образом, чтобы закодированный вектор токена содержал информацию об окружении этого токена в предложении. Полученные векторы токенов подаются на вход линейному слою нейронной сети размерности 3, активности которого интерпретируются как принадлежность слова к одной из меток схемы разметки: *B* (метка начала сущности), *I* (метка продолжения сущности), *O* (метка отсутствия сущности). Полученные активности нормализуются с помощью многомерной логистической функции (Softmax).

Полученные нормализованные активности обрабатываются с помощью метода условных случайных полей, что служит для уменьшения шума в сгенерированных моделью активностях и является одной из стандартных практик в решении задачи выделения именованных сущностей.

На последнем этапе активности отдельных токенов агрегируются для получения активностей слова посредством применения следующих правил: если для выбранного тега активности всех токенов слова составляют 0, активность данного тега для слова также принимается за 0. Если хотя бы одному токену слова присвоен тег *B*, считается, что слово имеет тег *B*. Если хотя бы одному токену слова присвоен тег *I*, считается, что слово имеет тег *I*, если оно не имеет тега *B*.

В [40] представлен генеративный подход на основе языковой модели BART [41] с механизмом указателя (*Pointer Network*) [42], который на основе текста обучается генерировать индексы слов (или токенов), относящихся к именованной сущности, а также индексы классов сущностей. В работе также определено, что принимать за атомарный элемент анализа текста – токен, спан или слово. Модель, которая обучается генерировать индексы слов, показывает большую точность по сравнению с использованием других вариантов разбивки текста.

*Методы определения наличия и типов связей между выделенными сущностями.* В научной литературе последних лет в задачах анализа естественного языка зарекомендовал себя подход на основе языковых моделей топологии Трансформер [43], предварительно обученных на больших объемах неразмеченных данных. Этот подход был подробно исследован в области анализа биомедицинских текстов, в качестве данных для предварительного обучения, как правило, используются база статей из области биомедицины *Pubmed*, база медицинских выписок *MIMIC*-III, а также иные крупные базы данных биомедицинской тематики. В качестве достоинств таких моделей следует отметить их обобщающую способность по всему используемому набору неразмеченных текстов, что позволяет в процессе анализа текстов схожей тематики использовать информацию, извлеченную в ходе предварительного обучения.

Языковые модели на основе топологии Трансформер часто используются в качестве базовых элементов для других моделей решения задач анализа текстов. Применяемые в составе алгоритмов языковые модели могут различаться как числом обучаемых параметров и конфигурацией слоев, так и принципами работы.

В [69] языковая модель используется для получения векторного представления слов, которое конкатенируется с векторами, кодирующими положение слова относительно сущностей, между которыми модель должна определить связь. Полученное векторное представление обрабатывается сверточной сетью. Информация о молекулярной структуре лекарств обрабатывается с помощью графовой нейронной сети (*graph neural network*, **GNN**). Полученные вектора для каждой целевой сущности конкатенируются. В результате в классификационный слой подается следующая информация: векторное представление входного текста, конкатенация векторных представлений описаний рассматриваемых лекарств, конкатенация векторных представлений молекулярной структуры рассматриваемых лекарств.

Большинство других работ, анализируемых далее, рассматривали задачу RE как классификационную: на вход модели подается текст, в котором сущности “экранированы”, т.е. заменены специальными токенами, отражающими, между какими сущностями необходимо определить наличие связи и какие еще сущности есть во входном тексте. Если связей в тексте несколько, создаются копии текста по числу связей, в каждой отмечается пара “целевых” сущностей. Таким образом, векторное представление слов текста формируется с учетом информации о том, какие сущности есть в тексте и между какими из них определяется связь.

Для решения классификационной задачи с помощью языковых моделей на основе топологии Трансформер ко входной последовательности токенов добавляется специальный токен [CLS]. Вектор, соответствующий этому токену, в процессе работы модели формируется таким образом, чтобы агрегировать информацию о контексте, релевантную решаемой задаче. Данный вектор подается в выходной слой нейронной сети, служащий для определения класса входного объекта. Таким образом, в процессе обучения сети веса модели настраиваются так, чтобы вектор токена [CLS] позволял предсказывать класс с наименьшей возможной ошибкой.

Существует широкий спектр моделей на основе наиболее популярной трансформер-архитектуры BERT [44] – sciBERT [45], BioBERT [46], DischargeBERT [47], BlueBERT [48], ClinicalBERT [49], PubMedBERT [50]. Согласно проекту BLURB [51] наилучшей моделью на основе BERT для решения задач определения сущностей и связей между ними является PubmedBERT. Базовая конфигурация модели BERT включает в себя 12 слоев типа Трансформер размерности 768 с 12 блоками механизмов внимания в каждом слое. Размер словаря англоязычного BERT – 30000 слов, многоязычного – 110000 слов. Суммарное число настраиваемых параметров (весов нейронной сети) составляет 110 миллионов для англоязычного BERT и 172 миллиона для многоязычного. В [56] представлена модификация архитектуры BERT (R-BERT). В данной модели BERT используется для получения векторного представления слов текста. Для получения векторного представления именованной сущности вектора слов сущности усредняются. Конкатенированные векторы сущностей подаются на вход полносвязному слою для нелинейного преобразования, полученный вектор подается в линейный слой с функцией активации Softmax, на основе которого определяется наличие связи.

Также появляются модели на основе более сложных и объемных языковых моделей, таких как RoBERTa, ELECTRA и ALBERT – bioRoBERTa [52], BioELECTRA [53], RoBERTa-sag [54], bioALBERT [55]. Базовая конфигурация модели XLM-RoBERTa включает в себя 24 слоя типа Трансформер размерности 1024 с 16 механизмами внимания в каждом слое. Размер словаря – 250000 слов. Итоговое число настраиваемых параметров – 550 миллионов. Хотя модели RoBERTa и ELECTRA более требовательны к вычислительным ресурсам, они обладают лучшей обобщающей способностью, что ведет к росту точности решения задач при условии правильного выбора параметров обучения.

В результате анализа современных методов на основе каскадного подхода можно сделать вывод о том, что популярным решением является использование предварительно обученных языковых моделей на основе топологии Трансформер как для решения задачи экстракции именованных сущностей, так и для определения связей между ними. Для достижения большей эффективности используются различные методы агрегации векторного представления токенов текста.

В случае работы в рамках каскадного подхода выделения связанных именованных сущностей последовательно решаются задачи NER и RE с использованием двух предварительно обученных языковых моделей, требующих значительных вычислительных ресурсов.

## МЕТОДЫ НА ОСНОВЕ СОВМЕСТНОГО ПОДХОДА

*Совместный метод на основе предобученной языковой модели*. Данный метод [57] нивелирует отмеченный ранее недостаток каскадного подхода, используя одну языковую модель для решения двух задач. В качестве языковой модели используется оригинальный мультиязычный BERT. Данная модель используется для получения векторного представления каждого токена входного текста, которые затем подаются на вход полносвязного слоя для классификации по выбранной схеме разметки токенов (в оригинальной работе используется схема *IOBES*). Предсказанные теги сущностей кодируются с помощью слоя вложения (*Embedding*), полученные вектора конкатенируются с векторным представлением токенов. Далее формируются потенциальные связи на основе полного перебора всех пар тегов конца сущности (*S*- и *E*-).

Каждая сущность пары подается в полносвязный слой для формирования представления “головной” (*head*) или “конечной” (*tail*) сущности связи. Полученные вектора подаются в биафинный классификатор, который определяет наличие связи между сущностями. Биафинный клас-

сификатор представляет собой линейный слой нейронной сети для обработки двух входов

$$Biaffine(x_1, x_2) = x_1^T U x_2 + W(x_1 \| x_2) + b,$$

где $x_1, x_2$ – входные векторы размерности $m$, соответствующие двум сущностям предполагаемой связи, $U$ – матрица весов нейронной сети размерности $m \times C \times m$ ($C$ – число типов связей, включая тип "связь отсутствует"), $W$ – матрица весов нейронной сети размерности $C \times (2*m)$, $b$ – аддитивный коэффициент смещения.

В процессе обучения используется совместная функция ошибки, которая представляет собой сумму ошибок части модели для определения именованных сущностей и части модели для определения связей между сущностями:

$$L_{model} = L_{NER} + \lambda L_{\text{Я}},$$

где $\lambda$ – поправочный коэффициент для части функции ошибки, связанной с определением связей. В течение первой эпохи обучения модели данный коэффициент линейно растет от 0 до 1, поэтому в течение первой итерации обучения приоритетным становится определение именованных сущностей. Такой механизм позволяет увеличить итоговую точность работы модели определения связей.

*SpERT.* В [58] представлена модель SpERT на основе языковой модели BERT. Модель последовательно решает задачи выделения именованных сущностей и связей между ними. Для решения задачи выделения именованных сущностей формируются все возможные последовательные наборы слов в тексте (ограниченные по длине в оригинальной статье, причем длина последовательности не может превышать десяти токенов), которые затем классифицируются моделью по типу сущности. Для этого используется линейный слой с функцией активации Softmax. На вход слою подается вектор предполагаемой сущности, который формируется функцией *max-pooling* по векторам всех токенов сущности, конкатенированный с вектором, кодирующим длину сущности в токенах. Результаты классификации фильтруются, формируются пары сущностей, для которых модель определяет наличие связи и ее тип. На вход части определения связей подаются две сущности, векторы которых аналогичны входу части определения именованных сущностей. Эти векторы конкатенируются с объединенным вектором контекста, под которым понимают все слова между рассматриваемой парой сущностей в тексте. Полученный вектор, представляющий пару сущностей, подается на вход полносвязному слою с сигмоидальной функцией активации и размерностью, равной числу классов связей. Результирующий класс связи между сущностями определяется по порогу (в оригинальной работе порог равен 0.5, если активность нейрона выше него, то паре сущностей присваивается связь).

*Объединенная модель на основе классификации наборов слов.* Данная модель основана на архитектуре SpERT [59], но содержит следующие дополнения: вектор для классификации набора слов дополняется вектором контекста предложения, закодированным вектором длины входной последовательности слов, векторным представлением первого и последнего слова набора. Исследовано использование различных вариантов механизма внимания в частях модели, а также использованы коэффициенты для слагаемых функции ошибки, отвечающих за решение разных задач. Сделан вывод о том, что наиболее эффективным механизмом внимания является множественный (*multihead*) механизм внимания, позволяющий достичь большей точности по сравнению с иными вариантами (базовым, аддитивным, мультипликативным).

*Объединенная модель на основе классификации наборов слов для распознавания разрывных и пересекающихся сущностей.* Модель [60] формирует векторное представление слова на основе языковой модели BERT, выход которой обрабатывается двунаправленным LSTM. Дополнительно модель использует информацию о синтаксической структуре предложения, используя графовые сверточные слои со множественным механизмом внимания для обработки векторного представления слов и матрицы смежности синтаксического графа. Полученные векторы используются для векторного представления последовательности спанов: в данной модели используется конкатенация векторов первого и последнего слова последовательности, а также закодированное вектором число слов в последовательности. Векторное представление последовательности подается в полносвязный слой и нормализуется многомерной логистической функцией для определения типа сущности. Определенные сущности попарно подаются в полносвязный слой для определения типа связи между ними.

*Объединенная сеть на основе гиперграфа областей текста.* Данный метод [61] основан на представлении текста в виде графа, где узлами являются части текста, разделенные сущностями. Каждая пара сущностей в предложении рассматривается как разделитель предложения. В результате образуется пять областей токенов – две области, соответствующие сущностям, и три, на которые эти сущности разделяют предложение. Данные области представляют собой наборы токенов для формирования векторных представлений каждой области предложения.

Для обработки формируемого графа регионов и векторного представления узлов используется архитектура *Sequence-Enhanced Graph* (**SEG**), ос-

**Таблица 1.** Статистика по корпусу CADEC

| Характеристика | Значение |
|---|---|
| Число отзывов | 1253 |
| Среднее число слов в отзыве | 98 |
| Число сущностей | 8708 |
| Число сущностей типа *ADR* | 5937 |
| Число сущностей типа *Drug* | 1796 |
| Число сущностей типа *Disease* | 282 |
| Число сущностей типа *Symptom* | 268 |
| Число сущностей типа *Finding* | 425 |

нованная на графовых сверточных слоях и двунаправленном LSTM, что позволяет ей формировать представление региона с учетом векторного представления отдельных токенов, входящих в регион. В дальнейшем векторное представление региона используется для определения именованных сущностей и связей между ними.

## НАБОРЫ ДАННЫХ

*CADEC*. Это корпус аннотаций о побочных эффектах лекарственных препаратов [32] для медицинских сообщений, взятых с форума AskaPatient. Аннотирование выполнено студентами-медиками и компьютерными специалистами. На форуме собраны рейтинги и обзоры лекарств от потребителей. Собранный корпус содержит 1253 сообщения с 7398 предложениями. Аннотированы следующие сущности: медикаменты, нежелательные реакции, симптомы, заболевания, находки (неблагоприятный побочный эффект, заболевание, симптом, или любое другое клиническое понятие, которое может подпадать под любую из этих категорий, но не ясно, к какой из них он принадлежит). В аннотировании приняли участие четыре студента-медика и два специалиста по информатике. Для согласования разметки все аннотаторы совместно размечали несколько текстов, после чего тексты распределяли по аннотаторам. Все аннотированные тексты были проверены тремя авторами корпуса на очевидные ошибки, например недостающие слова и символы, опечатки и т.д. В табл. 1 представлена статистика по корпусу CADEC.

Число отзывов, в которых присутствуют сущности и типа ADR, и типа Symptom, 139.

*DDI2013* [62] – семантически размеченный корпус документов, описывающих взаимодействие между препаратами. Источник – база данных *DrugBank* и аннотации базы *MEDLINE* по теме взаимодействия медикаментов.

Данный корпус специально сформирован для соревнования DDIExtraction 2013, посвященному автоматическому выделению из текстов информации о медикаментах: именованных сущностей и связей между ними. Корпус размечен специалистами в области фармакологии вручную. Итоговый корпус содержит 572 документа, которые описывают взаимодействие медикаментов из *DrugBank*, и 142 аннотации из базы *MedLine*. Разметка представлена в формате XML. В табл. 2 приведена краткая статистика по сущностям и связям корпуса DDI2013.

*ADE*. Корпус побочных эффектов медикаментов (*adverse drug effect*, **ADE**) [63] состоит из 2972 документов, случайно выбранных из 30000 аннотаций статей в *PubMed*, которые были размечены вручную тремя аннотаторами. Корпус содержит три типа сущностей: *Drug*, *Adverse Effect*, *Dosage*. Аннотаторы выделяли связи в предложениях между медикаментами (*Drug*) и побочными эффектами (*AdE*); медикаментами (*Drug*) и дозировками (*Dosage*), отдельно отмечали предложе-

**Таблица 2.** Краткая статистика по сущностям и связям корпуса DDI2013

| Характеристика | Часть *DrugBank* | Часть *MEDLINE* | Всего |
|---|---|---|---|
| Число текстов | 572 | 142 | 714 |
| Число предложений | 5675 | 1301 | 6976 |
| Число сущностей | 12929 | 1836 | 14765 |
| Число сущностей типа *Drugs* | 8197 | 1228 | 9425 |
| Число сущностей типа *Brand* | 1423 | 14 | 1437 |
| Число сущностей типа *Group* | 3206 | 193 | 3399 |
| Число сущностей типа *No Human* | 103 | 401 | 504 |
| Число связей | 3805 | 232 | 4037 |
| Число связей типа *ddi* | 178 | 10 | 188 |
| Число связей типа *advice* | 819 | 8 | 827 |
| Число связей типа *effect* | 1458 | 152 | 1610 |
| Число связей типа *mechanism* | 1206 | 62 | 1268 |

**Таблица 3.** Краткая статистика по корпусу ADE-EXT

| Характеристика | Значение |
|---|---|
| Число документов | 2972 |
| Число сущностей | 11070 |
| Число сущностей типа *Drug* | 5063 |
| Число сущностей типа *Adverse effect* | 5776 |
| Число сущностей типа *Dosage** | 231 |
| Число связей | 7100 |
| Число связей *Drug-ADE* | 6821 |
| Число связей *Drug-Dosage** | 279 |

* В соревновании не использовались, как правило, исключаются авторами других статей.

ния, которые не содержат побочных эффектов, относящихся к медикаментам. Задача, которую предполагается решать: выделение побочных эффектов медикаментов, упомянутых в предложении. Упоминания медикаментов, состояний и дозировок, не участвующие в описанных связях, не выделялись. С целью подготовки большего корпуса для обучения классификатора корпус был расширен текстами с автоматически выделенными сущностями, которые не включены в связи, но находятся в предложении со связями. Данный корпус называется ADE-EXT [64] и включает в себя еще 2269 медикаментов, 3437 состояний и 5986 ложных связей (совместно встречаемых пар "медикамент–состояние", которые не были размечены при аннотации). В табл. 3 представлена краткая статистика по корпусу ADE-EXT (ADEv2).

## МЕТРИКИ ОЦЕНКИ КАЧЕСТВА

В оценке эффективности моделей определения именованных сущностей могут быть использованы подходы точного сопоставления границ сущностей (*exact*) и частичного (*lenient*, *partial*, *relaxed*). Подход точного сопоставления границ сущностей предполагает, что модель определила сущность корректно лишь в случае, если границы определенной сущности в точности совпадают с эталоном. Подход частичного сопоставления предполагает, что токен сущности определен верно, даже если модель предсказала границы сущности, отличные от эталона.

Как правило, оценка точности определения связей конкретного класса проводится с помощью $f1$-меры.

Второй особенностью оценки моделей экстракции связанных именованных сущностей является агрегация результатов по отдельным классам сущностей и связей между ними. Так, может быть использована метрика *micro-averaged* $f1$-*score*, которая учитывает представительность каждого класса при интегральной оценке точности решения задачи и является средневзвешенным по всем классам; метрика *macro-averaged* $f1$-*score*, которая не учитывает представительность классов и является простым усреднением точностей по классам.

## РЕЗУЛЬТАТЫ

*Точность моделей выделения именованных сущностей на корпусе CADEC.* Набор данных CADEC является традиционным открытым корпусом текстов в области биомедицины, на котором проверяют методы экстракции именованных сущностей. Многие работы ограничивают задачу до выделения сущностей, обозначающих побочные эффекты от лекарств (*Adverse Drug Effect*, **ADE**). В табл. 4 представлено сравнение методов из публикаций последних лет в области определения побочных эффектов на корпусе ADE. Во всех рассматриваемых работах используется одинаковое разбиение данных: 70% выделяется для тренировки модели, 15% – для валидации, 15% – для тестирования.

*Точность моделей определения наличия связей между выделенными именованными сущностями на корпусе DDI2013.* Корпус DDI2013 включает в себя разметку по пяти типам сущностей и четырем типам связей. Встречаются разрывные и пересекающиеся сущности, что делает задачу определения связанных сущностей более сложной. В научной литературе существует несколько вариантов разбиения корпуса на обучающую и тестировочную выборки. Наиболее часто встречаются разби-

**Таблица 4.** Сравнение методов выделения именованных сущностей – побочных эффектов от лекарств на открытом корпусе медицинских текстов CADEC

| Метод | $f1$-*micro*, % (ADE класс) |
|---|---|
| Объединенная модель на основе классификации наборов слов для распознавания разрывных и пересекающихся сущностей [60] | 070 |
| Модель на основе переноса для разрывных именованных сущностей [59] | 69 |
| SpanBERT + CRF [33] | 68 |
| Объединенная генеративная сеть на основе BART и механизма указателя [40] | 71 |

**Таблица 5.** Сравнение точности методов определения наличия связей между заданными именованными сущностями на корпусе DDI2013 с разбиением данных с соревнования SemEval-2013

| Метод | *f*1-*micro*, % |
|---|---|
| Baseline SemEval [68] | 65 |
| BioBERT [69] | 79 |
| SciBERT [69] | 82 |
| Метод на основе описания лекарств и молекулярных структур из [69] | 84 |

**Таблица 6.** Сравнение точности методов определения наличия связей между заданными именованными сущностями на корпусе DDI2013 с разбиением данных из фреймворка BLURB

| Языковая модель | *f*1-*micro*, % |
|---|---|
| BERT | 79.4 |
| RoBERTa | 79.5 |
| BioBERT | 80.9 |
| SciBERT | 81.2 |
| ClinicalBERT | 78.2 |
| BlueBERT | 77.8 |
| PubMedBERT | 82.4 |
| BioELECTRA | 82.8 |
| BioALBERT | 84.1 |

ение с соревнования SemEval-2013 и разбиение, которое используется во фреймворке BLURB, обычно использующемся при решении задачи определения связанных именованных сущностей с помощью языковых моделей. В табл. 5 представлено сравнение точности методов определения связанных именованных сущностей на корпусе DDI2013 с разбиением данных с соревнования SemEval-2013. В табл. 6 представлено сравнение точности методов определения связанных именованных сущностей на корпусе DDI2013 с разбиением данных из фреймворка BLURB.

*Точность моделей определения связанных именованных сущностей на корпусе ADE.* Хотя корпус ADE содержит связи двух типов, как правило, рассматриваются именно связи между медицинскими средствами (*Drug*) и побочными эффектами (*Disease*). Стандартной практикой в работах является использование кросс-валидации по десяти частям данных с последующим усреднением результатов. В табл. 7 представлено сравнение методов определения связей между именованными сущностями на открытом корпусе медицинских текстов ADE.

## ЗАКЛЮЧЕНИЕ

Результаты проведенного обзора позволяют выделить два подхода к решению задачи экстракции связанных именованных сущностей: подход на основе топологии Трансформер и на основе графовых нейронных сетей с целью агрегации структурной информации текста с последующим ее использованием в целях дополнения векторного представления текстовых данных, которое также позволяет увеличить точность решения задач.

Одной из основных проблем, на решение которых направлены новые методы, является необходимость анализа текста при наличии пересекающихся и разрывных сущностей. Обработка таких случаев возможна на основе классификации принадлежности каждого токена определенному классу сущности. Использование такого решения в каскадном алгоритме выделения связанных именованных сущностей обладает недостатком – настройка моделей в его составе без учета общей ошибки, возникающей при совместном решении. Таким образом, ошибка решения общей задачи связана как с ошибкой распознавания именованных сущностей, так и с ошибкой установления наличия связи.

Существующие подходы на основе единой модели нивелируют указанный недостаток каскадного подхода. Однако модели данного типа более

**Таблица 7.** Сравнение методов экстракции связанных именованных сущностей на открытом корпусе медицинских текстов ADE

| Метод | *f*1-macro, % (среднее по десяти частям кросс-валидации |
|---|---|
| Совместный метод на основе предобученной языковой модели [57] | 86 |
| Совместная разделяющая фильтрующая сеть [66] | 83 |
| *A region-based hypergraph network for joint entity-relation extraction* [61] | 82 |
| Объединенная модель на основе классификации наборов слов [59] | 81 |
| Последовательные табличные кодировщики [67] | 80 |
| SpERT [58] | 79 |

ресурсоемки, что осложняется необходимостью подбора гиперпараметров и настройки блоков модели в рамках единого процесса обучения без разделения на отдельные этапы. Перспективным является предварительное обучение отдельных моделей в рамках каскадного алгоритма с выбором архитектур отдельных блоков модели и диапазонов гиперпараметров с последующим объединением в единую модель экстракции связанных именованных сущностей.

## СПИСОК ЛИТЕРАТУРЫ


1. *Alshammari N., Alanazi S.* // Egyptian Informatics J. 2021. V. 22 (3). P. 295.
2. *Sang E.T.K., Buchholz S.* // Proceedings of the fourth conference on computational natural language learning and of the second learning language in logic workshop (CONLL/LLL 2000). Lissabon, Portugal, 13–14 september 2000, ACL, 2000. P. 127.
3. *Ratinov L.A., Roth D.* // Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009). Boulder, Colorado, USA, 4–5 june, 2009, ACL, 2009. P. 147.
4. *Collobert R., Weston J., Bottou L. et al.* // J. Mach. Learn. Res. 2011. V. 12. P. 2493.
5. *Huang Z., Xu W., Yu K.* // arXiv preprint arXiv:1508.01991. 2015.
6. *Chiu J.P.C., Nichols E.* // Trans. Assoc. Comput. Linguistics. 2016. № 4. P. 357.
7. *Lample G., Ballesteros M., Subramanian S. et al.* // Proceedings of The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2016). San Diego California, USA, 12–17 june, 2016, ACL, 2016. P. 260.
8. *Alex B., Haddow B., Grover C.* // Proceedings of the Biological, translational, and clinical language processing conference (BioNLP@ACL 2007). Prague, Czech Republic, 29 june, 2007, ACL, 2007. P. 65.
9. *Strakova J., Straka M., Hajic J.* // Proceedings of the 57th Conference of the Association for Computational Linguistics (ACL 2019). Florence, Italy, 28 july, 2019, ACL, 2019. V. 1. P. 5326.
10. *Metke-Jimenez A., Karimi S.* // Proceedings of the First International Workshop on Biomedical Data Integration and Discovery (BMDID 2016). Kobe, Japan, 17 october, 2016, CEUR-WS, 2016. V. 1709, P. 1.
11. *Muis A.O., Lu W.* // Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017). Copenhagen, Denmark, 9–11 september, 2017, ACL, 2017. P. 2608.
12. *Dai X., Karimi S., Hachey B., Paris C.* // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020) Online, 5–10 july, 2020, ACL, 2020. P. 5860.
13. *Lafferty J.D., McCallum A., Pereira F.C.N.* // Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001) Williams College, Williamstown, MA, USA, 28 june 28–1 july, 2001. Morgan Kaufmann, 2001. P. 282.
14. *Ju M., Miwa M., Ananiadou S.* // Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2018) New Orleans, Louisiana, USA, 1-6 june, 2018, ACL, 2018. V. 1. P. 1446.
15. *Fisher J., Vlachos A.* // Proceedings of the 57th Conference of the Association for Computational Linguistics (ACL 2019), Florence, Italy, 28 july–2 august, 2019, ACL, 2019. V. 1. P. 5840.
16. *Shibuya T., Hovy E.H.* // Trans. Assoc. Comput. Linguistics. 2020. № 8. P. 605.
17. *Xu M., Jiang H., Watcharawittayakul S.* // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017). Vancouver, Canada, 30 july–4 august, 2017, ACL, 2017. V. 1. P. 1237.
18. *Luan Y., Wadden D., He L. et al.* // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019). Minneapolis, MN, USA, 2–7 june, 2019, ACL, 2019. V. 1 P. 3036.
19. *Yamada I., Asai A., Shindo H. et al.* // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020) Online, 16–20 november, 2020, ACL, 2020. P. 6442.
20. *Li X., Feng J., Meng Y. et al.* // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020) Online, 5–10 july, 2020, ACL, 2020. P. 5849.
21. *Yu J., Bohnet B., Poesio M.* // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020) 5–10 july, 2020, ACL, 2020. P. 6470.
22. *Wang J., Shou L., Chen K., Chen G.* // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020) 5–10 july, 2020, ACL, 2020. P. 5918.
23. *Lu W., Roth D.* // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015). Lisbon, Portugal, 17–21 september, 2015, ACL, 2015. P. 857.
24. *Katiyar A., Cardie C.* // Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2018). New Orleans, Louisiana, USA, 1–6 june, 2018, ACL, 2018. V. 1. P. 861.
25. *Wang B., Lu W.* // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018). Brussels, Belgium, 31 october–4 november, 2018, ACL, 2018. P. 204.
26. *Muis A.O., Lu W.* // Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017). Copenhagen, Denmark, 9–11 september, 2017, ACL, 2017. P. 2608.
27. *Wang B., Lu W.* // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, (EMNLP-IJCNLP 2019), Hong Kong, China, 3–7 november, 2019, ACL, 2019. P. 6215.

28. *Zheng C., Cai Y., Xu J. et al.* // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019), Hong Kong, China, 3–7 november, 2019, ACL, 2019, P. 357.
29. *Lin H., Lu Y., Han X., Sun L.* // Proceedings of the 57th Conference of the Association for Computational Linguistics (ACL 2019). Florence, Italy, 28 july–2 august 2019, ACL, 2019. V. 1. P. 5182.
30. *Wang Y., Li Y., Tong H., Zhu Z.* // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020). Online, 16–20 november, 2020, ACL, 2020. P. 6027.
31. *Luo Y., Zhao H.* // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020). Online, 5–10 july, 2020, ACL, 2020. P. 6408.
32. *Karimi S., Metke-Jimenez A., Kemp M., Wang C.* // J. Biomed. Inform. V. 1. № 55. P. 73.
33. *Dai X., Karimi S., Hachey B., Paris C.* // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020). Online, 5–10 july, 2020, ACL, 2020. P. 5860.
34. *Watanabe T., Sumita E.* // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing and the 7th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2015). Beijing, China, 2015, ACL, 2015. V. 2. P. 1169.
35. *Lample G., Ballesteros M., Subramanian S. et al.* // Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016), San Diego, California, USA, 12–17 june, 2016, ACL, 2016. P. 260.
36. *Peters M., Neumann M., Iyyer M. et al.* // Proceedings of the The 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2018), New Orleans, Louisiana, USA, 1–6 june, 2018, ACL, 2018. P. 2227.
37. *Luong T., Pham H., Manning C.D.* // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015). Lisbon, Portugal, 17–21 september, 2015, ACL, 2015. P. 1412.
38. *Portelli B., Passabi D., Serra G. et al.* // Proceedings of the 5th International Workshop on Health Intelligence (W3PHIAI-21). Online, 2–9 february, 2021, AAAI, 2021.
39. *Joshi M., Chen D., Liu Y. et al.* // TACL. 2020. V. 8. P. 64.
40. *Yan H., Gui T., Dai J. et al.* // arXiv preprint arXiv:2106.01223. 2021.
41. *Lewis M., Liu Y., Goyal N. et al.* // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020) Online, 5–10 july, 2020, ACL, 2020. P. 7871.
42. *Vinyals O., Fortunato M., Jaitly N.* // Adv. Neur. In. 2015. V. 28. P. 2692.
43. *Vaswani A., Shazeer N., Parmar N. et al.* // Adv. Neur. In. 2017. V. 30. P. 5998.
44. *Devlin J., Chang M.W., Lee K., Toutanova K.* // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019). Minneapolis, MN, USA, 2–7 june, 2019, ACL, 2019. V. 1. P. 4171.
45. *Beltagy I., Lo K., Cohan A.* // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, (EMNLP-IJCNLP 2019), Hong Kong, China, 3–7 november, 2019, ACL, 2019. P. 3615.
46. *Lee J., Yoon W., Kim S. et al.* // Bioinformatics. 2020. V. 36 (4). P. 1234.
47. *Van Aken B., Papaioannou J.M., Mayrdorfer M.* // Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2021). Online, 19–32 april, 2021, ACL, 2021. V. 1. P. 881.
48. *Peng Y., Yan S., Lu Z.* // Proceedings of the 18th BioNLP Workshop and Shared Task (BioNLP@ACL 2019), Florence, Italy, 1 august 2019, ACL, 2019P. 58.
49. *Huang K., Altosaar J., Ranganath R.* // arXiv preprint arXiv: 1904.05342. 2019.
50. *Gu Y., Tinn R., Cheng H. et al.* // HEALTH. 2021. № 3 (1). P. 1.
51. BLURB project. https://microsoft.github.io/BLURB/leaderboard.html
52. *Lewis P., Ott M., Du J., Stoyanov V.* // Proceedings of the 3rd Clinical Natural Language Processing Workshop (Clinical NLP 2020). Online, 19 november, 2020, ACL, 2020. P. 146.
53. *raj Kanakarajan K., Kundumani B., Sankarasubbu M.* // Proceedings of the 20th Workshop on Biomedical Language Processing (BioNLP@NAACL-HLT 2021), Online, 11 june, 2021, ACL, 2021. P. 143.
54. *Sboev A., Sboeva S., Moloshnikov I. et al.* // arXiv preprint arXiv:2105.00059. 2021.
55. *Naseem U., Dunn A.G., Khushi M., Kim J.* // arXiv preprint arXiv:2107.04374. 2021.
56. *Wu S., He Y.* // Proceedings of the 28th ACM international conference on information and knowledge management (CIKM 2019), Beijing, PRC, 3 november 2019, ACM Press, 2019. P. 2361.
57. *Giorgi J., Wang X., Sahar N. et al.* // arXiv preprint arXiv:1912.13415. 2019.
58. *Eberts M., Ulges A.* // European Conference on Artificial Intelligence (ECAI 2020). Online, 29 august–8 september 2020, IOS Press, 2020. P. 2006.
59. *Ji B., Yu J., Li S. et al.* // Proceedings of the 28th International Conference on Computational Linguistics (ICCL 2020), Enschede, The Netherlands, 28–30 september 2020, Springer, 2020. P. 88.
60. *Li F., Lin Z., Zhang M.* // Proceedings of The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021), Online, 1–6 august 2021, ACL, 2021. V. 1. P. 4814.
61. *Wan Q., Wei L., Chen X., Liu J.* // Knowl.-based Syst. 2021. № 228. P. 107298.

62. *Segura-Bedmar I., Martinez P., de Pablo-Sánchez C.* // J. Biomed. Inform. 2011. V. 44 (5). P. 789.
63. *Gurulingappa H., Rajput A.M., Roberts A. et al.* // J. Biomed. Inform. V. 45. P. 885.
64. *Gurulingappa H., Mateen-R.A., Toldo L.* // J. Biomed. Semant. 2012. № 3 (1). P. 1.
65. *Uzuner Ö., South B.R., Shen S., DuVall S.L.* // J. Am. Med. Inform. Assoc. 2011. V. 18 (5). P. 552 https://doi.org/10.1136/amiajnl-2011-000203
66. *Yan Z., Zhang C., Fu J. et al.* // Proceedings of the 201 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021), Online, 7–9 november 2021, ACL, 2021. P. 185.
67. *Wang J., Lu W.* // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020) Online, 16–20 november, 2020, ACL, 2020. P. 1706.
68. *Segura-Bedmar I., Martínez P., Herrero-Zazo M.* // Proceedings of the second Joint Conference on Lexical and Computational Semantics (*SEM-2013), Atlanta, Georgia, USA, 13–14 june, 2013, ACL, 2013. P. 341.
69. *Asada M., Miwa M., Sasaki Y.* // Bioinformatics. 2021. V. 37 (12). P. 1739.
70. *Patel R., Tanwani S., Patidar C.* // Informatica. 2021. V. 45 (3). P. 359.
71. *Sui D.* // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020) Online, 16–20 november, 2020, ACL, 2020. P. 2118.
72. *Roberts K., Rink B., Harabagiu S.* // Proceedings of the fourth i2b2/VA Shared-Task and Workshop Challenges in Natural Language Processing for Clinical Data (i2b2/VA 2010), Boston, MA, USA, november 2010, NCBI, 2010.
73. *Xu J., Lee H.J., Ji Z. et al.* // Proceedings of the Text Analysis Conference 2017 (TAC 2017), Gaithersburg, Maryland, USA, 13–14 november 2017, NIST, 2017, P. 34.
74. *Alimova I., Tutubalina E.* // J. Biomed. Informatics. 2020. V. 103. P. 103382.
75. *Yang X., Yu Z., Guo Y. et al.* // arXiv preprint arXiv: 2107.08957. 2021.