DOI 10.37882/2223-2966.2024.9.23

СОЗДАНИЕ ГРАФОВ ЗНАНИЙ С ИСПОЛЬЗОВАНИЕМ ВОЗМОЖНОСТЕЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

CREATING KNOWLEDGE GRAPHS USING THE CAPABILITIES OF ARTIFICIAL INTELLIGENCE

B. Mishchuk
I. Maklakhova
E. Stavitskaya

Summary. The paper presents and examines in detail the technique of automated construction of knowledge graphs using modern developments in the field of artificial intelligence. A feature of this work is the study and application of existing methods in the context of graph construction, despite the fact that many of the mentioned methods are also successfully used in the context of large language models.

Keywords: knowledge graph, artificial intelligence, machine learning.

Мищук Богдан Ростиславович
кандидат физ.-мат. наук, доцент, Балтийский федеральный университет им. Канта
b.mishchuk@yandex.ru
Маклахова Ирина Сергеевна
ст. преподаватель, Балтийский федеральный университет им. Канта
imaklakhova@mail.ru
Ставицкая Екатерина Петровна
ст. преподаватель, Балтийский федеральный университет им. Канта
noekaterina@yandex.ru

Аннотация. В работе представлена и детально рассмотрена методика автоматизированного построения графов знаний с использованием современных наработок в области искусственного интеллекта. Особенностью настоящей работы является изучение и применение существующих методов именно в контексте построения графов несмотря на то, что многие из упоминаемых методов также успешно используются в контексте больших языковых моделей.

Ключевые слова: граф знаний, искусственный интеллект, машинное обучение.

Введение

Как и многие новые подходы в области искусственного интеллекта, мир формальных определений графов знаний богат разнообразными интерпретациями. В 2014 году Ванг и его коллеги [2] представили граф знаний как многореляционный граф, в котором узлы выступают сущностями, а ребра отражают различные типы связей между ними. Однако такое определение не учитывает наличие семантических структур в графе знаний.

Позднее, в 2016 году, L. Ehrlinger и W. Wöß [3] уточнили, что граф знаний структурирует информацию в виде онтологии и способствует генерации новых знаний путем «логического обоснования». Подчеркивая фундаментальный компонент, поддерживающий информацию на уровне знаний, Wu и его коллеги [1] в 2017 году дали следующее определение: граф знаний является семантическим графом, в котором узлы представляют концепции (сущности/атрибуты/факты), а ребра изображают связи, объединяющие узлы на основе фоновых знаний о концепциях и их взаимоотношениях. Собственно, именно этим определением мы и будем пользоваться в рамках работы, имея в виду также что граф знаний в целом должен отражать некую онтологию.

Надо отметить, что многие известные, крупные системы графов знаний (открытые, и тем самым легко доступные для использования в исследованиях) были созданы с помощью краудсорсинга, например Freebase и Wikidata. Следовательно, программная система, позволяющая создавать граф знаний из «сырых» неструктурированных или полуструктурированных данных даст возможность существенно упростить и ускорить этот трудоемкий процесс.

Модель, способная анализировать сложные и длинные тексты, должна уделять особое внимание сложным взаимосвязям в предложениях при обработке информации о разных объектах, а также должна учитывать сложные взаимосвязи между предложениями и уметь обрабатывать несколько лингвистических элементов одновременно.

Следует отметить, что в тексте, написанный человеком, могут встречаться двусмысленные фразы, которые модели обычно неправильно интерпретируют без внешней информации. Еще одна сложность заключается в многоступенчатых логических рассуждениях. Для понимания сложных выражений необходимо изучить большое количество лингвистических структур.

В контексте обработки естественного языка для создания графа знаний можно выделить несколько ключевых задач, решаемых последовательно:

- Обнаружение сущностей в тексте — выявление значимых элементов;
- Устранение кореферентности — поиск и связывание всех терминов, отсылающих к одному объекту в тексте;
- Построение связей между сущностями — исследование взаимосвязей между выделенными элементами.

Все вышеперечисленные задачи будут детально рассмотрены в следующих разделах.

1. Обнаружение сущностей

Обнаружение сущностей — это процесс извлечения сущностей, или ключевых концепций, из полуструктурированных или структурированных данных. В структуре графа знаний узлами выступают сущности, которые отображают реальные объекты. Процесс выявления этих сущностей предполагает выполнение ряда шагов:

- Определение именованных сущностей.
- Типизация сущностей.
- Связывание сущностей.

Связывание сущностей связывает обнаруженные сущности с существующими узлами в графе знаний. Если соответствующий узел не найден, создается новый узел, представляющий новую сущность.

Обнаружение сущностей представляет собой базовое действие для построения графа знаний, поскольку именно оно обеспечивает основу для последующего связывания и тем самым, структурной организацией информации. Так же следует отметить, что обнаружение сущностей является одним из самых детально исследованных подходов из рассматриваемых в данной работе.

Распознавание именованных сущностей (Named Entity Recognition, NER) — это задача идентификации и классификации сущностей в полуструктурированных или неструктурированных данных. Полуструктурированные данные, примером которых являются веб страницы, сопровождаются семантическими подсказками, относящимися к структурам свойств и атрибутов, в то время как неструктурированные данные содержат только текст. Подходы, основанные на правилах [4, 5], являются общими решениями для NER. Наличие семантических подсказок, таких как теги и структура документа, позволяет использовать более сложные правила DBpedia [6] и YAGO [7].

Для NER [8] в неструктурированных данных используются статистические подходы, которые обучаются на аннотированных данных. Для улучшения производительности систем NER используются различные методы, такие как частично контролируемые подходы, которые объединяют правила, созданные человеком, с автоматическим обучением, и методы начальной загрузки, которые итеративно совершенствуют шаблоны правил.

Статистические подходы рассматривают распознавание именованных объектов как задачу последовательной классификации, где каждому слову присваивается тег, указывающий на тип сущности и ее границы в соответствии со схемой BIES (начало, середина, конец, одиночная).

В рамках исследования проблематики распознавания неструктурированных именованных сущностей, ключевую роль играет гипотеза о марковской зависимости тегов, при которой предполагается, что метка для каждого слова определяется исключительно его предшествующим контекстом. Эта предпосылка лежит в основе разработки алгоритмических приложений, основанных на скрытых марковских моделях (Hidden Markov Models, HMM) [9] и моделях условных случайных полей (Conditional Random Fields, CRF) [10], которые эффективно моделируют зависимости между последовательными элементами данных.

Иерархическая модель CRF также успешно обеспечивает рамки для анализа данных, которые имеют полуструктурированный характер и организованы в форме иерархического дерева, облегчая тем самым процесс совместного извлечения информации. Дополнительно, исследования A. Finn и N. Kushmerick [11] представили модель, основанную на методе опорных векторов (Support Vector Machine, SVM), для детектирования границ объектов в текстовых данных.

Глубокое обучение также становится основной тенденцией в распознавании именованных объектов, особенно для распознавания текстовых именованных объектов. Эти подходы к глубокому обучению обычно рассматривают распознавание именованных объектов как модель seq2seq (последовательности слов для обозначения последовательностей меток). Эти модели объединяют контекстные встраивания в соответствии с входными данными, а затем кодеры контекста выводят теги типа word с помощью декодеров тегов, таких как CRF structure или softmax structure.

CNN особенно эффективны для извлечения локальных характеристик, что полезно для обнаружения сущностей. R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. P. Kuksa [12] были первыми, кто использовал CNN с выходным уровнем CRF в качестве унифицированного решения для обнаружения объектов. E. Strubell, P. Verga, D. Belanger, and A. McCallum [13] усовершенствовали CNN с помощью расширенной свертки, которые увеличивают поле восприятия за счет пропуска части входных данных для усиленного обобщения.

Рекуррентные нейронные сети (RNN) хорошо подходят для обработки длинных предложений и усвоения глобальных контекстных особенностей. Однако RNN могут страдать от искажения контекста из-за более поздних слов. В контексте обработки естественного языка, рекуррентные нейронные сети (RNN) применяются для идентификации сущностей в тексте. Более того, бидирекциональные RNN, включая такие модели как Bi-LSTM-CRF и GRU, способны анализировать контекст, учитывая информацию как из прошлого, так и из будущего в потоке данных. Часто используются комбинированные подходы кодирования на уровне символов и слов, например, применение свёрточных нейронных сетей (CNN) для кодирования символов и долгой краткосрочной памяти (LSTM) для кодирования слов.

Еще одним направлением, которое проектирует важные взаимодействия для глобальных контекстов, является механизм внимания. Он позволяет моделям сосредотачиваться на важных контекстных элементах. L. Luo, Z. Yang, P. Yang, Y. Zhang, L. Wang, H. Lin, and J. Wang [14] внедряют мягкое внимание на уровне слов для улучшения распознавания именованных объектов. A.Z. Gregoric, Y. Bachrach, P. Minkovsky, S. Coope, and B. Maksak. [15] используют самостоятельное внимание к каждому слову для распознавания именованных объектов.

Сверточные графовые сети (GCN) обрабатывают контекст в лингвистических графовых структурах, таких как деревья зависимостей. Например, Cetoli и др. [16] предложили структуру GCN, которая кодирует функции из LSTM-proceed и использует синтаксические структуры для улучшения распознавания сущностей.

Предварительно обученные языковые модели, такие как Elmo [17], Ltp [18] и LUKE [19], предоставляют богатые представления слов и контекста. Они используются в качестве базовых знаний для обучения моделей распознавания сущностей, что приводит к повышению производительности.

Типизация сущностей в неструктурированных данных остается сложной задачей. Подходы с использованием глубокого обучения решают две основные проблемы при типизации объектов:

1. Нечастое использование редких типов.

Нейронные сети, такие как LSTM с вниманием , могут использовать иерархическое кодирование меток и контекстные представления для типизации редких сущностей.

2. Слишком специфичная типизация.

Чрезмерно специфичные аннотации типов выводят правильные типы, но не соответствуют текущему контексту данных.

Для дальнейшего изучения контекстных сценариев Zhang и др. [20] ввели представления на уровне документа, чтобы обеспечить глобальный контекст для обнаружения сущностей. Типичная модель типизации сущностей с помощью глубокого обучения включает:

— Кодирование сущностей и их контекста;
— Механизм внимания для выделения важных контекстных признаков;
— Функции для обработки редких и чрезмерно специфичных типов;
— Адаптивный порог вероятности для генерации меток типов в разных контекстах.

В новых моделях, основанных на внедрении, используются общие характеристики структуры глобальных графов и базовые знания для прогнозирования потенциальных типов объектов с помощью представлений.

C. Moon, P. Jones, и N. F. Samatova [21] предлагают модель TransE откорректированную путем оптимизации евклидова расстояния между объектами и представлениями их типов, ограниченного недостаточным количеством типов объектов и троекратных функций, чтобы в итоге получить модель TransE-ET. Новые решения создают различные графы, которые совместно используют разнообразные характеристики объектов, связанных с сущностями, для изучения внедрений с объектами типов сущностей.

Методика JOIE [22]: встраивает узлы сущностей в граф представления онтологии и в графы экземпляров, собирая типы сущностей по релевантности.

Задача связывания сущностей, также известная как устранение неоднозначности сущностей, состоит в том, чтобы связать упоминания сущностей с соответствующими объектами в графе знаний. Например, упоминание «Tesla» может относиться к автомобилю, корпорации или ученому.

В структурированных данных, таких как таблицы, семантические подсказки, такие как заголовки столбцов и метки типов, помогают идентифицировать сущности.

В неструктурированных данных, таких как тексты, модели связывания сущностей фокусируются на представлении контекста упоминаний сущностей.

Методы связывания сущностей в основном делятся на две категории:

Основанные на правилах методы: используют вручную созданные правила для поиска и связывания упоминаний сущностей.

Статистические методы: используют машинное обучение и статистические модели для автоматического обнаружения и связывания упоминаний сущностей.

Для неструктурированных данных модели связывания сущностей используют:

— Векторные представления слов и предложений.
— Контекстные эмбеддинги.
— Модели внимания для захвата важных контекстных признаков.

Статистические подходы, особенно основанные на вероятностных графах и моделях SVM, широко используются для связывания сущностей в полуструктурированных и неструктурированных данных. Такие модели строят вероятностный граф упоминаний сущностей. Семантические коэффициенты узлов графа используются для связывания объектов. Например, Исследование, проведённое в работе [23], представило новаторский подход к созданию факторного графа, который интегрирует алгоритм TF-IDF для анализа частотности меток сущностей в сочетаниях 'ячейка-текст' и типовых меток в 'заголовках столбцов'. Для повышения эффективности процесса связывания сущностей, были задействованы внешние базы данных. Примером такого усовершенствования является TabEL [24], который расширил свой факторный граф за счёт использования гиперссылок из Википедии для более точной оценки семантических связей, что способствует более точной классификации и устранению неоднозначностей.

Также Guo и др. [25] предложили вероятностную модель, адаптированную для работы с неструктурированными данными. Эта модель интегрирует априорные вероятности существования объекта, его контекста и именованных сущностей, что способствует более точному и обоснованному связыванию объектов. Применение вероятностного подхода позволяет учитывать неопределенность и шум в данных, что является ключевым аспектом при работе с неструктурированными информационными массивами.

Эти подходы демонстрируют применение статистического обучения и оптимизационных методов в задачах связывания объектов, подчеркивая важность точности и надежности в процессе классификации и анализа данных.

2. Устранение кореферентности

Кореферентность в лингвистике и обработке естественного языка (Natural Language Processing, NLP) относится к связыванию различных элементов текста, которые указывают на один и тот же объект или сущность.

В контексте неструктурированных текстов, кореферентные выражения часто встречаются в виде слов или фраз, отсылающих к определенным объектам или сущностям. Задача устранения кореферентности, или кореферентный анализ, заключается в идентификации и связывании таких выражений, которые относятся к одному и тому же объекту, несмотря на их различное лексическое представление.

Примером может служить текст: «При проверке газового оборудования в подъезде многоквартирного дома, хозяева трех квартир не предоставили вовремя доступ в своё жильё. Из-за нарушителей, всем людям подъезда пришлось сидеть без газа, пока не прошла проверка во всех квартирах.»

Здесь слово «нарушитель» кореферентно с фразой «хозяева трех квартир». Это демонстрирует, как кореферентность связывает различные части текста, обеспечивая его семантическую целостность и понимание.

Первые попытки охватить лингвистические объекты, на которые ссылаются в тексте, основывались на статистических характеристиках самих объектов, их упоминаниях и предшествующих событиях.

Другой подход демонстрируют модели, основанные на кластерах, рассматривающие задачу устранения кореферентности как задачу бинарной классификации (существуют ли между упоминаниями кореферентные связи или нет). Ранние кластерные модели учитывали только признаки, связанные с парой упоминаний. В работе [26] предложен метод кластеризации с одним звеном для обнаружения анафорических пар. Пример анафоры: «Через дорогу пробежала кошка, она была черного цвета». Слово «она» объясняется за счет предшествующей ей слова «кошка». В работе [27] продолжена разработка кластера на основе пар упоминаний, чтобы создать цепочки кореференции или отдельные списки. Позже исследователи сосредоточились на признаках, основанных на сущностях, чтобы учесть более сложные анафорические связи. Rahman и Ng [28] предложили модель кластеризации, основанную на классификации упоминаний, чтобы глубже изучить характеристики объектов. Stoyanov и Eisner [29] разработали сгруппированную кластеризацию на основе признаков объектов.

Методы, основанные на структурах деревьев и графов, зарекомендовали себя как эффективные инструменты для разрешения кореферентности. В рамках этих подходов разрабатываются гиперграфы, где рёбра способны соединять более двух вершин, что позволяет отражать множественные параллельные связи между различными упоминаниями. Исследование Cai и Strube [30] было посвящено анализу статистических свойств, чтобы назначить вес ребрам и получить разбиения на подмножества с основными ссылками с помощью алгоритмов кластеризации. Другие исследователи упростили графы, чтобы адаптировать их к древовидным методам разрешения кореферентности. Например, Bean и Riloff [31] разработали модель дерева принятия решений, которая

различает анафорические упоминания на основе контекстных признаков. Fernandes и другие [32] использовали алгоритм персептрона для определения деревьев кореференции для пар упоминаний.

Естественно для решения проблемы кореферентности существуют и модели на основе глубокого обучения, которые умеют автоматически обрабатывать текстовые данные и извлекать из них признаки, которые помогают определить, связаны ли два упоминания в тексте.

Многие ранние модели на основе глубокого обучения для разрешения кореферентности использовали сверточные нейронные сети (CNN). Xi и другие [33] использовали CNN для извлечения признаков из пар упоминаний, а затем оценивали их вероятность быть кореферентными. Одна из таких моделей, объединяла отдаленные объекты с помощью иерархических функций и оценивала пары упоминаний с помощью уровня softmax. Другая модель, разработанная Wu и другими. [34], была способна эффективно обрабатывать как кореферентные выражения, так и одноэлементные выражения. Эта модель использовала комбинации контекстных признаков из предшествующих элементов, упоминаний и пар упоминаний для улучшения производительности.

Рекуррентные нейронные сети (RNN) и их варианты, такие как LSTM, являются эффективными в выявлении глобальных закономерностей между парами упоминаний. Wiseman и другие [35] разработали модель устранения кореферентности на основе RNN. Gu и другие [36] преобразовали LSTM, чтобы исключать непохожие пары упоминаний.

Механизмы внимания позволяют моделям сосредоточиться на наиболее важных частях текста для принятия решения о кореферентности. Одним из распространенных примеров является структура Bi-LSTM с вниманием на уровне слов. Однако было разработано множество других механизмов внимания, специально предназначенных для устранения кореферентности. Например, биаффинная модель внимания для совместных задач, которая фиксирует взаимодействие в пределах слов для обнаружения связанных выражений; модель взаимного внимания, которая включает синтаксические функции с интерактивными функциями между структурами зависимостей и предшествующими элементами для определения промежутков слов. Clark и Manning [37] разработали стратегию на основе подкрепления обучения (RL), которая повышает надежность их нейронной модели разрешения кореферентности. Эта стратегия использует эвристическую сеть политик для фильтрации некорректных действий по сопоставлению кореферентности. Durrett и Klein [38] используют представления предшествующих элементов, чтобы выводить корреляции с помощью функций распределения. Martschat и Strube [39] исследуют семантику распределения по парам упоминаний и древовидным моделям, чтобы оптимизировать представление кореферентности.

3. Извлечение связей

Задачи извлечения отношений сосредоточены на извлечении взаимосвязей и свойств объектов из данных. Извлеченные отношения часто используются для классификации сущностей, так называемая классификация отношений. Существуют два основных типа извлечения отношений:

- Извлечение бинарных отношений: Извлечение тройки отношений между двумя сущностями (субъект, отношение, объект).
- Извлечение n-арных отношений: Извлечение тройки отношений для нескольких сущностей, таких как соавторы или участники проекта.

Извлечение отношений играет важную роль в создании графов знаний. Графы знаний представляют семантические связи между сущностями и используются для ответа на сложные вопросы и извлечения информации из данных.

В задачах извлечения открытых связей отношения извлекаются из неструктурированных данных без ограничений на заранее определенные типы связей. Эти методы находят существительные (субъект и объект) и глагольные фразы (предикат) в тексте, чтобы создавать структуру знаний в формате (субъект, предикат, объект). Статистические подходы также широко используются для извлечения открытых связей. Одним из популярных методов является построение вероятностных графов, которые позволяют передавать контекстную информацию в слабоструктурированных или неструктурированных текстах. Mulwad и его коллеги [40] разработали вероятностный граф для извлечения связей тегов из полуструктурированных табличных данных. Chen и Cafarella [41] использовали модуль, основанный на структуре CRF, с функцией поиска кадров, чтобы помечать ячейки метками их местоположения (например, слева, посередине и справа). На основе этого строится иерархическое дерево, в котором тройки отношений могут быть восстановлены с помощью родительско-дочерних структур. В рамках научных разработок, специалисты в области анализа данных активно внедряют вероятностные подходы для точного определения связей между элементами текста. Проект StatSnowball [42], эффективно демонстрирует использование марковских сетей, основанных на логических предположениях, что позволяет с высокой точностью выявлять структуры отношений.

Методы на основе правил были первыми попытками извлекать отношения из данных, используя шабло-

ны, созданные вручную. Однако эти методы требовали большого объема лингвистических знаний и были неэффективны. Позже исследователи перешли на автоматическое обнаружение шаблонов для извлечения отношений. Полуавтоматические методы, такие как DIPRE, KnowItAll и Snowball, позволяли сократить ручной труд и повысить надежность шаблонов. Некоторые методы на основе правил также учитывают лексические и синтаксические особенности текста. MetaPAD [43] объединяет лексическую сегментацию и кластеризацию синонимов для создания информативных и точных шаблонов отношений.

Для извлечения открытых связей также были разработаны модели глубокого обучения. Обычно эти модели основаны на архитектуре кодер-декодер. Copy Attention [44] использует механизм копирования слов из входной последовательности в выходную с помощью нейронного подхода. IMOJIE [45] улучшает внимание к копированию с помощью структур BERT-LSTM и включает в себя схему агрегации для итеративного извлечения отношений. Другой подход — перенести знания из контролируемых данных в модель, чтобы она могла извлекать неконтролируемые связи. Wu и его коллеги [46] разработали решение на основе метрического обучения, которое сочетает в себе реляционную сиамскую сеть (RSN) и стратегию кластеризации для обнаружения новых фактов.

В современной компьютерной лингвистике стратегии совместного использования параметров являются ключевым элементом в архитектуре нейронных сетей, предназначенных для решения множества задач обработки естественного языка. В современных исследованиях обработки естественного языка, Miwa и Bansal [47] предложили новаторский подход к интеграции признаков зависимостей. Они использовали комбинацию двунаправленных долгосрочных кратковременных памятей (Bi-LSTM) и двунаправленных деревьев LSTM (Bi-TreeLSTM) для улучшения распознавания именованных сущностей (NER) и отношений (RC). Эта методика позволяет более эффективно учитывать структурные и семантические связи в тексте. В то время как другие модели исследуют более деликатные стратегии для распределения признаков между различными задачами, подход Miwa и Bansal выделяется своей способностью обрабатывать сложные зависимости в данных. Например, модель GraphRel [48] использует двухэтапный механизм внимания для выделения взаимодействий между задачами на соответствующих уровнях двунаправленных графовых сверточных сетей (BiGCN). Фреймворки GCN объединяют граф зависимостей с графом отношений-сущностей для использования возможностей глубокого обучения и захвата сложных взаимосвязей в тексте.

Заключение

Представленная работа носит методологически-обзорный характер, представляя наряду с методикой построения программных систем графов знаний, обзор типовых подходов и соответствующих работ, реализующих шаги необходимые для построения программной системы для автоматического создания графа знаний. В рамках одной работы невозможно охватить все подходы к решению задач приводящих к построению графов знаний, поскольку данное направление динамично развивается, но авторы надеются, что приведенная обширная библиография поможет заинтересованным лицам в дальнейшем изучении проблемы.

На базе рассмотренной методики, авторами создан собственный алгоритм построения графа знаний и предложена комплексная метрика оценки успешности построения графа, но ее изложение будет темой следующей статьи.

ЛИТЕРАТУРА

1. X. Wu, J. Wu, X. Fu, J. Li, P. Zhou, and X. Jiang, «Automatic knowledge graph construction: A report on the 2019 ICDM/ICBK contest», in ICDM, 2019, pp. 1540–1545, 2019.
2. S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, «A survey on knowledge graphs: Representation, acquisition, and applications», IEEE Trans. Neural Networks Learn. Syst., vol. 33, no. 2, pp. 494–514, 2022.
3. L. Ehrlinger and W. Wöß, «Towards a definition of knowledge graphs», in SEMANTiCS, SuCCESS'16, 2016, vol. 1695 of CEUR Workshop Proceedings, 2016.
4. N. Kushmerick, «Wrapper induction: Efficiency and expressiveness», Artificial intelligence, vol. 118, no. 1-2, pp. 15–68, 2000.
5. D. Buttler, L. Liu, and C. Pu, «A fully automated object extraction system for the world wide web», in Proc. ICDCS,2001, pp. 361–370, 2001.
6. S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z.G. Ives, «Dbpedia: A nucleus for a web of open data», in ISWC + ASWC, 2007, vol. 4825, pp. 722–735, 2007.
7. F.M. Suchanek, G. Kasneci, and G. Weikum, «Yago: a core of semantic knowledge,» in Proc. WWW, 2007, pp. 697–706, 2007.
8. B.M. Sundheim, «The message understanding conferences», in TIPSTER TEXT PROGRAM PHASE II: Proceedings of a Workshop held at Vienna, 1996, pp. 35–37, 1996.
9. G. Zhou and J. Su, «Named entity recognition using an hmm-based chunk tagger», in Proc. ACL, 2002, pp. 473–480, 2002.
10. J.R. Finkel, T. Grenager, and C.D. Manning, «Incorporating non-local information into information extraction systems by gibbs sampling,» in Proc. ACL, 2005, pp. 363–370, 2005.
11. A. Finn and N. Kushmerick, «Multi-level boundary classification for information extraction», in Proc. ECML, 2004, vol. 3201, pp. 111–122, 2004.
12. R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P.P. Kuksa, «Natural language processing (almost. from scratch», J. Mach. Learn. Res., vol. 12, pp. 2493–2537, 2011.

13. E. Strubell, P. Verga, D. Belanger, and A. McCallum, «Fast and accurate entity recognition with iterated dilated convolutions», in Proc. EMNLP, 2017, pp. 2670–2680, 2017.
14. L. Luo, Z. Yang, P. Yang, Y. Zhang, L. Wang, H. Lin, and J. Wang, «An attention-based bilstm-crf approach to document-level chemical named entity recognition», Bioinform., vol. 34, no. 8, pp. 1381–1388, 2018.
15. A. Z. Gregoric, Y. Bachrach, P. Minkovsky, S. Coope, and B. Maksak, «Neural named entity recognition using a self-attention mechanism», in ICTAI, 2017, pp. 652–656, 2017.
16. A. Cetoli, S. Bragaglia, A.D. O'Harney, and M. Sloan, «Graph convolutional networks for named entity recognition», in Proc. TLT, 2018, pp. 37–45, 2018.
17. C. Dogan, A. Dutra, A. Gara, A. Gemma, L. Shi, M. Sigamani, and E. Walters, «Fine-grained named entity recognition using elmo and wikidata», CoRR, vol. abs/1904.10503, 2019.
18. M. Liu, Z. Tu, T. Zhang, T. Su, X. Xu, and Z. Wang, «Ltp: A new active learning strategy for crf-based named entity recognition», Neural Processing Letters, pp. 1–22, 2022.
19. I. Yamada, A. Asai, H. Shindo, H. Takeda, and Y. Matsumoto, «LUKE: deep contextualized entity representations with entity-aware self-attention», in EMNLP, 2020, pp. 6442–6454, 2020.
20. S. Zhang, K. Duh, and B. V. Durme, «Fine-grained entity typing through increased discourse context and adaptive classification thresholds», in Proc. *SEM@NAACL-HLT, 2018, pp. 173–179, 2018.
21. C. Moon, P. Jones, and N.F. Samatova, «Learning entity type embeddings for knowledge graph completion», in Proc. CIKM, 2017, pp. 2215–2218, 2017.
22. J. Hao, M. Chen, W. Yu, Y. Sun, and W. Wang, «Universal representation learning of knowledge bases by jointly embedding instances and ontological concepts», in KDD, 2019, pp. 1709–1719, 2019.
23. G. Limaye, S. Sarawagi, and S. Chakrabarti, «Annotating and searching web tables using entities, types and relation-ships», Proc. VLDB Endow., vol. 3, no. 1, pp. 1338–1347, 2010.
24. C. S. Bhagavatula, T. Noraset, and D. Downey, «Tabel: Entity linking in web tables», in ISWC, 2015, Proceedings, Part I, vol. 9366, pp. 425–441, 2015.
25. Y. Guo, W. Che, T. Liu, and S. Li, «A graph-based method for entity linking», in IJCNLP, 2011, pp. 1010–1018, 2011.
26. W.M. Soon, H.T. Ng, and C.Y. Lim, «A machine learning approach to coreference resolution of noun phrases», Comput. Linguistics, vol. 27, no. 4, pp. 521–544, 2001.
27. M. Recasens, M. de Marneffe, and C. Potts, «The life and death of discourse entities: Identifying singleton mentions», in NAACL-HLT, Proceedings, 2013, pp. 627–633, 2013.
28. M.A. ur Rahman and V. Ng, «Supervised models for coreference resolution», in Proc. EMNLP, 2009, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 968–977, 2009.
29. V. Stoyanov and J. Eisner, «Easy-first coreference resolution», in Proc. COLING ,2012, pp. 2519–2534, 2012.
30. J. Cai and M. Strube, «End-to-end coreference resolution via hypergraph partitioning», in Proc. COLING, 2010, pp. 143–151, 2010.
31. D L. Bean and E. Riloff, «Unsupervised learning of contextual role knowledge for coreference resolution», in HLT–NAACL, 2004, pp. 297–304, 2004.
32. E.R. Fernandes, C.N. dos Santos, and R.L. Milidiú, «Latent structure perceptron with feature induction for unrestricted coreference resolution», in EMNLP-CoNLL, ACL, 2012, pp. 41–48, 2012.
33. X. Xi, G. Zhou, F. Hu, and B. Fu, «A convolutional deep neural network for coreference resolution via modeling hierarchical features», in IScIDE, 2015, Revised Selected Papers, Proceedings, Part II, vol. 9243, pp. 361–372, 2015.
34. J. Wu and W. Ma, «A deep learning framework for coreference resolution based on convolutional neural network», in ICSC, 2017, pp. 61–64, 2017.
35. S. Wiseman, A. M. Rush, and S. M. Shieber, «Learning global features for coreference resolution», in NAACL-HLT, 2016, pp. 994–1004, 2016.
36. J. Gu, Z. Ling, and N. Indurkhya, «A study on improving end-to-end neural coreference resolution», in NLP-NABD, 2018, Proceedings, vol. 11221, pp. 159–169, 2018.
37. K. Clark and C.D. Manning, «Deep reinforcement learning for mention-ranking coreference models», in Proc. EMNLP, 2016, pp. 2256–2262, 2016.
38. G. Durrett and D. Klein, «Easy victories and uphill battles in coreference resolution», in Proc. EMNLP, 2013, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 1971–1982, 2013.
39. S. Martschat and M. Strube, «Latent structures for coreference resolution», Trans. Assoc. Comput. Linguistics, vol. 3, pp. 405–418, 2015.
40. V. Mulwad, T. Finin, and A. Joshi, «Semantic message passing for generating linked data from tables», in ISWC, 2013, Proceedings, Part I, vol. 8218, pp. 363–378, 2013.
41. Z. Chen and M.J. Cafarella, «Automatic web spreadsheet data extraction,» in 3RD International Workshop on Semantic Search over the Web, SSW '13, 2013, pp. 1:1–1:8, 2013.
42. J. Zhu, Z. Nie, X. Liu, B. Zhang, and J. Wen, «Statsnowball: a statistical approach to extracting entity relationships», in Proc. WWW, 2009, pp. 101–110, 2009.
43. M. Jiang, J. Shang, T. Cassidy, X. Ren, L. M. Kaplan, T. P. Hanratty, and J. Han, «Metapad: Meta pattern discovery from massive text corpora», in KDD, 2017, pp. 877–886, 2017.
44. L. Cui, F. Wei, and M. Zhou, «Neural open information extraction», in Proc. ACL, 2018, Volume 2: Short Papers, pp. 407–413, 2018.
45. K. Kolluru, S. Aggarwal, V. Rathore, Mausam, and S. Chakrabarti, «Imojie: Iterative memory-based joint open information extraction», in Proc. ACL,2020, pp. 5871–5886, 2020.
46. R. Wu, Y. Yao, X. Han, R. Xie, Z. Liu, F. Lin, L. Lin, and M. Sun, «Open relation extraction: Relational knowledge transfer from supervised data to unsupervised data», in Proc. EMNLP-IJCNLP, 2019, pp. 219–228, 2019.
47. M. Miwa and M. Bansal, «End-to-end relation extraction using lstms on sequences and tree structures», in Proc. ACL, 2016, Volume 1: Long Papers, 2016.
48. T. Fu, P. Li, and W. Ma, «Graphrel: Modeling text as relational graphs for joint entity and relation extraction», in Proc. ACL, 2019, Volume 1: Long Papers, pp. 1409–1418, 2019.