Entity Linking
Связывание именованных сущностей

Аленичева Алиса

Аннотация—В данном проекте была поставлена задача исследования подходов решения проблемы связывания именованных сущностей: поиска сущности в тексте и соотнесение его с записью в базе знаний. Были изучены нейросетевые модели, используемые для решения задачи для английского языка (Deeptype, Deep-ed и End2End) а так же рассмотрена адаптация модели End2End на русский язык.

I. Постановка задачи

ENTITY linking = Entity Recognition + Entity Disambiguation.

Задача связывания именованных сущностей состоит в определении упоминания (**mention**) сущности (**entity**) в тексте и соотнесения с определенной записью в базе знаний (**knowledge base**).

При анализе поставленной задачи основной акцент ставился на поиск решения проблемы многозначности (**disambiguation**) для выделенных упоминаний: например, слово «Иванов» в различном контексте может обозначать как личность (фамилия - Иванов), так и место (село Иванов), а так же на поиск абстрактных упоминаний, не имеющих внутри себя самого названия сущности (столица Франции)

Практические применения: разметка новостной ленты, чат-боты (ответы на вопросы), смысловой поиск.

II. Анализ подходов

1) Классический подход
   Классический подход включает в себя статистические расчеты количества **LinkCount** – количество раз, которое упоминание **m** указывает на сущность **e** и **Coherence** – количество раз, которое сущность $\mathbf{e_1}$ встречается рядом с сущностью $\mathbf{e_2}$. Минусы – дискретность (если упоминание ни разу не встречалось в тренировочном датасете – нет возможности его детектировать в тестовых текстах).
2) Нейронный подход
   Нейронный подход делает систему гибче и ускоряет обучение: использование рекуррентных сетей, которые проецируют mentions с контекстом на некоторое векторное пространство позволяет оперировать большим количеством упоминаний и строить более сложные связи с сущностями.

1) DeepType [3]
   Основная идея определения, к какой именно сущности принадлежит упоминание – на базе тренировочного датасета выбрать 100 категорий, к которым могут принадлежать сущности. (Для каждой сущности, т.е. страницы википедии существует информация о категориях). Для слова и его контекста можно обучить сеть, которая делает проекцию слова и контекста на 100-мерный бинарный вектор принадлежности к категориям. Тренировочный сет создается на базе **knowledge graph** википедии. При помощи этого маппинга можно определить, в каком контексте, например, ягуар будет животным, а в каком – маркой машины.

2) Deep Joint Entity Disambiguation with Local Neural Attention/Deep-ed [1]
   Идея данной работы – на базе взятого **wordembedding**'а, а так же информации **LinkCounts** из Википедии, датасета **Crosslinks** и **AIDA** натренировать вектора сущностей (**entity embeddings**), которые будут располагаться в том же

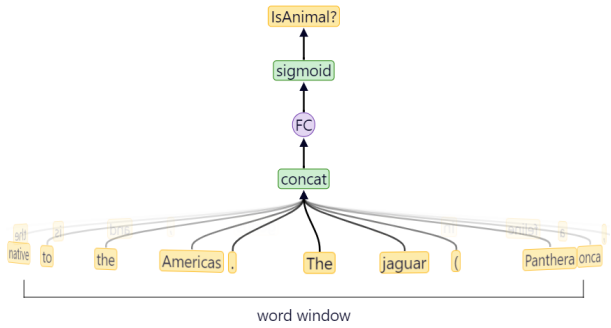

Рис. 1. DeepType, схема сети для бинарной классификации в системе типов

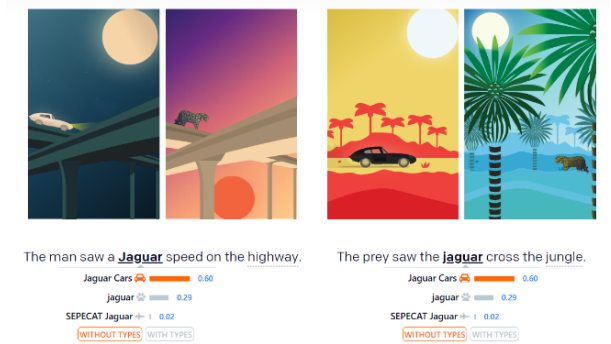

Рис. 2. DeepType, пример многозначности упоминания

пространстве, что и вектора слов. При наличии **word** и **entity embeddings**, из списка **LinkCounts** для кандидата отбираются наиболее вероятные соответствующие сущности и затем обучаются матрицы преобразований, выявляющие самые значимые слова в контексте. На выходе выдаются **scores** для кандидатов сущностей.

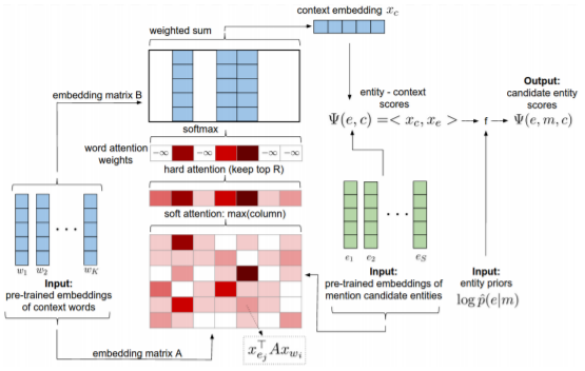

Рис. 3. Deep-ed, схема модели

3) End-to-End Neural Entity Linking/End2End [2]
Предложенный подход использует идею векторов сущностей из deep-ed проекта, но добавляет тренировку **bi – directional LSTM**, преобразуя слова в вектора на основе **char embeddings**, таким образом учитывая контекст. Из **LinkCounts** словаря выбираются кандидаты сущностей для данного упоминания и в качестве ответа выдается ближайший к упоминанию сущностный вектор.

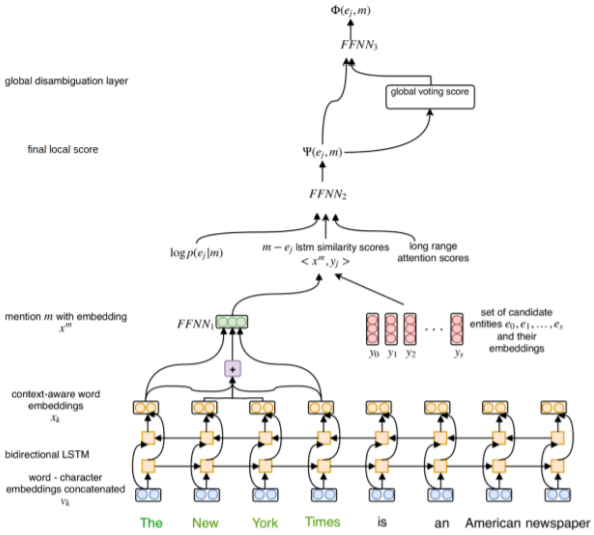

Рис. 4. End2End, схема модели

III. Сравнительная характеристика моделей

В то время как модели **DeepType** и **Deep – ed** решают задачу выбора конкретной сущности из множества значений для уже выделенных в тексте упоминаний, **End2End** модель ставит перед собой задачу как выделения, так и связывания сущностей. При этом **DeepType** для учитывания контекста упоминания при выборе сущности использует нейросеть на основе **LSTM**, в то время как **Deep – ed** тренирует вектора сущностей в векторном пространстве слов а затем тренирует матрицы преобразования векторов сущностей и упоминаний с контекстом. **End2End** модель строится на базе натренированных в **Deep – ed** векторов сущностей, но учет контекста производит при

Таблица I
Сравнительная характеристика моделей

	DeepType	Deep-ed	End2End
Язык	Python	Lua	Python
Тип модели	Нейросетевая	Векторная	Векторная+ Нейросетевая
Тип задачи	Dis-ambiguation	Dis-ambiguation	End2End
Датасет	Wikipedia	AIDA CoNLL-YAGO	AIDA CoNLL-YAGO
База знаний	Wikipedia	Wikipedia	Wikipedia
Micro-Precision	94.88	92.22	-
Micro-F1-strong	-	-	89.4
Micro-F1-strong	-	-	86.6

помощи $\mathbf{bi-directionalLSTM}$. Для тренировки **DeepType** конструировался датасет на основе дампа текста Википедии, а так же гиперссылок и их ближайшего констекста. $\mathbf{Deep-ed}$ и **End2End** тренировались на классическом размеченном для задачи связывания именованных сущностей датасете $\mathbf{AIDACONLL-YAGO}$.

IV. Адаптация на русский язык

Основной проблемой использования нейросетевого подхода в задаче EL является отсутствие тренировочного размеченного датасета на русском языке. При этом в xml-дампе Википедии есть выделенные гиперссылки формата Mention, где символы после тэга **href** - название статьи Википедии, на которое ссылается упоминание, а словосочетание перед тегом $< /\mathbf{a} >$? соответственно - упоминание данной сущности. В ходе проекта был написан скрипт, позволяющий обработать xml-дамп русскоязычной Википедии для получения тренировочного датасета с размеченными сущностями в формате AIDA датасета.

Кроме того, в End2End подходе при выборе кандидатов сущностей конкретного упоминания используются величины **LinkCounts** p(e|m) и словарь mention: entities. После обработки 865 статей Википедии было получено 42746 упоминаний, при этом распределение многозначностей этих упоминаний показано на Рис.5.

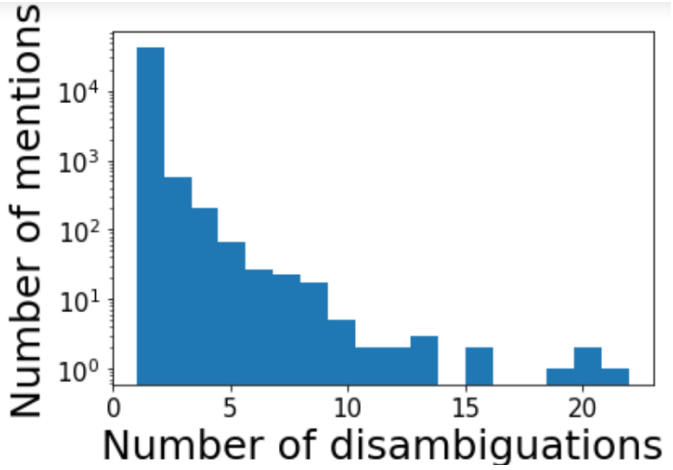

Рис. 5. Распределение количества сущностей, соответствующих определенному упоминанию

V. Возможные улучшения и пути развития решения

1) Поскольку Deep-ed вектора сущностей невозможно натренировать для русского языка из-за несовместимости кода модели на языке программирования Lua с кириллицей, можно использовать уже натренированные вектора слов и сущностей в едином пространстве Wikipedia2vec. 2) Для применения задачи к разметке новостной ленты можно использовать в качестве тренировочного датасета не только Википедию, но и дамп ресурса WikiNews. 3) Для выделения упоминаний из текста возможно использование мультиязычной BERT модели.

Список литературы

[1] Octavian-Eugen Ganea and Thomas Hofmann. Deep joint entity disambiguation with local neural attention. CoRR, abs/1704.04920, 2017.
[2] Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. End-to-end neural entity linking. CoRR, abs/1808.07699, 2018.
[3] Jonathan Raiman and Olivier Raiman. Deeptype: Multilingual entity linking by neural type system evolution. CoRR, abs/1802.01021, 2018.