Разработка метода выделения именованных сущностей

Владислав Кораблинов

26.11.2019

1 Постановка задачи

В данной практической работе перед нами ставилась задача обнаружения в тексте именованных сущностей двух категорий - люди (персоналии) и организации. На вход подается набор предложений, для каждого предложения необходимо вывести список троек *(позиция, длина, тег)*, каждая из которых означает, что в данном предложении на указанной позиции находится слово (токен) указанной длины, являющееся частью обнаруженной именованной сущности. Если эта сущность является персоналией, тег принимает значение *PERSON*, если организацией - *ORG*.

В качестве примера модельной разметки предлагается 2453 размеченных предложения. Тестирование проводится на наборах из 500 предложений.

1.1 Метрики качества

Для оценки качества используется F1-мера по объединению множеств размеченных токенов.

Из-за того, что полученное решение использует несколько эвристик, в ходе его разработки было проведено много экспериментов по опеределению хорошо работающих правил. В таких случаях, к сожалению, не проводилось логирования результатов, поэтому в при описании результатов будут даваться примерные полученные оценки. Для моделей, которые являлись окончательными в рамках конкретного подхода, будут даны минимальная, средняя и максимальная оценка по 5 наборам данных.

2 Общая модель решения

Мы попробуем построить решение на основе словарей и эвристик. Таким образом, нашей основной задачей является обнаружение в предложении подпоследовательностей, которые отвечают какому-либо известному нам объекту одного из двух классов.

2.1 Потокенная модель

Наиболее простым решением может быть классификация каждого токена в отдельности. В этом случае мы просто ищем такой токен в одном из словарей. В этом случае у нас возникают две существенные сложности. Многие слова могут в зависимости от контекста быть как частью наименования организации, так и не иметь к ним никакого отношения. Если мы поместим такие слова в словарь, то получим достаточно много ложноположительных срабатываний. С другой стороны, если мы не хотим добавлять их в словарь, мы должны научиться отделять их от специфичных слов в наборе обучающих предложений. Так как разметка дана для каждого токена в отдельности, сделать это достаточно затруднительно.

2.2 Оконная модель

Так как главной проблемой потокенной модели является полное игнорирование контекста, стоит попробовать определять тег для последовательности токенов. Мы будем перебирать все куски последовательных токенов предложения, и для каждого определять, имеется ли у нас такой текст в одном из словарей.

3 Предобработка

Каждое предложение сначала было преобразовано в список токенов, каждый из которых отвечал регулярному выражению '\w+'. Для всех токенов была создана лемматизированная с помощью *Mystem* копия с сохранением капитализации первой буквы.

4 Словари

Для построения словарей использовалось несколько различных источников.

4.1 Данные для обучения

В первую очередь, словари были пополнены размеченными токенами из предложенных данных. При этом для персоналий каждый токен добавлялся отдельно, так как эти токены почти всегда являются именами, фамилиями или должностями, а такие слова действительно минимально контекстно-зависимы. Для организаций добавлялись соединенные пробелом последовательные токены с тегом *ORG*.

4.2 Коллекция Named Entities 3

Также словари были аналогичным образом пополнены размеченными данными проекта Named Entities.

4.3 Wikidata

Наиболее интересным источником была база знаний Wikidata. Записи этой базы представляют собой тройки *сущность, отношение, сущность*. Мы будем использовать два очень полезных отношения - *instance of* и *subclass of*. Первое говорит о том, что сущность является частным случаем другой, категориальной сущности, а второе - о том, что одна категориальная сущность является подклассом другой категориальной сущности. Далее станет понятно, как использовать эти отношения для получения интересующих нас данных.

Для определения персоналий из Wikidata были извлечены все сущности, для которых существовала тройка *сущность, instance of, человек*. Для каждой сущности в Wikidata существует список алиасов - различных имен, которые даны этой сущности. Поэтому для каждой найденной сущности мы получили список ее имен, таким образом, мы получили список всех известных базе людей.

Организации извлекались в два этапа. Сначала нужно было выяснить, какая абстракция Wikidata совпадает с понятием организации, принятым конкретно в этом задании. Путем экспериментов было установлено, что оптимальными для нашей задачи является сущности Wikidata бизнес и предприятие. Сначала были извлечены все сущности, которые были связаны с указанными двумя отношнием subclass of и найдены все их алиасы. Таким образом, были получены слова, выражающие абстрактные организации, такие как *банк*, *компания* или *ОАО*. Далее были извлечены все сущности, для которых существовала следующая цепочка: *(сущность, instance of, категория), (категория, subclass of*, бизнес | предприятие)*. Звездочка означает, что цепочка отношения может быть любой длины. Для найденных сущностей также были извлечены все их алиасы, в результате чего было получено около 30000 новых названий различных организаций.

5 Эвристические правила

В ходе экспериментов было опробовано несколько различных эвристических правил.

5.1 Сработавшие

Если токен размечен как персоналия и следующий за ним токен начинается с заглавной буквой, то его тоже следует разметить как персоналию. Это объясняется тем, что часто мы находим в

словаре имя, но не находим фамилию. Кроме того, в обучающем наборе практически все персоналии начинаются с заглавной буквы, а после них идет слово со строчной.

Игнорировать короткие слова в начале предложения. Часто предложения начинаются с коротких слов, которые в другом контексте могут быть сокращением части название именованной сущности, поэтому если мы уберем такие слова из словаря, то сильно проиграем. Оказывается выгоднее пропускать короткие слова в начале.

Размечать как организацию последовательность длины больше 2, состоящую из слов латиницей . Так как нам даны предложения на русском языке, в подавляющем большинстве случаев слова латиницей в них являются именами собственными. Судя по обучающему набору, эти предложения являются чем-то вроде новостных заголовков, поэтому почти всегда длинные последовательности латинских токенов в нем размечены как организации. Отметим, что эта эвристика стала вносить меньше вклада после хорошего расширения словаря.

5.2 Не сработавшие

Пытаться разметить каждое из слов, начинающихся с заглавной буквы, одним из тегов. Ясно, что такая попытка очнеь наивна, потому что в текстах будет встречаться достаточно имен собственных других классов, но на всякий случай стоило попробовать.

Игнорировать капитализацию при нахождении названий организаций. Попытка была мотивирована тем, что иногда в названия организаций входят нарицательные существительные, начинающиеся с заглавной буквы. Однако при стирании капитализации терялось много информации.

6 Ход экспериментов

Сначала внимание было сфокусировано на нахождении персоналий. Первое, самое простое, решение с использованием потокенного подхода на словаре, составленном только из обучающего набора, получало оценку около 38. Еще раз отметим, что токены, соответствующие персоналиям действительно в большинстве случаев являются контекстно независимыми, поэтому для них не был применен оконный подход. Однако, оконный подход был применен после добавления в словарь сущностей из Wikidata, для них совпадения искались по 2 или 3 подряд идущим токенам, начинающимся с заглавной буквы, если такие были, причем с использованием нечеткого поиска (это было возможным благодаря тому, что на компьютере к моменту начала решения уже был построен индекс со всеми алиасами Wikidata). Это заметно увеличило оценку, примерно до 52 пунктов, эвристика продолжения тега на следующий токен с первой заглавной буквой повысила результат еще на 1-2 пункта.

Для персоналий также была сделана попытка интегрировать внешний словарь имен, собранный, по всей видимости, из социальных сетей, но он оказался слишком грязным, из-за чего происходило огромное количество ложноположительных срабатываний.

Дальнейшая работа полностью состояла в выделении организаций. Сначала был применен простой потокенный подход с словарем, посторенным на обучающем наборе. Этот метод был измерен, и были получены следующие результаты:

Мин.	**Среднее**	**Макс.**
62.23	63.68	65.32

В процессе изучения выдачи метода стало ясно, что для организаций потокенный подход показывает себя довольно плохо, например, часто в качестве организаций размечались государства, хотя судя по обучающему набору, они в рамках этой задачи таковыми не являлись. Это происходило из-за того, что названия государств часто входят в названия организаций, и множество ошибок возникало из-за потери сопутствующих слов. Поэтому сначала было произведено слияние последовательных отмеченных как организации токенов.

Тем не менее, некоторые слова, как мы указывали раньше, действительно почти всегда означают организации. Для решения этой проблемы была добавлена первая часть данных Wikidata, описывающих категории организаций. Кроме того, была применена эвристика со словами латиницей (как выяснилось позже, она дает прирост около 1.5). Новое решение с использованием оконного подхода дало следующий результат:

	Мин.	**Среднее**	**Макс.**
Потокенный подход для организаций	62.23	63.68	65.32
Оконнный подход для организаций + категории из Wikidata	67.96	68.61	69.55

Такой подход действительно дал ощутимый прирост. Наконец, была добавлена вторая часть данных из Wikidata, состоящия из большого количества названий известных организаций. Алгоритм поиска при этом не менялся. Это итоговое решение показало такой результат:

	Мин.	**Среднее**	**Макс.**
Потокенный подход для организаций	62.23	63.68	65.32
Оконнный подход для организаций + категории из Wikidata	67.96	68.61	69.55
Оконнный подход для организаций + все данные Wikidata	71.07	71.97	72.95

Расширение словаря организаций тоже дало ожидаемый заметный прирост. Справделиво, что именно в этот момент эвристика слов латиницей перестала вносить значимый вклад в оценку.

После этого была также произведена попытка добавить все сущности Wikidata, являющиеся частным случаем сущности организация, и хотя таких было очень много, результаты сильно ухудшились, потому что снова стало происходить много ложноположительных срабатываний.

7 Нереализованные идеи

Разумеется, задачу разметки именованных сущностей успешно решают с помощью методов машинного обучения. Для нужно посмотреть на эту задачу как на задачу классификации по классам тегов. Но есть довольно большая сложность в том, что сразу непонятно, какие признаки важны в этой задаче и как их извлекать. Этот факт в сочетании с тем, что, как мы поняли, для определения тега очень важен контекст токена, подводит к тому, что для этой задачи должна хорошо подойти какая-нибудь рекуррентная модель нейронной сети со слоем эмбеддингов слов на входе. К сожалению, времени и навыков было недостаточно для попытки реализации такой модели.

8 Выводы

По итогам проделанной работы сделаем несколько выводов:

1. Для поиска персоналий можно довольно успешно пользоваться решением с использованием словарей.
2. Для выделения сущностей для классов, которые не имеют ярко выраженных признаков, становится довольно тяжело придумать качественный набор правил.
3. При использовании словаря очень важно следить за его точностью, пожалуй, важнее, чем за полнотой.