Выделение названий географических объектов при обработке текстовых сообщений

А. М. Лопушанский[1], Я. А. Бекенева[2]
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)
[1]sashalopy@mail.ru [2]yabekeneva@etu.ru

Аннотация. **Выявление нештатных ситуаций локального характера и их последующее геокодирование предполагает выделение названий географических объектов в текстовых сообщениях. В качестве одного из инструментов решения такой задачи была предложена библиотека natasha, которая позволяет выделять именованные сущности в тексте. Однако проведенное исследование показало, что такой инструмент выделяет сущность как имя собственное, что создает сложности при определении точного адреса, где произошла нештатная ситуация.**

Ключевые слова: обработка текстов, именованные сущности, геокодирование, географические объекты

I. Введение

Извлечение именованных сущностей (англ. Named Entity Recognition, NER) – это процесс выделения из текста упоминаний конкретных объектов, таких как имена людей, названия организаций, местоположения, даты, времени и т.д. Эта информация может использоваться для определения предмета обсуждения в тексте записей в социальных сетях. Перед выделением именованных сущностей желательно выполнить предварительную обработку текста, включающую в себя токенизацию, удаление стоп-слов и лемматизацию. При этом нежелательно изменение регистра текста, так как инструменты выделения сущностей чувствительны к расположению заглавных букв в каждом слове [1]. Так, словами, потенциально отмеченными алгоритмом, являются слова, написанные с большой буквы, состоящие целиком из заглавных букв, или содержащие заглавные буквы в середине. На основе контекста слова используемый инструмент должен определить, является ли это слово именованной сущностью, и если да, то к какому классу она относится, например, имя, организация или местоположение.

Результатом проведения NER является набор извлеченных из текста именованных сущностей и их классов, которые могут быть использованы для дальнейшего анализа текста [2]. Например, эти данные могут быть использованы для определения ключевых участников события в новостной статье или для анализа общественного мнения в социальных сетях относительно конкретной компании.

Одна из самых точных библиотек, позволяющих провести NER-разметку – natasha. Для разметки текста внутри библиотеки используется модель Slovnet NER. Рассмотрим результат работы инструмента для выделения именованных сущностей на примере следующего текста: «Басманный суд Москвы отправил Татьяну под арест на два месяца». Для данного текста будет выделено три сущности: «Басманный суд» – организация, «Москва» – местоположение, «Татьяна» – персона. Сущности приводятся после выделения к лемме, с целью облегчения поиска связанных текстов, содержащих те же сущности, но в другом падеже.

Одна из задач, которую позволяет решить NER – определение, какое именно место описывается в анализируемом тексте. Выделение сущностей типа локация позволяет разметить новость на карте. Особенности размещения новостей на карте описаны далее.

II. Геокодирование

Геокодирование – это процесс преобразования адреса или названия места в координаты на карте (широту и долготу) и наоборот. Использование сервисов геокодирования необходимо чтобы отметить какое-либо географическое место на карте. Также существует понятие обратного геокодирования – это процесс преобразования координат какого-либо места на карте в его адрес. Обычно в данной ситуации возвращается адрес, максимально точно описывающий отмеченное место, иногда это может быть номер дома и название улицы, иногда – название области страны, если в указанном месте ничего не находится.

Из анализируемого текста возможно извлечь название какой-либо локации, это может быть название города, улицы, района или целая страна. Размещение анализируемого текста на карте может быть полезно по нескольким причинам: во-первых, отображение места на карте более наглядно, чем текстовое описание, во-вторых, преобразование текста в координаты позволяет проводить анализ географических закономерностей в тексте. Например, при анализе новостей возможно определять, о каких местах чаще упоминают в тексте, какие страны в центре внимания, о каком городе в данный момент больше новостей, на какой улице чаще бывают аварии.

Одна из проблем при процессе геокодирования с использованием извлеченных из текста именованных сущностей – ошибки при лемматизации названий улиц, районов, городов. При попытке закодировать адрес с названиями, не приведенными к нормальному виду, обычно возвращается отрицательный результат – ненайденная локация, либо адрес некорректно распознанного местоположения. Один из способов избежать ошибок при лемматизации и геокодировании адресов – составление списка с названиями мест, которые могут быть найдены.

Для того чтобы составить список всех улиц, которые могут быть найдены пользователем, и таким образом повысить вероятность отметки верного места на карте, необходимо сузить область, в которой происходит поиск. Для этого определим зону, в которой будет производиться поиск. Для наглядности можно

ограничиться одним городом – собрать все адреса Санкт-Петербурга. Перечень всех наименований улиц города установлен постановлением правительства Санкт-Петербурга «О Реестре наименований элементов улично-дорожной сети (за исключением автомобильных дорог федерального значения), элементов планировочной структуры, территорий зеленых насаждений общего пользования, расположенных на территории Санкт-Петербурга». Все названия, перечисленные в постановлении, представлены в виде списка на сайте-справочнике. Использование справочника оправдано простотой преобразования перечня в вид, пригодный для использования разрабатываемым инструментом.

Для сбора данных с сайта используется библиотека BeautifulSoup, назначение которой – синтаксический разбор HTML файлов. Использование этого инструмента позволяет быстро отфильтровать в файле содержимое разметки [3]. На рассматриваемом сайте каждый адрес записан в одном классе тэгов, таким образом, указав класс тэга, можно легко собрать все адреса. Вопрос хранения собранных данных будет рассмотрен далее в этом разделе.

Хранение списка адресов позволяет уточнять выделенную сущность-локацию, и так избегать ошибок при вводе адреса для геокодирования. По необходимости масштаб области поиска может быть изменен, для этого достаточно дополнить список адресами новой рассматриваемой области.

Вопрос геокодирования достаточно сложен и требует дополнительной проработки. Одна из проблем, которую следует решить – как отображать место, упоминаемое в анализируемом тексте, если дано только название улицы без уточнений дома. Иногда отметки точкой может быть недостаточно, особенно при указании длинной улицы. Необходимо отобразить ее целиком, чтобы дать пользователю представление, чем ограничена зона, описываемая в тексте.

Чтобы получить описание геометрии улицы с помощью OpenStreetMap Nominatim, необходимо выполнить запрос к API. Запрос осуществляется HTTP-методом GET. Указывается URL адрес Nominatim, сопровождаемый параметрами запроса, обычно записывается название искомого места, устанавливается лимит возвращаемых результатов, устанавливается флаг фильтрации дублируемых результатов, указывается вид, в котором требуется получить ответ.

После совершения запроса сайт возвращает json-файл, содержащий список всех областей, подходящих под запрос. Длинные улицы в OpenStreetMap разделены на маленькие отрезки, каждый из которых является отдельной областью. Чтобы отобразить контур улицы необходимо собрать все полученные отрезки этой улицы и последовательно отрисовать их на карте города.

Описанный выше метод позволяет отмечать на карте отдельные точки или отдельные улицы. Существует несколько сценариев, где этого недостаточно [4]. В анализируемых текстах могут упоминаться целые области, например, район города, название города или страны. Для того, чтобы дать пользователю представление о географическом расположении таких областей обычно достаточно отметить точкой центр искомой области, при этом нет необходимости отмечать административные границы этой области. Другая проблема – разметка пересечения улиц. Например, при анализе текстов о дорожной ситуации будут регулярно упоминаться перекрестки дорог, так как на перекрестке чаще происходят дорожно-транспортные происшествия. Данная проблема требует сложного решения, чтобы отметить только точку пересечения дорог, так как OpenStreetMap не хранит данные о пересечениях улиц. Для наглядности достаточно разметить на карте обе улицы, упоминаемые в тексте – пересечение будет визуально заметно.

В результате анализа инструментов геокодирования был выявлен наиболее подходящий для решения задачи – сервис Nominatim. Были проработаны вопросы разметки на карте сложных территорий, состоящих из нескольких частей. Таким образом, найденные с помощью выделения именованных сущностей локации можно доступно отобразить на карте для следующего анализа пользователем разрабатываемой программы.

Функция отметки происшествия на карте будет более полезной при описании в записи более конкретного адреса происшествия, а не только района. Лучший способ найти большое количество записей, в которых регулярно описываются нештатные ситуации – анализ постов в группах, посвященным ДТП в определенных городах. Рассмотрим такую группу, ориентированную на события исключительно в Санкт-Петербурге. Ограничение зоны позволяет автоматически исправлять ошибки при определении нормальной формы адреса, упомянутого в тексте.

На основании данных, указанных в приведенных выше таблицах, а также полагаясь на установленные ранее связи между сущностями, была разработана даталогическая схема данных, изображенная на рис. 1.

Рис. 1. Схема БД в SQLite

Рассмотрим, какая информация может помочь пользователю выявить пост, потенциально описывающий нештатную ситуацию. Одним из возможных показателей может быть частота отправки сообщений, которая возрастает при наступлении нештатного события [5]. В данном случае, имеется доступ не только ко времени создания поста, но и его характеристикам – реакции людей на него. Исходя из этого, было принято решение предпринять попытку отобразить пользователю только характеристики записей и не отображать их частоту. Это также связано с тем, что часть сообществ публикует записи не по мере возникновения новостей, а равномерно, в течение дня.

Интерфейс должен содержать основную информацию об анализируемой записи – текст поста, реакции людей на просматриваемую запись, статистика реакций на все записи за рассматриваемый промежуток. На графиках со статистикой всех записей необходимо отметить текущую просматриваемую запись, чтобы пользователь мог понимать, какие записи требуют внимания, а какие нет. Также пользователю нужно отобразить выделенные ключевые слова и дать возможность отобразить на карте места, упоминаемые в посте.

Учитывая все требования, был разработан интерфейс приложения. Для разработки использовалась библиотека PyQT6. Интерфейс представляет собой несколько окон, среди которых основное – окно с информацией обо всех постах. Также в программе создается окно для выбора группы, посты которой будут анализироваться, и окно просмотра карты с отмеченными координатами.

Окно с информацией о постах состоит из набора виджетов, каждый виджет выполняет отдельную функцию в работе приложения. Основной виджет содержит текст поста и выделенные ключевые слова. Еще один виджет позволяет разметить на карте обнаруженную в посте локацию. И, наконец, виджет, отображающий статистику постов в группе.

III. Эксперименты

При анализе записей в таких группах нет необходимости ориентироваться на статистику записей, так как практически каждая запись является описанием дорожно-транспортного происшествия. При описании места аварии обычно используется три основных способа объяснить, где конкретно случилось происшествие. Самый неточный способ, который не дает возможности понять, где конкретно случилась авария – указание названия улицы, площади или шоссе. Второй вариант – указание адреса дома, около которого случилось ДТП. И самый распространенный вариант – указание на перекрестке каких улиц случилась авария. Третий вариант часто встречается из-за того, что на перекрестках происходит наибольшее число аварий.

Рассмотрим результат работы программы для каждого варианта описания аварии. При поиске улицы без уточнения дома нельзя точно сказать, где случилось происшествие. В таком случае пользователю отмечается целиком зона, где могло случиться происшествие. Например, при анализе записи следующего содержания: «Двойное ДТП около 19 вечера на площади Конституции», будет выделено название места, некорректно приведенное к форме «Конституция». При сверке с составленным ранее списком всех названий мест в городе, с помощью нечеткого поиска было установлено, что нормальная форма названия места – «площадь Конституции». Исправленное название отправляется сервису геокодирования, и на карте размечается вся площадь Конституции.

При упоминании адреса дома, около которого случилось происшествие самый надежный вариант – отметить на карте дом, упоминаемый в записи, и оставить задачу определения конкретного участка дороги, на котором произошла авария, пользователю. Рассмотрим запись, содержащую следующий текст: «Известно, что трём женщинам и мужчине удалось спастись. Возгорание произошло в квартире дома 9, корпус 1, по проспекту Косыгина. Пожарные справились с огнём за 20 минут. По словам местных жителей, горевшая квартира имела плохую репутацию». Извлеченная сущность – «Косыгин», при сопоставлении с составленным ранее списком было уточнено название «проспект Косыгина». Однако в этом адресе указаны так же номер дома и корпуса, то есть, для данной записи должен быть извлечен адрес «проспект Косыгина дом 9 корпус 1».

При описании аварии на перекрестке в записях обычно встречается следующий шаблон: используется форма слова «перекресток», «пересечение» или «угол», далее следует два названия улиц, извлекаемых как сущности типа локация. Применив поиск такого шаблона можно отметить на карте обе улицы, тем самым изобразив их пересечение. Сложность разметки перекрестков с помощью Nominatim заключается в том, что сервис не хранит координаты перекрестков. Из-за этого в данном решении при поиске перекрестка отмечается две улицы, а не точка – перекресток.

Рассмотрим следующую запись: «Видео аварии на пересечении Лесного проспекта и Кантемировской улицы 10 апреля в 21:19. Дата и время на регистраторе неверные». В тексте встречается шаблон – «пересечение» и две улицы. При отметке на карте будут изображены обе улицы, и видно место их пересечения.

IV. Заключение

Разметка событий на карте – удобный способ визуализировать информацию, представленную в записи. Использование такого инструмента совместно с программами навигации может позволить строить маршруты, избегая упоминаемых в записях участков дороги. При анализе записей за длинный промежуток времени можно изучать закономерности, где чаще всего происходят происшествия. Для людей незнакомых с географией города данный инструмент поможет понять направление, где случилось происшествие. Код для разметки фигур на карте и их создания представлен в приложении В.

Разработанный инструмент требует дальнейшей доработки для повышения точности разметки событий на карте и улучшения качества распознавания названий мест. В данный момент инструменты библиотеки для анализа естественного языка natasha не позволяют проводить извлечение адресов в слабоструктурированных текстовых данных. Для обнаружения именованных сущностей нужно составлять дополнительные правила, которые повысят качество нахождения адресов в тексте.

Список литературы

[1] A. Araujo, M. Golo, R. Marcacini, “Opinion mining for app reviews: an analysis of textual representation and predictive models”, Automated Software Engineering, 2022,– Vol. 29.

[2] A. Kadhim, “Term Weighting for Feature Extraction on Twitter: A Comparison Between BM25 and TF-IDF”, ICOASE, 2019, pp. 124-128.

[3] X. Gao, J. Cao, Q. He et al. “A novel method for geographical social event detection in social media”, 5-th ACM International Conference Proceeding Series, New York, 2013, pp. 305-308.

[4] R. Lee, K. Sumiya, “Measuring geographical regularities of crowd behaviors for Twitter-based geo-social event detection”, Proceedings of the 2010 International Workshop on Location Based Social Networks, 2010, pp. 1-10.

[5] C.J. Hutto, E. Gilbert “VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text”, ICWSM, 2015.