МЕТОДЫ АВТОМАТИЗИРОВАННОЙ ГЕНЕРАЦИИ ПРОГРАММНОГО КОДА ПО ТЕКСТУ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Шпарута С.К., студент магистратуры 1-го курса кафедры информационно-аналитеских систем СПбГУ, sofish7184@gmail.com;
Научный руководитель: Графеева Н.Г., доцент кафедры информационно-аналитических систем СПбГУ, кандидат физико-математических наук, n.grafeeva@spbu.ru

Аннотация

В статье представлен обзор существующих решений на указанную тему, а также реализация нового приложения, которое переводит вопрос на естественном языке в SQL-запрос, выполняет его и выдаёт пользователю ответ.

Введение

В настоящее время множество приложений имеют естественно-языковой интерфейс. Алиса, Google-поиск, Яндекс-карты и многие другие программы позволяют нам «общаться» с ними на естественном для нас, людей, языке.

Поэтому, задача перевода из естественного языка в программный код очень актуальна в наши дни. Сейчас существует очень много различных статей и решений на эту тему. В частности, естественно-языковые интерфейсы к реляционным базам данным привлекают довольно много внимания.

Хотелось бы создать систему, которая транслировала бы вопросы с естественного языка в SQL-запрос. Это позволило бы очень многим людям, не владеющим техническими знаниями, делать различные действия с базами данных. В дополнение, большое удобство принесла бы быстрая скорость подобной системы.

В данной работе представлены:

- обзор существующих решений;
- описание базы данных достопримечательностей Санкт-Петербурга;
- описание приложения, которое переводит вопрос на естественном языке в запрос к базе данных и выдает пользователю ответ, .

Обзор существующих решений

Генерация SQL-запросов с использованием синтаксических зависимостей и метаданных на естественном языке

Авторы статьи **[6]** решают проблему перевода вопроса на естественном языке во «что-то понятное машиной» автоматическим способом, генерируя SQL-запросы, структура и компоненты которых соответствуют концепциям NL (выраженным как слова) и грамматическим зависимостям. Их новая идея заключается в том, как и где это сопоставление можно найти.

SQLNet: Генерация структурированных запросов по вопросам на естественном языке без усиленного обучения. [7]

Фактически стандартный подход для решения проблемы семантического разбора заключается в том, чтобы рассматривать как описание естественного языка, так и SQL-запрос в качестве последовательностей и обучать модель S2S или ее варианты, которые могут использоваться как синтаксический анализатор. Одна из таких проблем заключается в том, что разные SQL-запросы могут быть эквивалентны друг другу из-за коммутативности и ассоциативности.

SQLizer: Синхронизация запросов с естественного языка [9]

Существующие методы автоматического синтеза SQL-запросов подразделяются на два разных класса: подходы, основанные на шаблонах и те, которые основаны на естественном языке.

Методы программирования по шаблону требуют от пользователя представления миниатюрной версии базы данных вместе с ожидаемым выходом. Недостатком методов, ориентированных на шаблоны, является то, что они требуют, чтобы пользователь был знаком со схемой базы данных. Более того, поскольку реальные базы данных обычно включают в себя множество таблиц, пользователю может быть довольно сложно выразить свое намерение с использованием примеров ввода-вывода.

Генерация SQL-запросов из естественного языка [10]

Подход авторов основывается на глубокой нейронной сети, которая переводит вопросы на естественном языке в SQL-запросы. SQL-запрос может быть разбит на 3 части: SELECT, operator и WHERE. Идея заключается в том, чтобы избежать S2S подхода, где порядок не имеет значения.

Независимая архитектура схемы БД для перевода из NL в SQL [11]

На протяжении многих лет программисты пытаются преодолеть языковой барьер между пользователем и компьютером. В большинстве мест компьютер используется для сортировки данных и поиска (в соответствии с потребностями и требованиями пользователя). В течение последних десятилетий в преобразовании NL в SQL существует много работ. Процесс преобразования NL в SQL разделен на пошаговые уровни. Например, морфология имеет дело с наименьшей частью слова. Лексический уровень имеет дело с структурой предложения и символикой. Здесь также рассмотрены возможные значения и выбирается то, которое подходит.

Генерация SQL-запросов из естественного языка [10]

Подход авторов основывается на глубокой нейронной сети, которая переводит вопросы на естественном языке в SQL-запросы. SQL-запрос может быть разбит на 3 части: SELECT, operator и WHERE. Идея заключается в том, чтобы избежать S2S подхода, где порядок не имеет значения.

Выбор базы данных

На рисунке представлена база данных достопримечательностей Санкт-Петербурга, которая составлена вручную. Сейчас содержится около 20 записей. Скрипты для её создания находятся в открытом доступе по ссылке:
https://github.com/shparutask/TranslationSystem/tree/master/Scripts/SPB

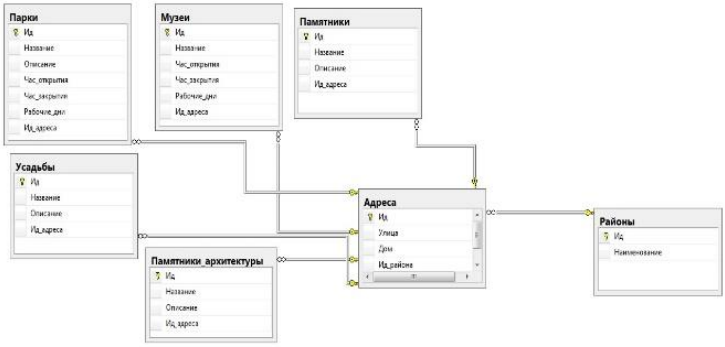

Рисунок 1: Схема для базы данных достопримечательностей Санкт-Петербурга

Результат базовой реализации

В результате работы реализовано приложение, в котором в поле ввода вводится вопрос, затем, по нажатии кнопки “Выполнить”, выводится результат запроса. Также есть возможность увидеть сам запрос, нажав “Показать запрос” (см. Рисунок 2).

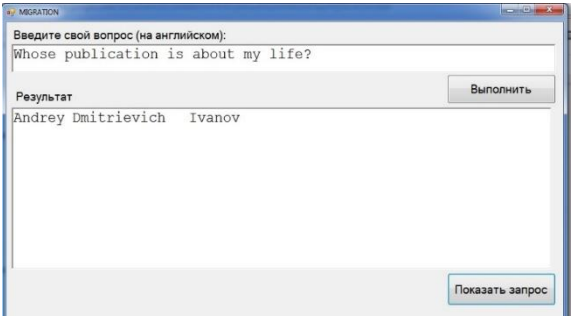

Рисунок 2: Интерфейс программы

Проект можно скачать по ссылке https://github.com/shparutask/TranslationSystem.

Методология работы

По итогам обзора было решено, что решение, используемое в работе **[3]** наиболее подходящее, и, согласно этому подходу, трансляция вопроса на естественном английском языке в SQL-запрос будет производиться в 4 этапа, аналогичным первым четырем этапам в работе **[3]**. Также использовались материалы работ **[12]** и **[9]** для попытки усовершенствовать систему, и более глубокого понимания материала.

До появления вопроса на естественном языке проводится препроцессинг. После его завершения, выполняется тегирование, парсинг, абстрактная семантическая интерпретация, конкретная семантическая интерпретация (**[3]**).

Описанная ранее реализация позволяет перевести именно с английского языка на язык SQL. В данный момент стоит задача перевода с русского языка. Для этого к существующей реализации был добавлен класс YandexTranslator, который содержит в себе API компании Yandex для перевода с русского языка на английский, а также класс Tesaurus, который

представляет собой пополняемое хранилище – аналог русско-английского словаря.

Заключение

Итак, на данный момент есть приложение для Windows, настроенное на конкретную базу, которое выдает ответы на вопрос пользователя на русском и английском языках. В дальнейшем планируется улучшить грамматику, расширить базу вопросов, архитектуру приложения, а также улучшить графический интерфейс.

Литература

1. Alexander Ran, Raimondas Lencevicius. Natural Language Query System for RDF Repositories // Proceedings of the 7-th International Symposium on Natural Language Processing, SLNP. –2007. –6.
2. Alessandra Giordani and Alessandro Moschitti. Generating SQL Queries Using Natural Language Syntactic Dependencies and Metadata // Department of Computer Science and Engineering University of Trento. – 2012. – 6.
3. Florin Brad, Radu Iacob, Ionel Hosu, and Traian Rebedea. Dataset for a Neural Natural Language Interface for Databases (NNLIDB) // Proceedings of the 8-th International Joint Conference on Natural Language Processing. –2017. – c.906-914.
4. Fred Popowich, Milan Mosny, David Lindberg. Interactive Natural Language Query Construction for Report Generation // Proceedings of the 7-th International Natural Language Generation Conference. –2012. – c.115-119.
5. Ikshu Blalla, Archit Gupta. Generating SQL queries from natural language // Department of Computer Science of Stanford University. –2017. – 9.
6. Navid Yaghmazadeh, Yuepeng Wang, Isil Dillig, and Thomas Dillig. SQLizer: Query Synthesis from Natural Language // ACM Program. Lang. 1, 1, Article 1 (January 2017). –2017. – 25.
7. Nicolas Kuchmann-Beauger. Question Answering System in a Business Intelligence Context // HAL archives-ouvertes.fr. –2017. – c.15-137.
8. Saima Noreen Khosa, Muhammad Rizwan. Database schema independent architecture for NL to SQL query conversion // Khwaja Fareed University of Engineering and IT. –2014. – c. 95-99.
9. Shadi Abdul Khalek, Sarfraz Khurshid. Automated SQL Query Generation for Systematic Testing of Database Engines // The University of Texas at Austin. –2010. – c.2-5.
10. Shay Cohen, Toms Bergmanis. A Natural Language Query System in Python/NLTK. – https://github.com/andrrra/Natural-Language-Query-

System.
11. Xiaojun Xu, Chang Liu, Dawn Song. SQLNet: Generating structured queries from Natural Language without reinforcement learning // ICLR-2018. –2017. – 15.
12. Посевкин Р. В. Модели, методы и программные средства построения естественно-языкового пользовательского интерфейса к базам данных // СПб-2018. -138