

В. М. Борисюк¹
А. В. Козловський¹

ІНТЕЛЕКТУАЛЬНА ТЕХНОЛОГІЯ ПЕРЕТВОРЕННЯ ПРИРОДНОЇ МОВИ В SQL-ЗАПИТИ

¹Вінницький національний технічний університет

Дані стали невід'ємною складовою сучасного світу, відіграючи критичну роль в глобальній економіці та вирішенні соціальних проблем. Вони дозволяють здійснювати глибокий аналіз та враховувати кількісну інформацію в процесі прийняття обґрунтованих рішень. Однак, ефективне маніпулювання даними вимагає від користувачів глибоких знань Structured Query Language (SQL), що може стати великою перешкодою для різних груп користувачів, включно з власниками малих підприємств та працівниками великих промислових компаній. Зазвичай, введення даних контролюється людьми, що може призвести до людських помилок і значних часових витрат під час формування складних SQL-запитів.

Актуальність впровадження методів, які полегшують цей процес, є високою, і вибрана для розгляду у статті технологія генерації SQL-запитів з введення природною мовою, становить значний інтерес. Цей метод автоматизації може радикально підвищити продуктивність, зменшуючи помилки та складність, що часто асоціюється з SQL-запитами, і дозволяючи користувачам концентруватися на внесенні ідей, які можуть трансформувати реальність. Ця інноваційна модель базується на передових технологіях обробки природної мови (Natural Language Processing, NLP) і глибокого навчання. Використання мереж Long Short Term Memory (LSTM) дозволяє системі ефективно розуміти природну мову та передбачати відповідні SQL-запити. Результатом є сформований на основі запиту користувача SQL-запит, представлений у зрозумілому для користувача форматі.

Впровадження такої системи не тільки спрощує вивчення SQL для нових користувачів, але й збільшує ефективність для тих, хто вже знайомий з SQL, дозволяючи їм працювати продуктивніше.

Ключові слова: штучний інтелект, нейронні мережі, машинне навчання, LSTM, обробка природної мови, SQL, СУБД.

Вступ

У епоху технологій багато робіт стали набагато простішими та ефективнішими. Технології, такі як штучний інтелект (AI), машинне навчання (ML), глибинне навчання (DL), допомагають людям у багатьох галузях, таких як медицина, дослідження космосу, біотехнології, автомобільна промисловість та інших. Технологія видаляє бар'єри спеціалізованих знань для виконання певних робіт: будь-хто може створювати веб-сторінки, використовуючи готові шаблони та модифікувати їх за допомогою методології перетягування та відпускання; будь-хто може створювати 3D-моделі, використовуючи 3D-принтери. Отримання необхідних даних чи інформації з реляційних баз даних — це скрупульозний процес. Людина повинна мати спеціалізовані знання з управління базами даних, повинна вивчити синтаксис різних складних запитів. Це ускладнює отримання даних з баз даних для нефахівця. Розвинута технологія може допомогти вирішити цю проблему, де користувачам не потрібно запам'ятовувати фрагменти коду та спеціалізуватися у базах даних. Для взаємодії з комп'ютерами можна використовувати обробку природної мови. Нейронні мережі можна використовувати для передбачення семантики вимог користувача та передачі їх комп'ютерам.

Метою роботи є поєднати ці технології та створити середовище, де будь-хто може створити SQL-запит та працювати з реляційними базами даних. Користувач може взаємодіяти з комп'ютером, вказуючи свої потреби у даних з баз даних природною мовою, вимовляючи їх програмі, яка обробляє текст для розуміння потреб користувача та створення структурованого запиту, який можна використовувати для отримання даних з бази даних. Нейронна мережева модель використовується, оскільки вона дає найточніші результати порівняно з іншими технологіями. Запропонована модель приймає мовлення англійською мовою як вхідні дані.

Ідея розв'язання задачі

Для розв'язання поставленої задачі застосовано класичний узагальнений алгоритм машинного навчання [1]:

1. Побудова і розмітка навчальної вибірки даних.
2. Обробка даних за використання методів NLP:
 - а) токенизація (tokenization) — передбачає сегментування тексту на менші одиниці, які аналізуються окремо;
 - б) лематизація (lemmatisation) — це процес групування відмінюваних форм слова так, щоб їх можна було проаналізувати як єдиний елемент, ідентифікований за лемою слова або словниковою формою;
 - с) видалення стоп слів.
3. Побудова моделі на основі пошуку LSTM (Long short-term memory) [2].
4. Навчання моделі на навчальних даних.
5. Аналіз ефективності.

Побудова навчального набору даних

Створено оригінальний набір даних для навчання моделі. Вибраним форматом для набору даних є формат JSON (JavaScript Object Notation). Цей формат ефективний у передаванні, обробці та аналізі даних. Це формат пар атрибут-значення. Використано 3 пари атрибут-значення для однієї функції SQL. Атрибути — це тег, шаблон, відповідь. Структура набору даних для вибору певної таблиці з бази даних подана на рис. 1.

```
{
  "dataset": [
    {
      "tags": "select_table",
      "patterns": [
        "give me the table table_name",
        "What is the query to print the table table_name",
        "show me the table table_name"
      ],
      "reponse": [
        "select * from table_name"
      ]
    }
  ]
}
```

Рис. 1. Структура набору даних в форматі JSON

У атрибуті *tags* вказано назву функції SQL. У атрибуті *patterns* вказано можливі способи, якими користувач може сформулювати вимогу до певного запиту для його активації, і в атрибуті *responses* ми маємо скелетну структуру запиту, яку можна завершити, обробивши інформацію з бази даних. Набір даних повинен включати всі методи та функції, що використовуються в SQL. У наборі даних користувач повинен надати можливі команди, які він може використати, щоб отримати потрібний запит. Кількість наданих команд буде змінюватися відповідно до функції SQL. Таким чином, чим більше користу-

вач може надати команд для певного запиту тим більше буде максимальна точність, проте це збільшить час тренування моделі. Запит користувача також буде оброблений для того, щоб привести його до формату, показаного в атрибуті *patterns*.

Передобробка навчального набору даних

Оскільки ймовірність того, що користувач може надати зайві дані разом зі своїми вимогами, що може вплинути на продуктивність моделі, стає необхідним попереднє оброблення даних перед їхньою передачею до моделі. Таким чином, існують різні кроки, які потрібно виконати для перетворення даних у необхідний формат. Кроки показано на рис. 2.

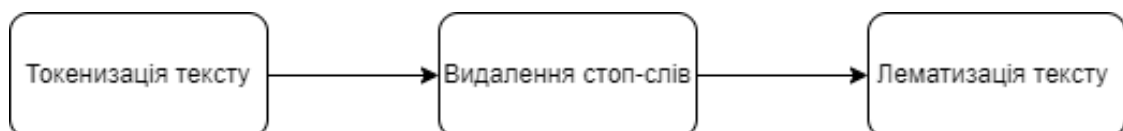


Рис. 2. Кроки передобробки набору даних

Токенизація означає розбиття речень на їхні складові слова [3]. Це можна зробити так, що кожний раз, коли система зустрічає пробіл, вона вважає його за токен. Приклад токенизації подано в табл. 1.

Таблиця 1

Приклад токенизації

Вхідні дані	Результат токенизації
"Show me the Student table"	["Show", "me", "the", "Student", "table"].

Видалення зайвих слів [4], [5]. Існують різні слова, які часто використовуються, але не мають значення під час їхнього аналізу. Ці слова називаються зайвими словами (stopwords). Тому їх потрібно видалити з речення, щоб покращити продуктивність моделі. Це робиться за допомогою створення списку всіх зайвих слів. Потім перевіряється, чи знаходиться слово з речення в цьому списку зайвих слів, якщо так, то воно видаляється. Приклад видалення зайвих слів відображено в табл. 2.

Таблиця 2

Приклад видалення зайвих слів

Вхідні дані	Результат видалення зайвих слів
["Show", "me", "the", "Student", "table"]	["Show", "Student", "table"]

Як видно з таблиці 2, слова "me" та "the" видалені, як неважливі для побудови моделі перетворення природної мови в SQL запит.

Лематизація займається пошуком кореневих слів [6]. Різні слова можуть бути записані у різних формах, враховуючи час, кількість і т. д. Це може понизити точність моделі, оскільки вона буде інтерпретувати слова як зовсім різні. Приклад лематизації відображено в табл. 3.

Таблиця 3

Приклад лематизації

Вхідні дані	Результат лематизації
Two cars collided	Two car collide

Модель буде навчатися на вхідних та вихідних даних. Вхідні дані будуть складатися зі зразків — різних способів, якими користувач може висловити свою вимогу для активації певної функції. Вихідні дані будуть тегом або класом, до якого належить цей вхідний зразок.

Тренування моделі

Вхідні та вихідні дані будуть навчатися за допомогою LSTM. LSTM є найвикористовуванішим алгоритмом у багатьох галузях для вирішення різних проблем, таких як прогнозування патернів на біржі за допомогою вхідних даних з продажу, машинний переклад мови та прогнозування подальших слів у багатьох програмах обміну повідомленнями. У запропонованій моделі використовуватиметься модель LSTM для отримання інтену вимог користувача. Аналізуючи послідовність слів, які користувач використав у запиті, модель поверне тег, який вказує на функцію або запит, який користувач хоче виконати. Після отримання тегу модель отримає скелетну структуру запиту, яка зберігається в наборі даних. Скелетний запит буде оброблений за допомогою інформації, яку користувач надав про набір даних у додатку. Потім модель поверне користувачеві дійсний SQL запит, як показано на рис. 3.

Використання метаданих з баз даних може значно підвищити точність моделі, яка перетворює природну мову на SQL запити.

Контекстне розуміння: метадані, такі як назви таблиць, типи стовпців і зв'язки між таблицями, надають моделі контекст про схему бази даних. Цей контекст є важливим для генерації моделлю семантично коректних SQL запитів, які відповідають правильній структурі бази даних.

Розв'язання неоднозначностей: природна мова часто може бути неоднозначною. Метадані допомагають моделі розв'язати терміни у введенні користувача. Наприклад, якщо користувач використовує термін «продажі» у своєму запиті, метадані можуть допомогти визначити, чи це стосується таблиці «продажі» або стовпця «продажі» в таблиці.

Передбачення з'єднань: розуміння зав'язків між таблицями є важливим для генерації складних запитів, особливо тих, що вимагають з'єднань. Метадані про зовнішні та первинні ключі можуть допомогти моделі передбачити, коли і як з'єднувати таблиці для отримання правильних даних.



Рис. 3. Схема використання LSTM моделі для перетворення природної мови в SQL запит

Висновок про типи: знаючи типи даних стовпців (такі як ціле число, текст, дата), модель може відповідно формувати SQL запит. Наприклад, якщо користувач хоче фільтрувати результати на основі дати, модель може використовувати метадані для того, щоб правильно сформувати дату в запиті.

Оптимізація запитів: метадані можуть допомогти оптимізувати запити, дозволяючи моделі розуміти, які поля індексовані, що може вплинути на структуру запиту з метою оптимізації продуктивності.

Автозаповнення та перевірка на помилки: під час генерації запиту метадані можуть бути використані для автозаповнення назв таблиць або стовпців, а також для ранньої перевірки на помилки, щоб переконатися, що запит відноситься до наявних елементів бази даних.

Аналіз результатів

Набір даних навчено за допомогою моделі на основі нейронних мереж LSTM, яка забезпечує достатню точність. Модель показала точність на рівні 93,47 %. Для перевірки точності модель запущено для кількох тестових випадків, і відповідно розраховано точність. Також помічено, що модель потребує часу для повернення запиту у порядку мілісекунд, що свідчить про швидкість прийняття рішень моделлю. Помічено, що чим простіше виражені вимоги, тим точніший результат буде повернутий. Таким чином, результат також залежить від користувача, наскільки ефективно та просто користувач може вказати свої вимоги. Загалом, розробка системи є ідеальною для вилучення даних з бази даних. Результат роботи моделі відображено в табл. 4.

Таблиця 4

Приклад роботи моделі

Запит користувача	SQL запит
Show the entire student table	select * from student
Show the name column from student table	select name from student
Give me grade column from student table	select grade from student
Delete the student table	drop table student
Alter the table student and drop email	alter table student drop column email

Висновки

Розглянуто актуальну задачу перетворення запитів, сформульованих природною мовою, у SQL запити за допомогою нейромережових моделей. Досліджено і формалізовано методи обробки природної мови і машинного навчання, зокрема, використання LSTM мереж, для розробки моделі, що дозволяє автоматизувати процес формування SQL запитів. Визначено найефективніші технологічні рішення для розв'язання поставленої задачі. Обґрунтовано, що використання комбінації різних операцій обробки даних, валідаційних наборів даних, а також методів ансамблювання моделей може суттєво підвищити точність роботи системи.

Проведено розвідувальний аналіз і розроблено алгоритм, який підвищує точність перетворення текстових даних у структуровані SQL запити. Аналіз ефективності алгоритму показав, що зазначений підхід забезпечує високу точність розпізнавання і відтворення запитів. Експерименти з різними наборами даних демонструють значні переваги запропонованого методу, особливо у складних запитах до баз даних. Основною перевагою цього підходу є використання метаданих з бази даних, що підвищує точність побудови SQL запиту.

Запропонована система відкриває нові можливості для користувачів без спеціалізованих знань SQL, дозволяючи їм ефективно здійснювати запити до баз даних. Результати дослідження підтверджують, що сучасний стан науки і техніки дозволяє створювати інтуїтивно зрозумілі інтерфейси для роботи з великими обсягами даних, що може бути адаптовано для використання в різних галузях і профілях діяльності.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- [1] Chollet Francois, *Deep Learning with Python*, 2018, pp 83-90.
- [2] S. Hochreiter, and J. Schmidhuber, *Long short-term memory*, 1997, pp. 1735-1780.
- [3] *Anatomy of a Compiler*, 2024. [Electronic resource]. Available: <https://www.cs.man.ac.uk/~pjj/farrell/comp3.html>.
- [4] A. Rajaraman, and J. D. Ullman, *Data Mining of Massive Datasets*, 2011, pp. 1-17.
- [5] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008, p. 27.
- [6] Thomas Müller, Ryan Cotterell, Alexander Fraser, and Hinrich Schütze, "Joint Lemmatization and Morphological Tagging with LEMMING", in *Conference on Empirical Methods in Natural Language Processing*, Lisbon: Association for Computational Linguistics, 2015, pp. 2268-2274.

Рекомендована кафедрою комп'ютерних наук ВНТУ

Стаття надійшла до редакції 24.02.2025

Борисюк Володимир Миколайович — аспірант кафедри комп'ютерних наук; e-mail: volodymyr.borysiuk0@gmail.com ;

Козловський Андрій Володимирович — канд. техн. наук, доцент кафедри комп'ютерних наук; e-mail: akozlovskiy@vntu.edu.ua.

Вінницький національний технічний університет, Вінниця

V. M. Borysiuk¹
A. V. Kozlovskiy¹

Intelligent Technology for Converting Natural Language into SQL Queries

¹Vinnitsia National Technical University

Data has become an integral component of the modern world, playing a critical role in the global economy and the resolution of social issues. They enable deep analysis and the consideration of quantitative information in decision-making. However, effective data manipulation requires users to have deep knowledge of Structured Query Language (SQL), which can be a significant barrier for various user groups, including small entrepreneurs and large industrial companies. Typically, data entry is controlled by humans, which can lead to human errors and significant time expenditures during the formation of complex SQL queries. The relevance of implementing methods that facilitate this process is high, and the technology for generating SQL queries from natural language input, discussed in this article, represents significant interest. This automation method can radically enhance productivity by reducing errors and complexities often associated with SQL queries, allowing users to focus on contributing ideas that can transform reality. This innovative model is based on advanced technologies for processing natural language (Natural Language Processing, NLP) and deep learning. The use of Long Short Term Memory (LSTM) networks enables the system to effectively understand natural language and predict the appropriate SQL queries. The result is processed by the system, and the final query is displayed to the user in an understandable format. Implementing such a system not only simplifies the learning of SQL for new users but also increases efficiency for those already familiar with SQL, allowing them to work more productively.

Keywords: AI, neural networks, machine learning, LSTM, natural language processing, SQL, DBMS.

Borysiuk Volodymyr M. — Post-Graduate Student of the Chair of Computer Sciences, e-mail: volodymyr.borysiuk0@gmail.com ;

Kozlovskiy Andrii V. — Cand. Sc., Associate Professor of the Chair of Computer Sciences, e-mail: akozlovskiy@vntu.edu.ua