

УДК 004.8:004.65

ВИКОРИСТАННЯ ГЕНЕРАТИВНОГО ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АВТОМАТИЧНОГО СТВОРЕННЯ СТРУКТУР БАЗ ДАНИХ

Запорожцев Д.С.

*Львівський національний університет ветеринарної медицини та
біотехнологій ім. С.З. Гжицького, Львів*

Розробка схеми бази даних (БД) є критичним, але часто трудомістким етапом у життєвому циклі програмного забезпечення.

Це дослідження націлене на потенціал використання Генеративного Штучного Інтелекту (ГШІ), зокрема Великих Мовних Моделей (LLMs), для автоматизації процесу створення та оптимізації структур реляційних баз даних.

Пропонується концептуальний підхід, заснований на перетворенні природної мови або бізнес-вимог у структурований SQL-код визначення даних (DDL), з фокусом на системі MySQL.

Для цього потрібно проаналізувати переваги, технічні виклики та напрямки подальших досліджень у цій галузі.

Традиційне проектування схем БД вимагає глибокого знання моделювання даних, нормалізації, особливостей конкретної СКБД (наприклад, MySQL) та доменних вимог.

Зростаюча складність сучасних застосунків і потреба у швидкій розробці вимагають нових, більш автоматизованих інструментів.

Генеративний ШІ зараз пропонує революційну можливість, використовуючи свою здатність аналізувати величезні обсяги інформації та генерувати структурований код на основі контексту та обмежень, заданих природною мовою.

Запропонована система на основі ГШІ для створення схем БД у MySQL складається з трьох основних етапів.

Етап перший – ввід даних, це може бути опис бізнес-логіки природною мовою (наприклад: «Створити базу даних для інтернет-магазину. Потрібні таблиці для товарів, замовлень, клієнтів та категорій. Товар належить до однієї категорії. Замовлення містить багато товарів.») або ж невеликі набори неструктурованих даних, які модель може аналізувати для виявлення закономірностей та зв'язків.

На другому етапі LLM виконує ключові завдання, а саме: ідентифікація таблиць та зв'язків (модель виділяє ключові сутності та визначає їхні зв'язки), визначення атрибутів та типів даних (модель обирає оптимальні типи даних MySQL (наприклад, INT, VARCHAR(255), DATETIME, DECIMAL) та встановлює обмеження), застосування принципів нормалізації (ГШІ прагне генерувати схему, що відповідає принципам нормалізації, зменшуючи надлишковість даних), генерація SQL DDL для MySQL (результатом є послідовність команд CREATE TABLE та ALTER TABLE з урахуванням специфічного синтаксису MySQL, включно з визначенням первинних та зовнішніх ключів, а також індексів).

Третій етап складається з синтаксичної перевірки, де перевіряється коректність SQL-коду MySQL, семантична перевірки, де оцінюється відповідність схеми початковим бізнес-вимогам та логічній цілісності (наприклад, чи коректно реалізовані всі необхідні зв'язки), та оптимізації, де ШІ може запропонувати або автоматично застосувати оптимізації, такі як вибір оптимальних індексів, механізмів зберігання (наприклад, InnoDB) або партиціонування для великих таблиць, ґрунтуючись на прогнозованих шаблонах запитів.

Разом з цим, при розробці БД за допомогою ГШІ слід взяти до уваги такі виклики, які представлені в табл.1.

Все це дає уявлення, що застосування генеративного ШІ для автоматичного створення структур баз даних, на прикладі MySQL і не тільки, має значний потенціал для прискорення розробки, демократизації доступу до проектування БД для неспеціалістів та зменшення кількості ручних помилок.

Таблиця 1 – Виклики при розробці БД за допомогою ШІ

Виклик	Опис та вплив на MySQL	Рішення на основі ГШІ
Неоднозначність вимог	Природна мова часто неоднозначна, що може призвести до неоптимального вибору сутностей або зв'язків.	Використання механізму повторного запиту (уточнення) або нульового навчання для виділення ключових термінів.
Оптимізація продуктивності	Генерація схеми без урахування шаблонів запитів може призвести до повільної роботи БД.	Інтеграція прогнозування запитів на основі опису використання системи для автоматичного додавання вторинних індексів.
Специфічні Типи mysql	Вибір найкращого типу даних.	Навчання моделі на векторній базі знань з документацією БД та найкращими практиками.
Складні зв'язки	Реалізація зв'язків "багато-до-багатьох" через проміжні таблиці.	Навчання на великому корпусі існуючих схем БД для правильного застосування патернів.

Таким чином, розробникам треба враховувати, що запропонований підхід не тільки може вирішувати проблеми формалізації, але має спроможність породжувати унікальні виклики та вразливості, які не властиві традиційному ручному проектуванню.

Література:

1. Victoria Lazarus, «How to Use AI for MySQL: Optimizing Queries and Database Management». [Он-лайн]. Доступно: <https://blog.devart.com/how-to-use-ai-for-mysql.html>
2. Бурлака Я., «Технології штучного інтелекту в наукометричних базах даних». [Он-лайн]. Доступно: <https://nim.media/articles/tekhnologiyi-shtuchnogo-intelektu-v-naukometrichnikh-bazakh-danikh>
3. Волнянский А., «Що таке генеративний штучний інтелект і де він застосовується?» [Он-лайн]. Доступно: <https://blog.colobridge.net/uk/2025/08/generative-artificial-intelligence-ua/>
- 4, Gopikrishna Maddali, «Enhancing Database Architectures with Artificial Intelligence (AI)», May 2025 International Journal of Scientific Research in Science and Technology 12(3):296-308 [Он-лайн]. Доступно: <https://doi.org/10.32628/IJSRST2512331>

5. Yugabyte Team, «Generative AI for Databases: Choosing the Right AI and Putting it to Work». [Он-лайн]. Доступно: <https://www.yugabyte.com/blog/generative-ai-for-databases/>
6. TiDB Team, «Understanding AI in Database Management». [Он-лайн]. Доступно: <https://www.pingcap.com/article/understanding-ai-in-database-management-transforming-dbms/>