TRZYDZIESTE DRUGIE FORUM TELEINFORMATYKI®

„Generowanie, optymalizacja i wizualizacja zapytań SQL na podstawie poleceń tekstowych z wykorzystaniem sztucznej inteligencji”

Magdalena Szpitalna

Prof. dr hab. Ewa Łaźniewska

Uniwersytet Ekonomiczny w Poznaniu

Problem badawczy i cel pracy

PROBLEM

- **Dostęp do danych wymaga znajomości SQL**
  bariera dla analityka nietechnicznego
- **Ograniczona racjonalność- H. Simon**
  decydent nie sięga po dane zbyt kosztowne do pozyskania

CEL

Aplikacja Text-to-SQL z optymalizacją i wizualizacją

HIPOTEZA

Aplikacja skraca czas pozyskiwania informacji przy zachowaniu poprawności danych i bezpieczeństwa

Istniejące rozwiązania i luka rynkowa

Narzędzie	Mocna strona	Ograniczenie
Power BI	Lider rynku BI	Model semantyczny + DAX, Copilot tylko w planach płatnych
Chat2DB	Najbliższy funkcjonalnie	Optymalizacja tylko w zamkniętym wariancie Pro
Wren AI	Open-source, warstwa semantyczna	Brak modułu optymalizacji zapytań

LUKA: żadne z analizowanych narzędzi nie łączy otwartego kodu z optymalizacją opartą na planie wykonania.

Tę lukę wypełnia zbudowany prototyp

Podstawy teoretyczne

Z pytania w języku naturalnym powstaje gotowe zapytanie SQL

ŚRODOWISKO **PostgreSQL 18** **Node.js** **Next.js** **Gemini 2.5**

BEZPIECZEŃSTWO PRACY Z LLM

- do modelu trafia tylko schemat bazy
- konto bazodanowe read-only
- walidacja składniowa i semantyczna
- kontrakt Zod na wyjściu modelu

OPTYMALIZACJA

- plan wykonania (EXPLAIN)
- indeksy
- optymalizacja zapytań

Metodyka badania

1. ŹRÓDŁA
baza PostgreSQL + schemat

→

2. GENEROWANIE
prompt + schemat → SQL, kontrakt Zod

→

3. OPTYMALIZACJA
EXPLAIN → sugestie → Apply index

→

4. WIZUALIZACJA
wynik + automatyczny wykres

Dwa scenariusze: analityk nietechniczny lub specjalista bazodanowy

OSIE PROJEKTU

- otwarta architektura
- specjalizacja w PostgreSQL
- generowanie i optymalizacja oparte na stanie bazy

ŚRODOWISKO TESTOWE

- ~8,7 mln wierszy - dane TPC-H
- 22 zapytania wzorcowe
- before / after - pomiar w aplikacji

Wyniki

Top-N (sortowanie)	239 ms → **1 ms**
Przepisanie zapytania	6,6 s → **3,5 s**

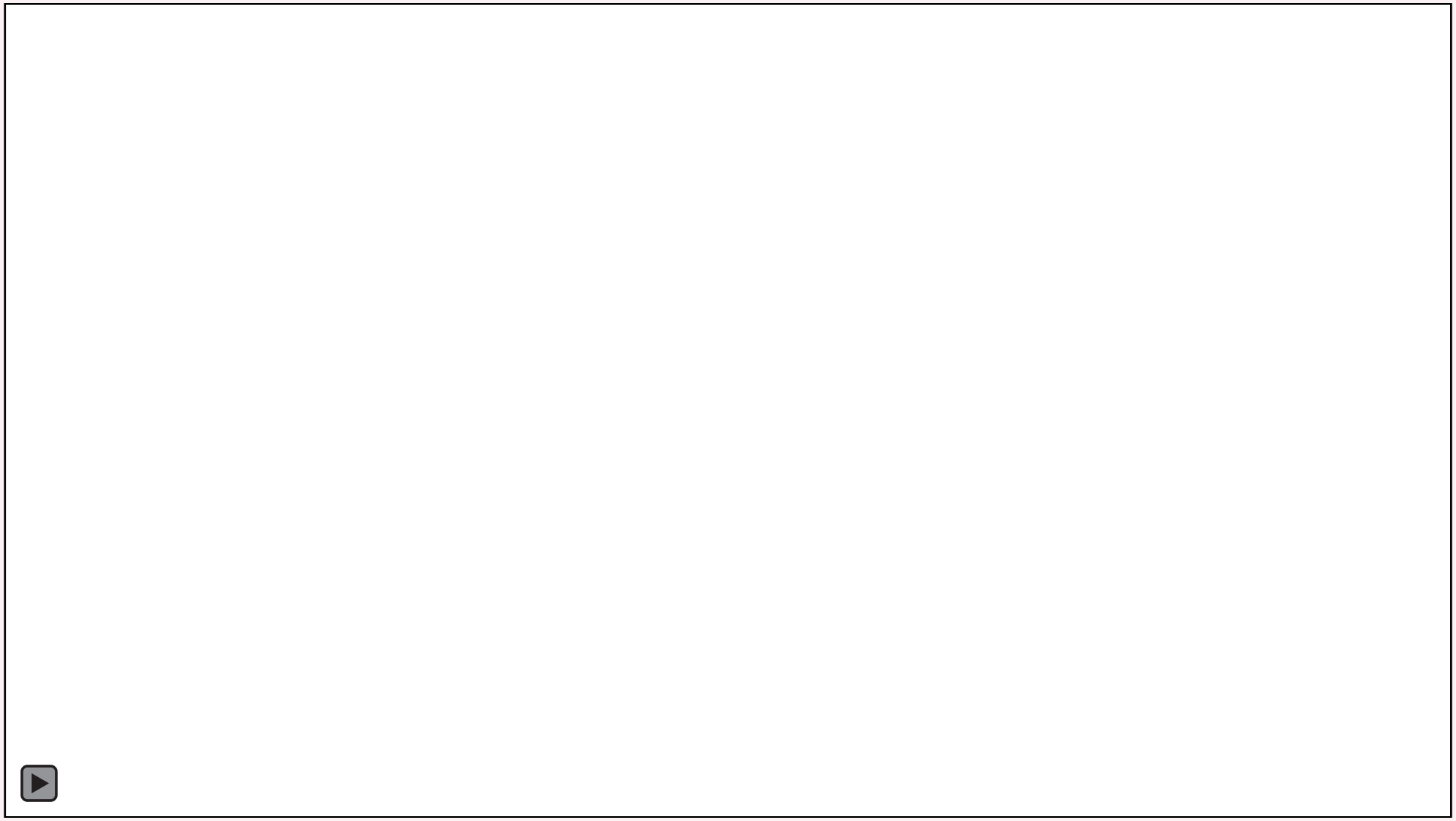
80% poprawności generowania w teście walidacyjnym

KONTRPRZYKŁAD

Indeks oceniony przez AI jako „wysoki wpływ” dał ~zero zysku

2,8 s → **2,8 s**

Demonstracja



Wnioski i podsumowanie

Krótszy czas otrzymania informacji - przy zachowanym bezpieczeństwie i poprawności (80%)

GŁÓWNY WKŁAD

Moduł optymalizacji oparty na planie wykonania z weryfikacją pomiaru

WARTOŚĆ EKONOMICZNA

Dane dostępne dla analityka bez znajomości SQL

KIERUNKI DALSZYCH PRAC

- HypoPG - wirtualne indeksy
- historia zapytań i pulpit
- większa skala danych
- audyt baz

Dziękuję za uwagę