Sztuczna Inteligencja

Wprowadzenie do modeli językowych (LLM) i Anthropic API

Anna Nowak 01.10.2026 613 wyświetleń 3 komentarzy

Dostępne w językach: PL EN

Co to jest LLM?

Large Language Model (LLM) to model uczenia maszynowego trenowany na ogromnych zbiorach tekstu. Potrafi:

- generować tekst, kod, tłumaczenia
- odpowiadać na pytania w języku naturalnym
- analizować, podsumowywać i klasyfikować dokumenty
- pisać i debugować kod

Najpopularniejsze modele

Model	Producent	Kontekst	Mocne strony
Claude Opus 4	Anthropic	200k tokenów	Rozumowanie, kod, bezpieczeństwo
Claude Sonnet 4	Anthropic	200k tokenów	Balans cena/jakość
GPT-4o	OpenAI	128k tokenów	Multimodalność
Gemini 1.5 Pro	Google	1M tokenów	Długi kontekst
Llama 3.1	Meta	128k tokenów	Open-source

Wywołanie Claude przez API

Python

```
import anthropic

client = anthropic.Anthropic(api_key="sk-ant-...")

message = client.messages.create(
    model="claude-opus-4-5",
    max_tokens=1024,
    system="Jesteś ekspertem PHP i Symfony.",
    messages=[
        {"role": "user", "content": "Wyjaśnij w 3 p
    ]
)

print(message.content[0].text)
```

PHP (Guzzle)

```php
use GuzzleHttp\Client;

$client = new Client(['base_uri' => 'https://api.a

$response = $client->post('/v1/messages', [
    'headers' => [
        'x-api-key'         => $_ENV['ANTHROPIC_AP
        'anthropic-version' => '2023-06-01',
        'content-type'      => 'application/json',
    ],
    'json' => [
        'model'      => 'claude-opus-4-5',
        'max_tokens' => 1024,
        'messages'   => [
            ['role' => 'user', 'content' => 'Napis
        ],
    ],
]);

$data = json_decode($response->getBody(), true);
echo $data['content'][0]['text'];
```

Streaming odpowiedzi

Zamiast czekać na całą odpowiedź, możesz strumieniować token po tokenie:

```python
with client.messages.stream(
    model="claude-opus-4-5",
    max_tokens=1024,
    messages=[{"role": "user", "content": "Napisz
) as stream:
    for text in stream.text_stream:
        print(text, end="", flush=True)
```

Typowe zastosowania w projektach webowych

1. **Generowanie opisów SEO** — wklejasz tytuł i treść, dostajesz meta description
2. **Asystent kodu** — code review, wyjaśnianie błędów
3. **Chatbot wsparcia** — odpowiada na pytania na podstawie FAQ
4. **Analiza sentymentu** — klasyfikacja komentarzy
5. **Tłumaczenia** — lepsze niż Google Translate dla tekstów technicznych

> ***Koszty:*** *Claude Sonnet to ~$3/1M tokenów wejściowych. Typowy artykuł blogowy to ok. 1500 tokenów. Rozsądne dla asystentów, drogie przy przetwarzaniu milionów dokumentów.*

Dobre praktyki

- zawsze waliduj i sanitizuj dane wejściowe od użytkownika
- nigdy nie wklejaj wrażliwych danych (hasła, klucze API) do promptu
- ustal `max_tokens` — bez limitu model może generować bardzo długie odpowiedzi
- używaj `system` promptu do nadania roli i ograniczeń modelowi

Tagi: #Python #AI #REST API

Udostępnij: Facebook X LinkedIn Kopiuj link

Komentarze (3)

J

Jan Kowalski 21.10.2026, 14:27

Przykład z Anthropic API działa jak marzenie. Polecam też tryb streaming!

P

Piotr Wiśniewski 21.10.2026, 14:27

Testuję Claude do generowania testów jednostkowych — wyniki rewelacyjne.

A

Anna Nowak 21.10.2026, 14:27

Ja używam go do code review. Oszczędza mnóstwo czasu w PR-ach.

Zaloguj się, aby dodać komentarz.

GitHub

© 2026 FactoryCode. Wszelkie prawa zastrzeżone.

Zbudowane z Symfony 8 & Claude AI