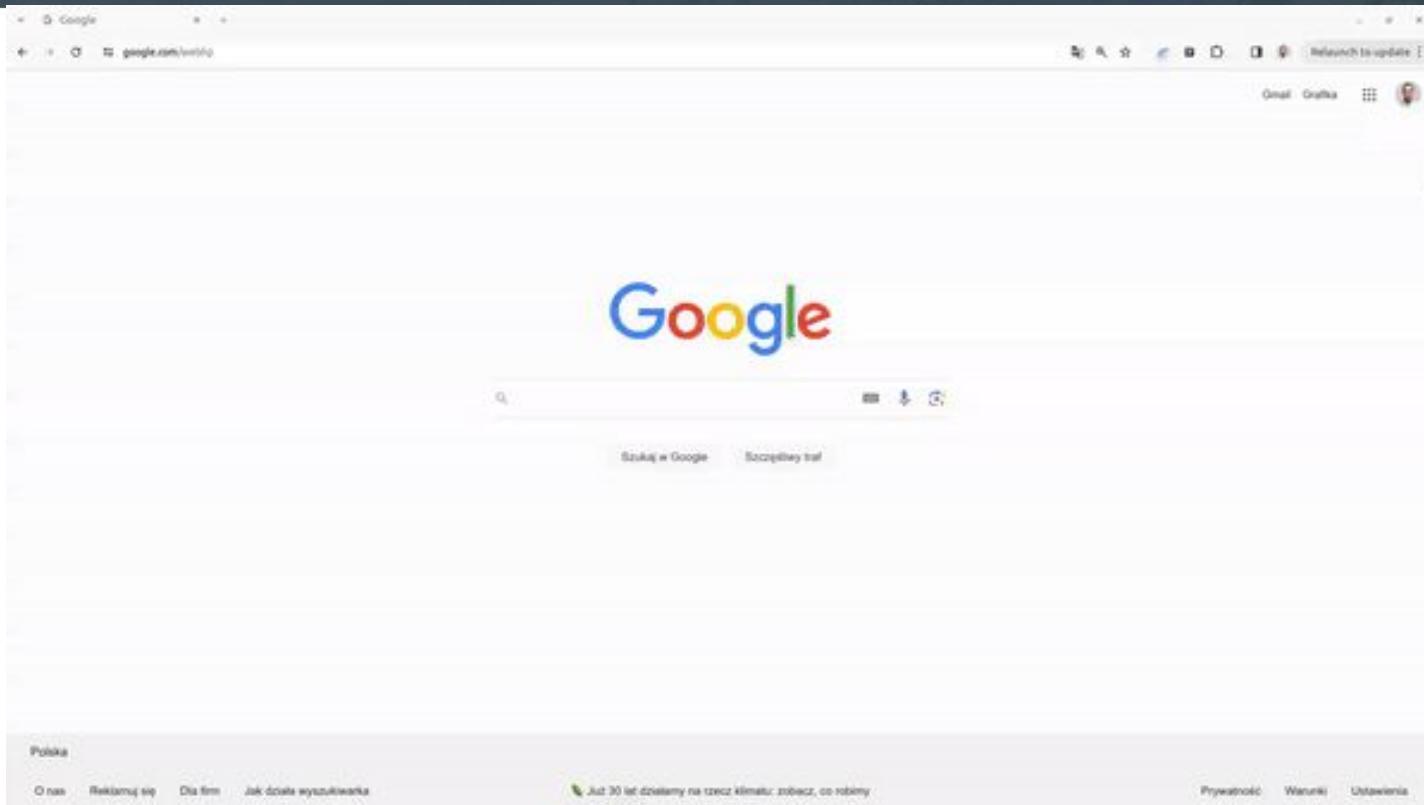


Generatywna sztuczna inteligencja z dużymi modelami tekstowymi

Wykorzystanie LLM i RAG - case study

Michał Żarnecki





29.01.2024



Michał Żarnecki

Audio engineer



Michał Żarnecki (ur. 12 listopada 1946 w Warszawie, zm. 21 listopada 2016, tamże) – **polski operator i reżyser dźwięku.**



Wikipedia

[https://pl.wikipedia.org › wiki › Michał_Żarnecki](https://pl.wikipedia.org/wiki/Michał_Żarnecki)

[Michał Żarnecki – Wikipedia, wolna encyklopedia](#)

Born: November 12, 1946, [Warsaw](#)

Died: November 21, 2016, [Warsaw](#)

Nominations: Polish Academy Award for Best Sound

Siblings: [Andrzej Żarnecki](#)

case study - błędne wyniki wyszukiwania w Google

Problem:

W wynikach wyszukiwania Google pod moim zdjęciem i nazwiskiem widnieje błędnie dopasowana data śmierci :O

**W jaki sposób doszło
do pomyłki algorytmu?**

**...może zdjęcia osób o tym samym imieniu
i nazwisku są podobne?**

Reference face



Comparison faces



▼ Results



▼ Request

```
{  
    "TargetImage": {  
        "Bytes": ()  
    },  
    "SourceImage": {  
        "Bytes": ()  
    },  
    "SimilarityThreshold": 0  
}
```

▼ Response

```
{  
    "SourceImageFace": {  
        "BoundingBox": {  
            "Width": 0.39943283796310425,  
            "Height": 0.5625719428062439,  
            "Left": 0.2943730950355553,  
            "Top": 0.15183748304843903  
        },  
        "Confidence": 99.9990463256836  
    }  
}
```

"FaceMatches": [
 {
 "Similarity": 0.06796722859144211,

Amazon Rekognition Face comparison



**...może zdjęcia osób o tym samym imieniu
i nazwisku są podobne?**

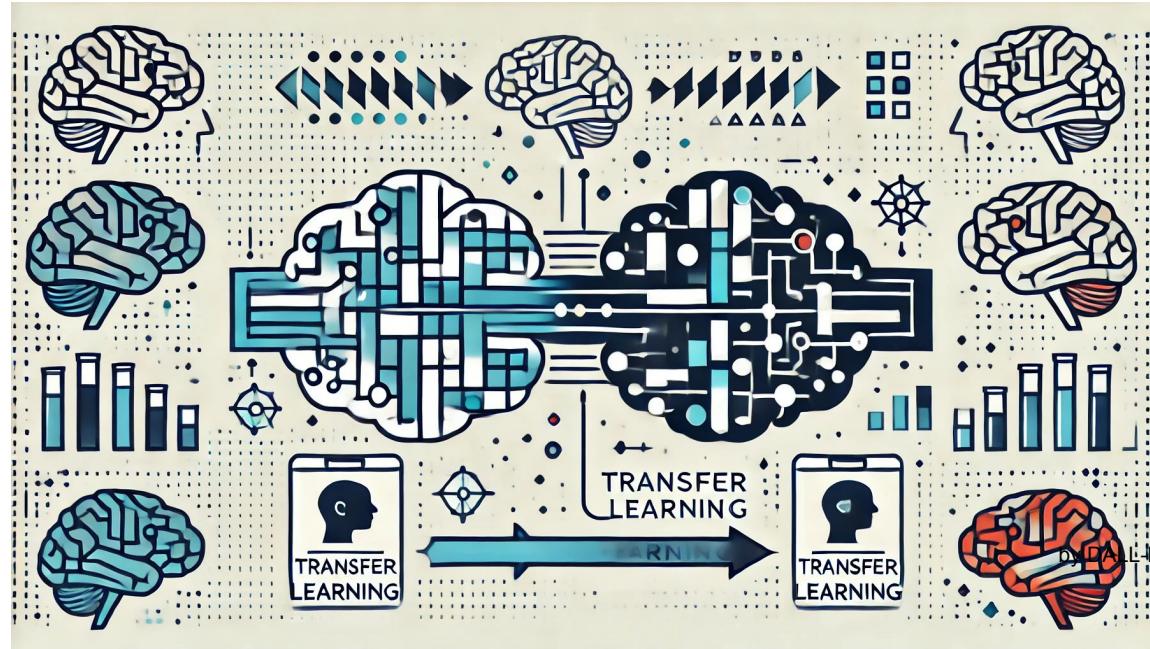
**Algorytm dopasował teksty zawierające
moje imię i nazwisko?**



**W jaki sposób przetwarzać teksty,
aby uniknąć pomyłek?**

RAG

Retrieval Augmented Generation



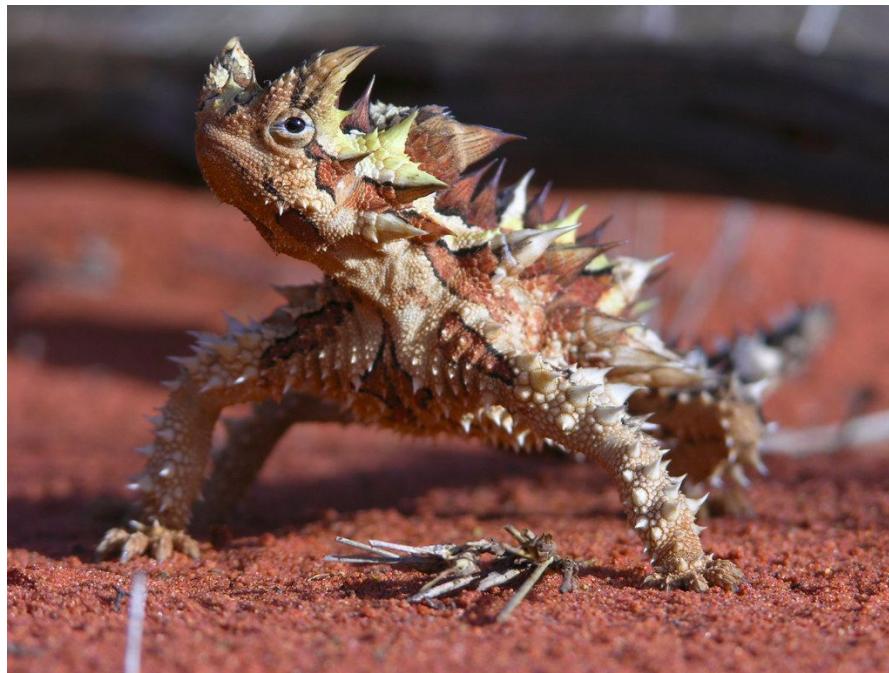
Transfer learning

Pytanie

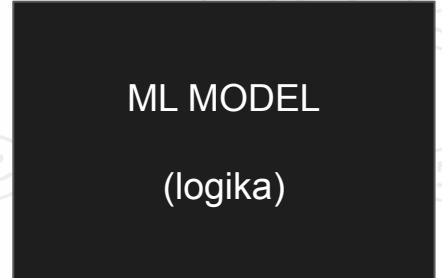
**Kto wie jak wygląda
Moloch straszliwy?**



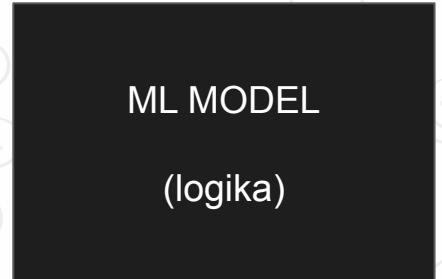




Czym jest uczenie maszynowe?



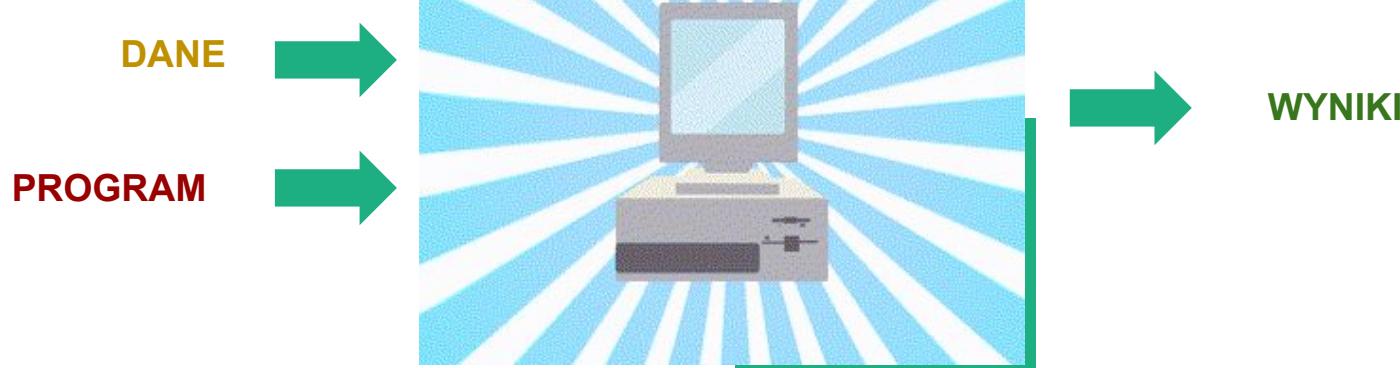
**moloch
straszliwy**



**nie moloch
straszliwy**

Czym się różni ML od "tradycyjnego" programowania?

Podejście tradycyjne



Czym się różni RAG/LLM od "tradycyjnego" programowania?

Wykorzystanie uczenia maszynowego

INSTRUKCJE
TEKSTOWE



PRZYKŁADY



PROGRAM



Czym się różni RAG/LLM od "tradycyjnego" programowania?

Wykorzystanie uczenia maszynowego

1 **INSTRUKCJE
TEKSTOWE**



PROGRAM

PRZYKŁADY



2

DANE
PROGRAM



WYNIKI

Aplikacja

Find answer in websites database

What is specialization of programmer and lecturer Michał Żarnecki based on his website content.

Generate answer

The screenshot shows a web application interface. At the top, there is a search bar with the placeholder "Przeszukaj Wikipedię" and a "Szukaj" button. To the left of the search bar is a sidebar with a menu icon, the word "WIKIPEDIA", and "Wolna encyklopedia". Below the search bar, the main content area displays a page for "Michał Żarnecki". The page includes a "Spis treści" (Table of Contents) section with links to "Artykuł", "Dyskusja", "Zyciorys", "Filmografia", "Nagrody i nominacje", and "Przypisy". On the right side of the page, there are buttons for "Czytaj", "Edytuj", "Edytuj kod źródłowy", and "Wyświetl". A sidebar on the far right lists "Michał Żarnecki" and "Michał Żarnecki Portfolio". The portfolio page itself has sections for "events", "lectures", and "projects". The bottom of the portfolio page features a footer with various technology-related terms like "Cassandra", "Docker", "Big Data", "langchain", "regular expression", etc.

Aplikacja na Github

PHP / Docker

<https://github.com/rzarno/php-rag>



Screenshot of the GitHub repository page for <https://github.com/rzarno/php-rag>.

The repository has 1 branch and 0 tags. The most recent commit was made by Michal Zarnecki 14 hours ago, with 23 commits in total.

Commit History:

File	Description	Time
.idea	fix paths	2 days ago
app	fix paths	2 days ago
documents	change website name	14 hours ago
script	vector postgres extension	3 days ago
.gitignore	fix dependencies	3 days ago
LICENSE.txt	implement RAG app with PHP	3 weeks ago
README.md	fix typo	yesterday
ai_chatbot_llm_rag.jpg	fix dependencies	3 days ago
docker-compose.yaml	get documents from vector db based on similarity	3 days ago
env	fix dependencies	3 days ago
index.php	fix dependencies	3 days ago

README | **GPL-3.0 license**

Creating RAG (Retrieval Augmented Generation) application in PHP

based on <https://github.com/Krisseck/php-rag>

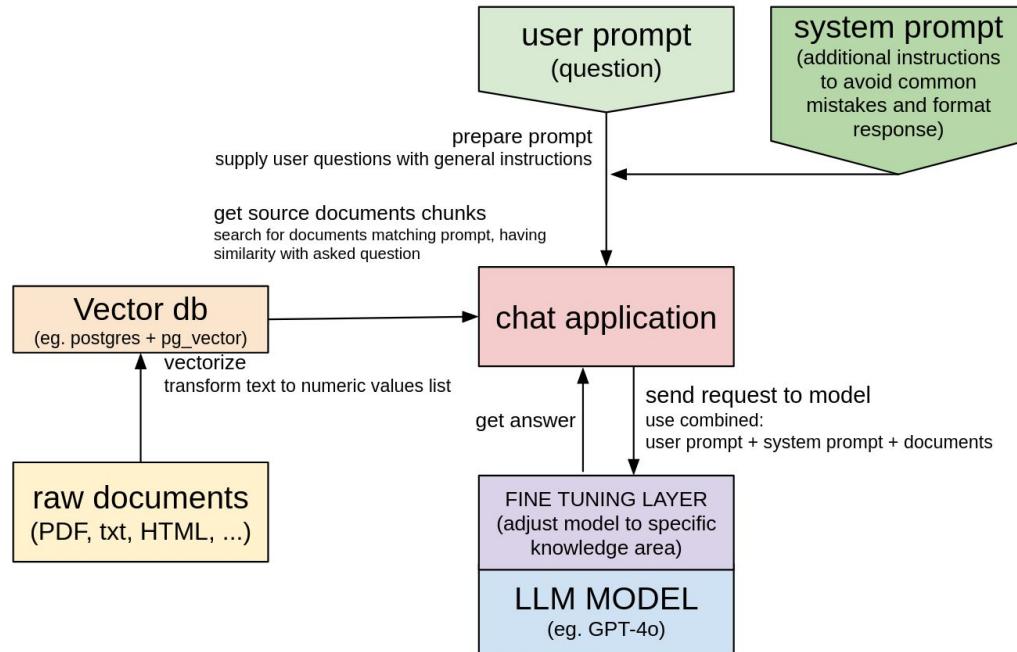
This application uses LLM (Large Language Model) GPT-4o accessed via OpenAI API in order to generate text based on the user input. The user input is used to retrieve relevant information from the database and then the retrieved information is used to generate the text. This approach combines power of transformers and access to source documents.

Setup:

1. Run in CLI: `cd app/src && composer install`
2. Create `api_key.txt` file inside `app/src` and put there your OpenAI API key
3. Run in CLI: `cd ../../ && composer install`

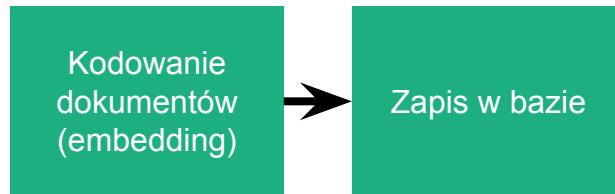
Schemat aplikacji

AI CHATBOT (LLM + RAG)

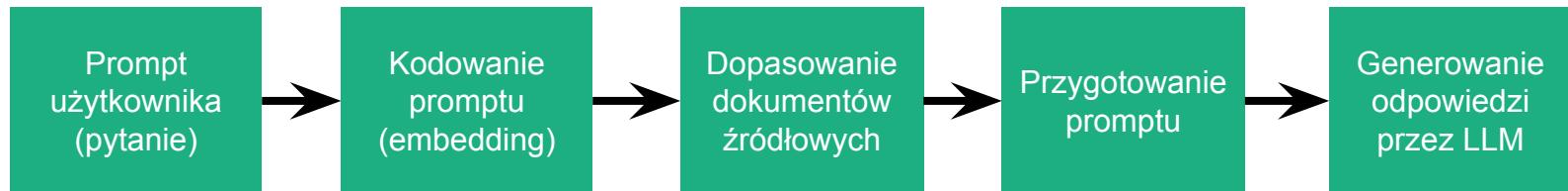


Proces przetwarzania zapytania

1. Import dokumentów do bazy wektorowej



2. Proces generowania odpowiedzi



Baza stron internetowych



WIKIPEDIA
Wolna encyklopedia

Przeszukaj Wikipedię Szukaj

Spis treści

[ukryj](#)

Początek

Zyciorys

Filmografia

Nagrody i nominacje

Przypisy

Michał Żarnecki [edytuj]

Artykuł

Dyskusja

Czytaj

Edytuj

Edytuj kod źródłowy

Wyświetl

Michał Żarnecki (ur. 12 listopada 1946 w Warszawie, zm. 21 listopada 2016, tamże^[1]) – polski operator i reżyser dźwięku.

Zyciorys [edytuj | edytuj kod]

Laureat Nagrody za dźwięk na Festiwalu Polskich Filmów Fabularnych w Gdyni oraz pięciokrotnie nominowany do Polskiej Nagrody Filmowej, Orzel w kategorii najlepszy dźwięk.

Michał

Data i miejsce urodzenia

Data i miejsce śmierci

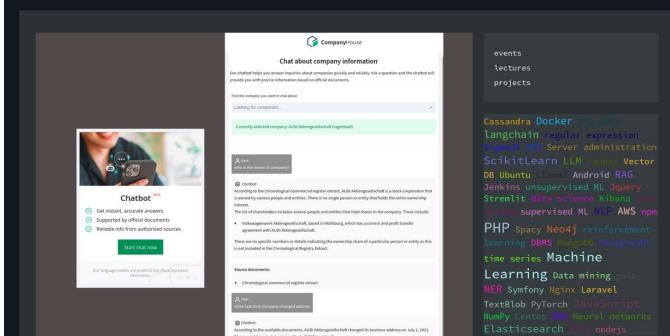
Zawód, zajęcie

Michał Żarnecki Portfolio

I'm a programmer and lecturer. My work is related to programming in Python/PHP/Javascript and designing systems and solutions related to AI/machine learning, data mining, big data and natural language processing.

LinkedIn

Github



HETUL MEHTA - UPDATED 3 YEARS AGO

84

New Notebook

Download (2 MB)



Website Classification

classify website URLs to different categories

Data Card Code (10) Discussion (2) Suggestions (0)

About Dataset

Context

This dataset was created by scraping different websites and then classifying them into different categories based on the extracted text.

Content

Below are the values each column has. The column names are pretty self-explanatory.

website_url: URL link of the website.

cleaned_website_text: the cleaned text content extracted from the

<https://www.kaggle.com/datasets/hetulmehta/website-classification>

Usability

10.00

License

CC0: Public Domain

Expected update frequency

Annually

Tags

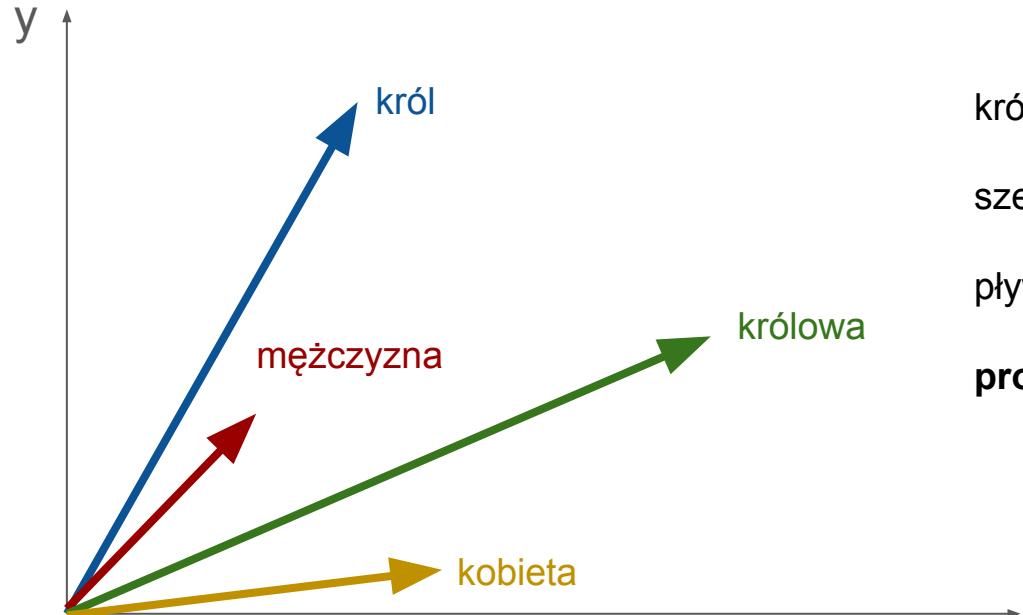
Kodowanie dokumentów (embedding)

Zapis w bazie

Baza wektorowa

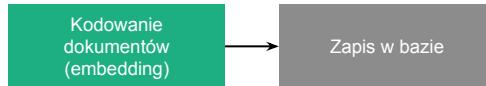
 PostgreSQL + pg_vector

Kodowanie tekstu w bazie do postaci wektorowej



król [2, 4]
królowa [5, 3]
...

- król - mężczyzna + kobieta \approx królowa
- szedłem - idę \approx biegłem - biegnę
- pływamy - my \approx pływam - ja
- projekt + deadline \approx stackoverflow + copy/paste**



Kodowanie tekstu w bazie do postaci wektorowej

```
1 model['go']
```

```
array([-0.078894,  0.4616 ,  0.57779 , -0.71637 , -0.13121 ,  0.4186 ,  
       -0.29156 ,  0.52006 ,  0.089986, -0.35062 ,  0.51755 ,  0.51998 ,  
       0.15218 ,  0.41485 , -0.12377 , -0.37222 ,  0.0273 ,  0.75673 ,  
      -0.8739 ,  0.58935 ,  0.46662 ,  0.62918 ,  0.092603, -0.012868,  
     -0.015169,  0.25567 , -0.43025 , -0.77668 ,  0.71449 , -0.3834 ,  
     -0.69638 ,  0.23522 ,  0.11396 ,  0.02778 ,  0.071357,  0.87409 ,  
     -0.1281 ,  0.063576,  0.067867, -0.50181 , -0.28523 , -0.072536,  
     -0.50738 , -0.6914 , -0.53579 , -0.11361 , -0.38234 , -0.12414 ,  
     0.011214, -1.1622 ,  0.037057, -0.18495 ,  0.01416 ,  0.87193 ,  
    -0.097309, -2.3565 , -0.14554 ,  0.28275 ,  2.0053 ,  0.23439 ,  
    -0.38298 ,  0.69539 , -0.44916 , -0.094157,  0.90527 ,  0.65764 ,  
     0.27628 ,  0.30688 , -0.57781 , -0.22987 , -0.083043, -0.57236 ,  
     -0.299 , -0.81112 ,  0.039752, -0.05681 , -0.48879 , -0.18091 ,  
    -0.28152 , -0.20559 ,  0.4932 , -0.033999, -0.53139 , -0.28297 ,  
    -1.4475 , -0.18685 ,  0.091177,  0.11454 , -0.28168 , -0.33565 ,  
    -0.31663 , -0.1089 ,  0.10111 , -0.23737 , -0.64955 , -0.268 ,  
     0.35096 ,  0.26352 ,  0.59397 ,  0.26741 ], dtype=float32)
```

Kodowanie
dokumentów
(embedding)

→ Zapis w bazie

Kodowanie tekstu w bazie do postaci wektorowej

```
1 model['away']
```

```
array([-0.10379 , -0.014792,  0.59933 , -0.51316 , -0.036463,  0.6588 ,  
       -0.57906 ,  0.17819 ,  0.23663 , -0.21384 ,  0.55339 ,  0.53597 ,  
       0.041444,  0.16095 ,  0.017093, -0.37242 ,  0.017974,  0.39268 ,  
      -0.23265 ,  0.1818 ,  0.66405 ,  0.98163 ,  0.42339 ,  0.030581,  
       0.35015 ,  0.25519 , -0.71182 , -0.42184 ,  0.13068 , -0.47452 ,  
      -0.08175 ,  0.1574 , -0.13262 ,  0.22679 , -0.16885 , -0.11122 ,  
      -0.32272 , -0.020978, -0.43345 ,  0.172 , -0.67366 , -0.79052 ,  
       0.10556 , -0.4219 , -0.12385 , -0.063486, -0.17843 ,  0.56359 ,  
       0.16986 , -0.17804 ,  0.13956 , -0.20169 ,  0.078985,  1.4497 ,  
       0.23556 , -2.6014 , -0.5286 , -0.11636 ,  1.7184 ,  0.33254 ,  
       0.12136 ,  1.1602 , -0.2914 ,  0.47125 ,  0.41869 ,  0.35271 ,  
       0.47869 , -0.042281, -0.18294 ,  0.1796 , -0.24431 , -0.34042 ,  
       0.20337 , -0.93676 ,  0.013077,  0.080339, -0.36604 , -0.44005 ,  
      -0.35393 ,  0.15907 ,  0.55807 ,  0.1492 , -0.86433 ,  0.040305,  
      -1.0939 , -0.26386 , -0.29494 ,  0.25696 , -0.33718 , -0.086468,  
      -0.24246 , -0.21114 ,  0.099632,  0.12815 , -0.78714 , -0.51785 ,  
      -0.10944 ,  0.9763 ,  0.57032 ,  0.13581 ], dtype=float32)
```

Kodowanie
dokumentów
(embedding)

→ Zapis w bazie

Kodowanie tekstu w bazie do postaci wektorowej

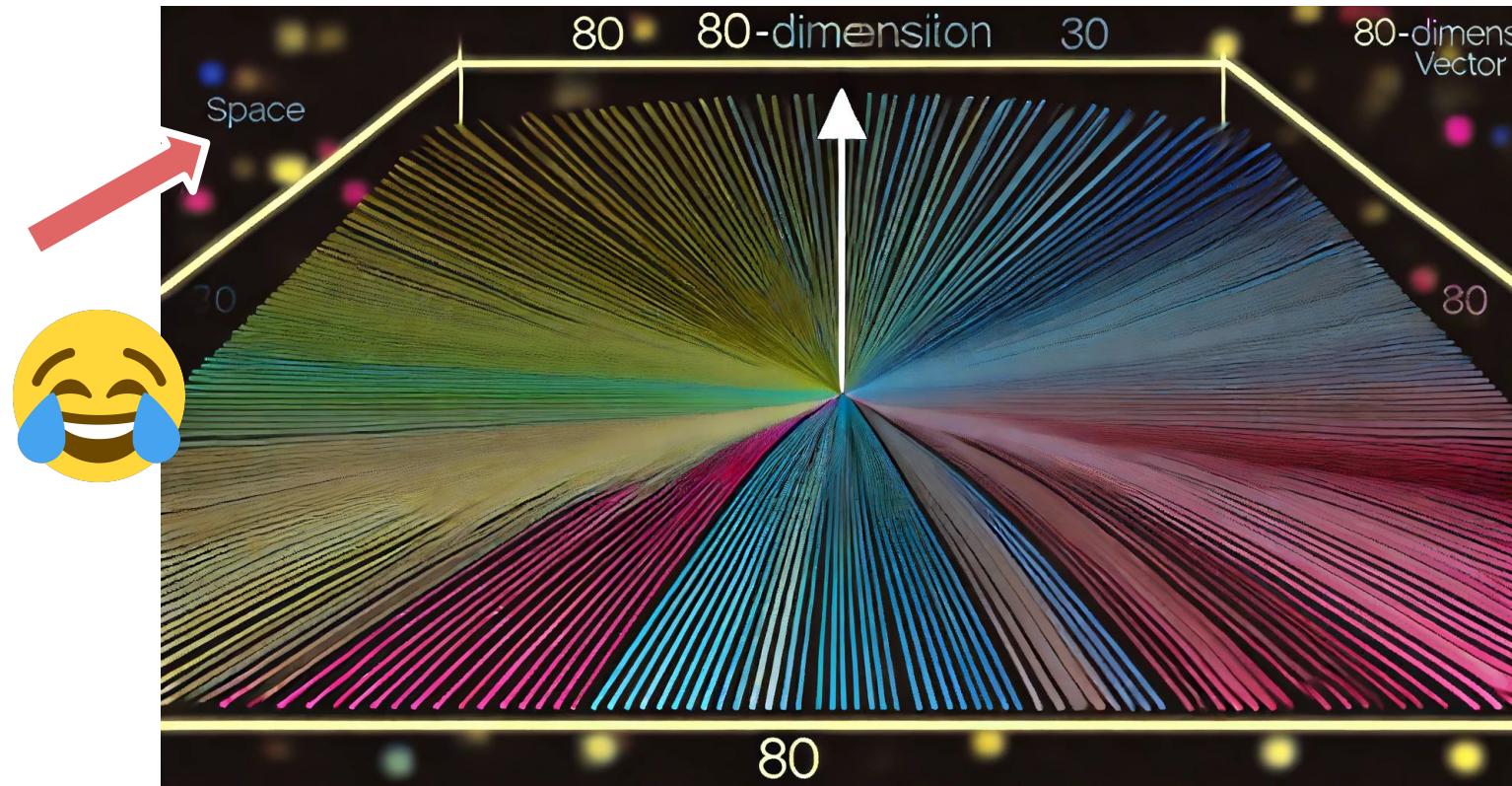
```
1 (model['go'] + model['away'])/2
```

```
array([-0.091342,  0.223404,  0.58856, -0.614765, -0.0838365,  
       0.5387,   -0.43531,  0.349125,  0.163308, -0.28223, ,  
       0.53547,   0.52797496,  0.096812,  0.2879,  -0.0533385, ,  
      -0.37232,   0.022637,  0.574705, -0.553275,  0.385575, ,  
       0.565335,   0.805405,  0.2579965,  0.0088565,  0.1674905, ,  
       0.25543,   -0.571035, -0.59926,  0.422585, -0.42896, ,  
      -0.389065,   0.19631,  -0.00933,  0.127285, -0.0487465, ,  
       0.381435,  -0.22540998,  0.021299, -0.1827915, -0.16490501,  
      -0.47944498, -0.431528,  -0.20091, -0.55665, -0.32982, ,  
      -0.088548,  -0.28038502,  0.219725,  0.090537, -0.67012, ,  
       0.0883085,  -0.19332,  0.0465725,  1.160815,  0.0691255, ,  
      -2.47895,   -0.33707,  0.083195,  1.86185,  0.283465, ,  
      -0.13081,   0.927795,  -0.37028,  0.1885465,  0.66198, ,  
       0.505175,  0.37748498,  0.1322995, -0.380375, -0.025135, ,  
      -0.1636765, -0.45639,  -0.047815, -0.87394,  0.0264145, ,  
       0.0117645,  -0.427415,  -0.31048, -0.317725, -0.02326, ,  
       0.525635,  0.05760051, -0.69786, -0.1213325, -1.2707, ,  
      -0.225355,  -0.1018815,  0.18575001, -0.30943, -0.211059, ,  
      -0.279545,  -0.16002001,  0.100371, -0.05461, -0.71834505, ,  
      -0.392925,  0.12075999,  0.61991,  0.582145,  0.20161, ],  
      dtype=float32)
```

Kodowanie
dokumentów
(embedding)

Zapis w bazie

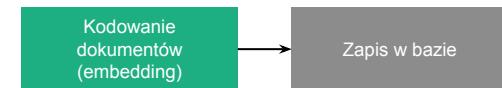
Create an image with a vector in 80 dimensional space



Kodowanie tekstu w bazie do postaci wektorowej

```
class TextEncoder extends AbstractGPTAPIClient implements StageInterface
{
    private string $embeddingModel = 'text-embedding-ada-002';

    public function getEmbeddings(string $document): string
    {
        $response = $this->client->embeddings()->create([
            'input' => $document,
            'model' => $this->embeddingModel
        ]);
        return json_encode($response->embeddings[0]->embedding);
    }
}
```



Wczytanie dokumentów

```
class DocumentLoader extends AbstractDocumentRepository
{
    public function loadDocuments(): void
    {
        $path      = __DIR__ . '/..../documents';
        $files = array_diff(scandir($path), array('.', '..'));
        foreach($files as $file) {
            $document = file_get_contents($path . '/' . $file);

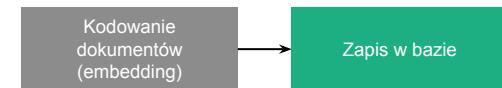
            #load documents to postgres database
            $responseDocument = $this->textEncoder->getEmbeddings($document);

            $this->insertDocument($document, $responseDocument);
        }
    }
}
```



Wczytanie dokumentów

```
CREATE EXTENSION IF NOT EXISTS vector;  
  
CREATE TABLE IF NOT EXISTS document (  
    id serial PRIMARY KEY,  
    embedding vector,  
    text text,  
    created_at timestampz DEFAULT now()  
);
```

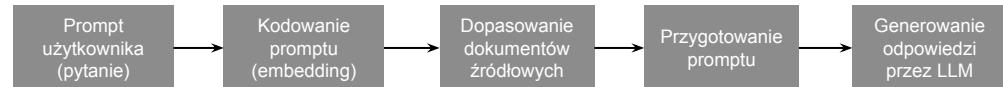


Proces przetwarzania zapytania

```
$promptResolver = new PromptResolver();
$textEncoder = new TextEncoder();
$documentProvider = new DocumentProvider();
$ragPromptProvider = new RAGPromptProvider();
$generatedTextProvider = new GeneratedTextProvider();
$payload = new Payload();

$pipeline = (new Pipeline(new FingersCrossedProcessor()))
    ->pipe($promptResolver) //get prompt from POST or CLI
    ->pipe($textEncoder) //get embeddings for prompt
    ->pipe($documentProvider) //find documents with similarity to prompt
    ->pipe($ragPromptProvider) //combine prompt with matched source documents
    ->pipe($generatedTextProvider); //get API response

$response = $pipeline->process($payload);
```



Zapytanie użytkownika

Find answer in websites database

What is specialization of programmer and lecturer Michał Żarnecki based on his website content.

Generate answer



Zapytanie użytkownika

You asked: what is specialization of Michał Żarnecki based on his website content

Response: Based on the content of Michał Żarnecki's portfolio website, his specialization lies in the following areas:

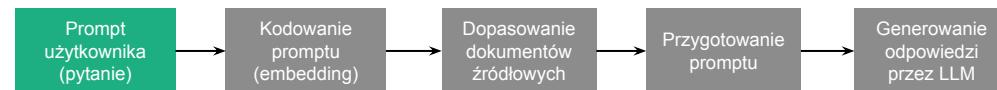
1. **Programming Languages**: Proficient in Python, PHP, and JavaScript.
2. **System Design and Development**: Expertise in designing and developing systems and solutions.
3. **Mobile and Web Applications**: Experienced in creating mobile applications for platforms like Android and iOS.
4. **ERP/CRM Systems**: Extensive experience in developing ERP and CRM systems for various industries.
5. **Artificial Intelligence (AI) and Machine Learning (ML)**: Skilled in AI/ML-related technologies, including deep learning.
6. **Data Analysis and Processing**: Involved in data mining, text mining, and handling big data.
7. **Project Management**: Proficient in technical project management and optimization.
8. **API Development and Integration**: Developing and integrating API services.
9. **Testing and Optimization**: Conducting unit tests, interface tests with Selenium IDE, and optimizing performance.
10. **Technologies and Tools**: Familiar with various technologies and tools such as Docker, AWS, PyTorch, and Kubernetes.

In summary, Michał Żarnecki is specialized in software development with a focus on AI/ML, ERP/CRM systems, and system design.

Find answer in websites database.

What is specialization of programmer and lecturer Michał Żarnecki based on his website content...

Ctrl+D to submit

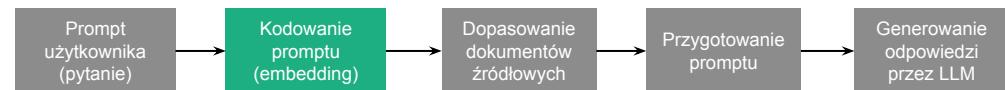


Kodowanie tekstu w bazie do postaci wektorowej

```
class TextEncoder extends AbstractGPTAPIClient implements StageInterface
{
    private string $embeddingModel = 'text-embedding-ada-002';

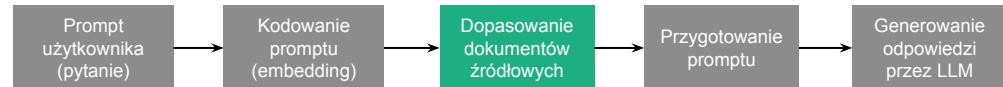
    public function getEmbeddings(string $document): string
    {
        $response = $this->client->embeddings()->create([
            'input' => $document,
            'model' => $this->embeddingModel
        ]);
        return json_encode($response->embeddings[0]->embedding);
    }

    /**
     * @param Payload $payload
     * @return Payload
     */
    public function __invoke($payload)
    {
        return $payload->setEmbeddingPrompt($this->getEmbeddings($payload->getPrompt()));
    }
}
```

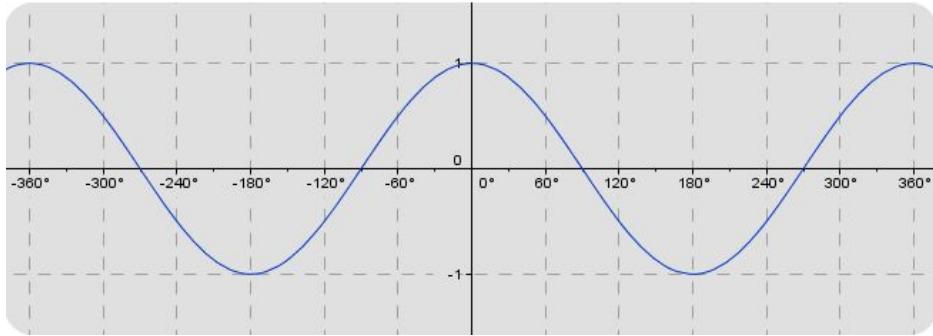
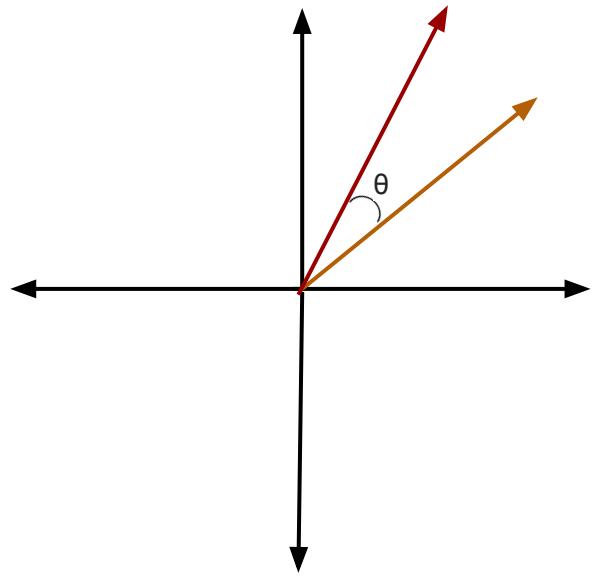


Dopasowanie "pasujących" dokumentów

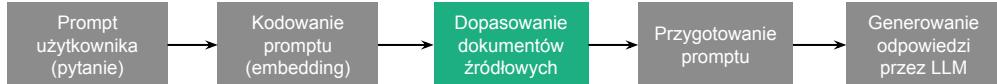
```
SELECT text FROM document order by embedding <=>  
'[-0.0014472235,-0.0001540061,0.0052023693,...]' DESC limit 3;
```



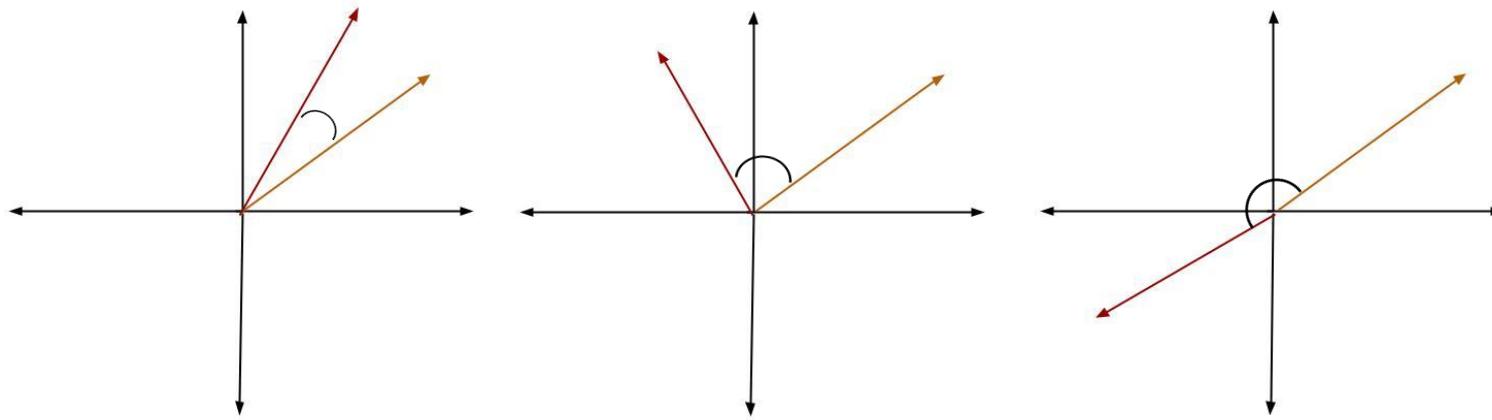
Podobieństwo cosinusowe



$$\cos \theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$



Dopasowanie "pasujących" dokumentów



Dopasowanie "pasujących" dokumentów

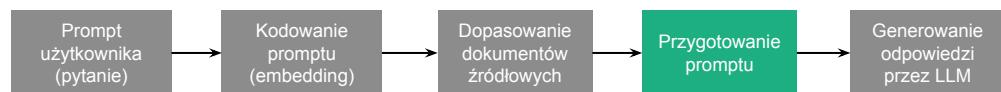
```
class DocumentProvider extends AbstractDocumentRepository implements StageInterface
{
    public function getSimilarDocuments(string $embeddingPrompt): array
    {
        $stmt = $this->connection->prepare("SELECT text from document order by
embedding <=> :embeddingPrompt DESC limit 5;");
        $stmt->execute(['embeddingPrompt' => $embeddingPrompt]);
        return $stmt->fetchAll();
    }
}
```



Prompt engineering + one shot learning



by DALL-E



Prompt engineering + one shot learning

INPUT:

You are a helpful AI assistant with access to a set of websites content. Your role is to provide information and answer questions based solely on these documents.

You should respond directly and concisely, using the information contained within the documents without quoting or revealing actual document content. Do not infer or guess information that is not explicitly stated in the documents.

If a question relates to information not present in the documents, state that the information is not available.

Your goal is to be helpful by providing factual and document-backed answers.

Here is question:

<QUESTION>

Source documents:

<DOCUMENTS>

Example:

when receive question and content like below:

<WEBSITE CONTENT>

you should answer like below:

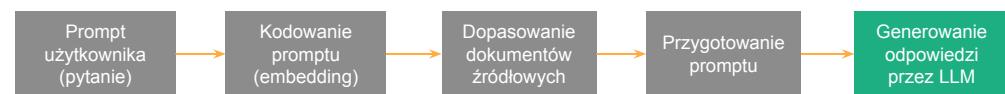
<EXPECTED ANSWER>



Model



by DALL-E



Model



GPT4o

API OPENAI

175 mld parametrów

ANTHROPIC

CLAUDE 3

API AWS BEDROCK

2 tryliony parametrów



MIXTRAL

API / STANDALONE

45 mld parametrów



LLAMA3

API / STANDALONE

70 mld parametrów

kontekst ~15 str. po 400 słów



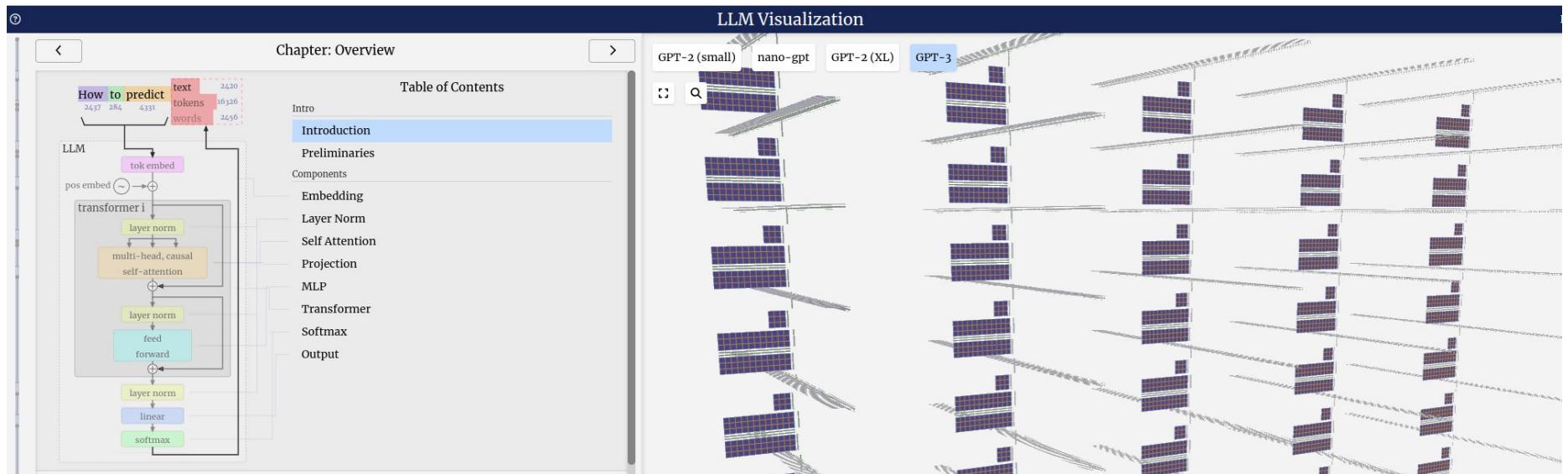
open source models available on:



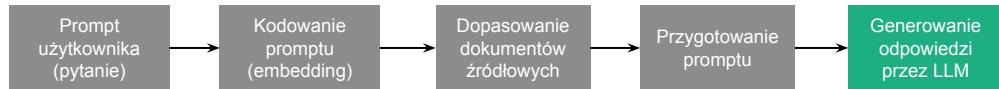
Hugging Face



Model



<https://bbycroft.net/llm>



Zadanie

Zadanie:

Uruchom <https://bbycroft.net/llm> i opisz (bardzo ogólnie) różnice w budowie GPT-2 (small) i GPT-3.

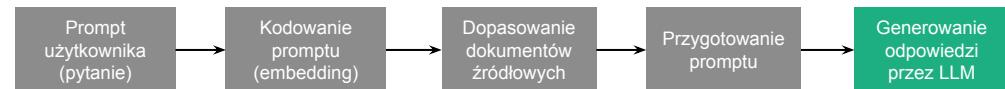
Komunikacja z API

```
class GeneratedTextProvider extends AbstractGPTAPIClient implements StageInterface
{
    private string $model = 'gpt-4o';

    public function generateText(string $prompt, string $sourceDocuments): string
    {
        # prepare API input
        $input = $sourceDocuments . "\n\n##### INPUT: \n" . $prompt . "\n##### RESPONSE:\n";

        # get API response
        $response = $this->client->chat()->create([
            'model' => $this->model,
            'messages' => [
                [
                    'content' => $input,
                    'role' => 'user'
                ]
            ]
        ]);

        return $response->choices[0]->message->content;
    }
}
```



Przeszukiwanie treści stron internetowych

INPUT:

what is specialization of Michał Żarnecki based on website content?

Przeszukiwanie treści stron internetowych

INPUT:

what is specialization of Michał Żarnecki based on website content?

RESPONSE:

Michał Żarnecki is a programmer and lecturer specializing in several key areas related to modern software development and data-driven technologies.

His expertise includes:

1. **Programming Languages**: Python, PHP, JavaScript.
2. **AI and Machine Learning**: Designing systems and solutions related to artificial intelligence and machine learning.
3. **Data Mining and Big Data**: Extracting valuable insights from large datasets.
4. **Natural Language Processing (NLP)**: Working on systems that understand and generate human language.
5. **Software Development Frameworks**: Utilizing various tools and frameworks such as Streamlit, TensorFlow, PyTorch, and langchain.
6. **Database Systems**: Implementing and working with databases like PostgreSQL, Elasticsearch, Neo4j, and others.

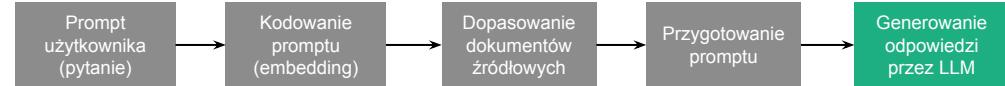
His portfolio highlights projects such as an AI chatbot for analyzing company documents and a self-driving vehicle based on TensorFlow and Raspberry Pi.

Additionally, he has contributed to conferences and created e-learning courses focused on machine learning, underscoring his dual role as a developer and educator.

Fine tuning (opcjonalnie)



by DALL-E





Michał Żarnecki

Audio engineer :



Michał Żarnecki (ur. 12 listopada 1946 w Warszawie, zm. 21 listopada 2016, tamże) – **polski operator i reżyser dźwięku.**

 Wikipedia

[https://pl.wikipedia.org › wiki › Michał_Żarnecki](https://pl.wikipedia.org/wiki/Michał_Żarnecki)

[Michał Żarnecki – Wikipedia, wolna encyklopedia](#)

Born: November 12, 1946, Warsaw

Died: November 21, 2016, Warsaw

Nominations: Polish Academy Award for Best Sound

Siblings: Andrzej Zarnecki

W jaki sposób przetwarzać teksty, aby uniknąć pomyłek?

Find answer in
websites database

What is specialization of programmer and lecturer Michał Żarnecki based on his website content.

Generate answer

Przeszukiwanie treści stron internetowych

INPUT:

Is Michał Żarnecki programmer the same person as Michał Żarnecki audio engineer?

Przeszukiwanie treści stron internetowych

INPUT:

Is Michał Żarnecki programmer the same person as Michał Żarnecki audio engineer?

RESPONSE:

Based on the information provided, it appears that Michał Żarnecki the programmer and Michał Żarnecki the audio engineer are not the same person.

Here's why:

1. **Fields of Expertise**:

- Michał Żarnecki, the audio engineer, was a well-known operator and reżyser dźwięku (sound director) in Poland, with notable contributions to the Polish film industry, as mentioned in the Wikipedia excerpt.

- Michal Żarnecki, the programmer, has a portfolio focused on programming in Python, PHP, and JavaScript, with projects related to AI, machine learning, data mining, and software development.

2. **Lifespan**:

- **Michał Żarnecki the audio engineer was born on November 12, 1946, and passed away on November 21, 2016.**

- **The projects listed in Michał Żarnecki the programmer's portfolio date from 2014 to 2016, which would be conflicting if he had passed away in 2016 and was actively working in those years.**

3. **Occupational Focus**:

- The audio engineer has a career documented in film sound engineering and education.

- The programmer's career is centered around software development, mobile applications, ERP systems, and consulting in technology.

Given the distinct differences in their professional domains, timelines, and expertise, it is highly unlikely that they are the same individual

Przeszukiwanie treści stron internetowych

INPUT:

Is Michał Żarnecki programmer the same person as Michał Żarnecki audio engineer?

RESPONSE:

Based on the information provided, it appears that Michał Żarnecki the programmer and Michał Żarnecki the audio engineer are not the same person.

Here's why:

1. **Fields of Expertise**:

- Michał Żarnecki, the audio engineer, was a well-known operator and reżyser dźwięku (sound director) in Poland, with notable contributions to the Polish film industry, as mentioned in the Wikipedia excerpt.

- Michal Żarnecki, the programmer, has a portfolio focused on programming in Python, PHP, and JavaScript, with projects related to AI, machine learning, data mining, and software development.

2. **Lifespan**:

- **Michał Żarnecki the audio engineer was born on November 12, 1946, and passed away on November 21, 2016.**

- **The projects listed in Michał Żarnecki the programmer's portfolio date from 2014 to 2016, which would be conflicting if he had passed away in 2016 and was actively working in those years.**

3. **Occupational Focus**:

- The audio engineer has a career documented in film sound engineering and education.

- The programmer's career is centered around software development, mobile applications, ERP systems, and consulting in technology.

Given the distinct differences in their professional domains, timelines, and expertise, it is highly unlikely that they are the same individual



Wyzwania

- **wydajność zapytania dopasowującego źródła przy milionach rekordów**

Wyzwania

- **wydajność zapytania dopasowującego źródła przy milionach rekordów**

```
CREATE INDEX ON document
    USING hnsw (embedding vector_cosine_ops);
```

Wyzwania

- wydajność zapytania dopasowującego źródła przy milionach rekordów
- **halucynowanie i ewaluacja modelu**

Wyzwania

- wydajność zapytania dopasowującego źródła przy milionach rekordów
- **halucynowanie i ewaluacja modelu**



by DALL-E

Wyzwania

- wydajność zapytania dopasowującego źródła przy milionach rekordów
- halucynowanie i ewaluacja modelu
- **wybór właściwych dokumentów źródłowych**

Wyzwania

- wydajność zapytania dopasowującego źródła przy milionach rekordów
- halucynowanie
- **wybór właściwych dokumentów źródłowych**

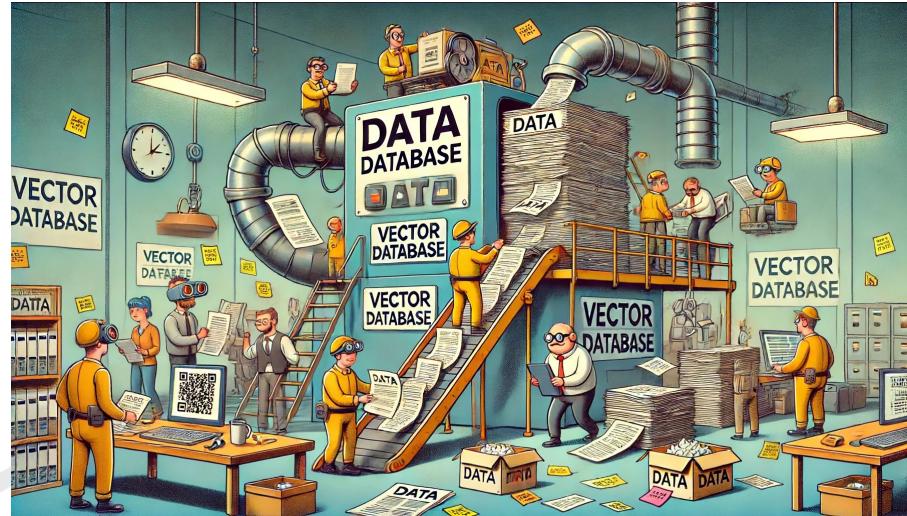
metadata column:

```
{  
    "company_id": "3b011094-df66-451e-a55c-94fd204f2f72",  
    "company_nameCobol&Fortran Innovations inc.",  
    "file_type": "excerpt",  
    "file_nameChronologischer Handelsregisterauszug",  
    "entry_id": 2,  
    "date2022-08-31",  
    "source_document_id": "b7600b45-8f89-4600-be7c-58714c11bca5"  
}
```

Jak wdrożyć RAG w projekcie?

Jak wdrożyć RAG w projekcie?

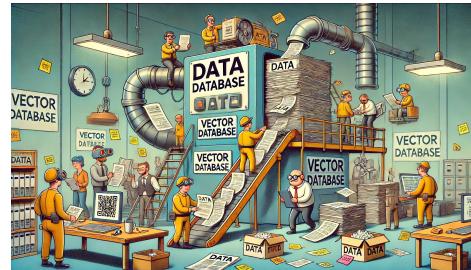
1. Pozyskanie danych do bazy wektorowej



by DALL-E

Jak wdrożyć RAG w projekcie?

1. Pozyskanie danych do bazy wektorowej



2. Import i konfiguracja modelu lub API LLM



by DALL-E

Jak wdrożyć RAG w projekcie?

1. Pozyskanie danych do bazy wektorowej

2. Import i konfiguracja modelu lub API LLM



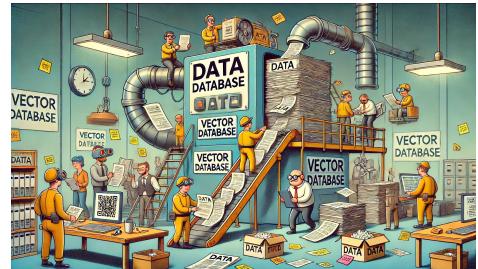
3. Przygotowanie promptu



by DALL-E

Jak wdrożyć RAG w projekcie?

1. Pozyskanie danych do bazy wektorowej



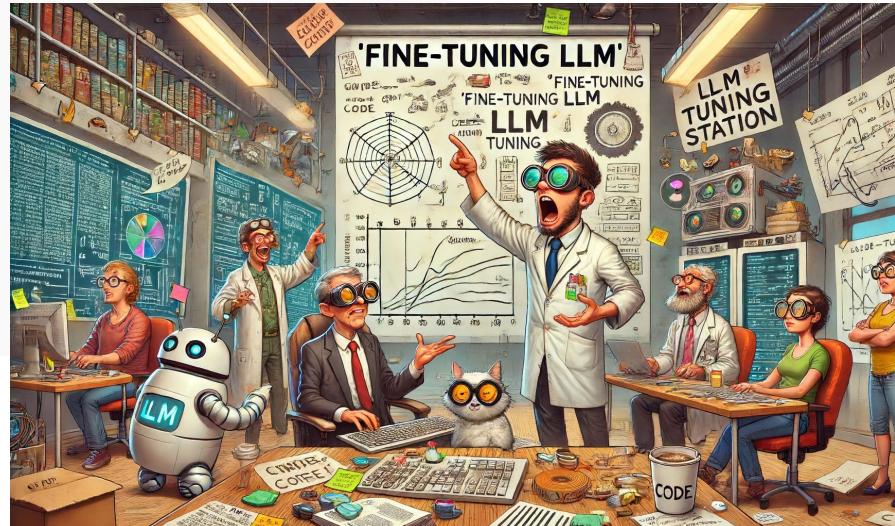
2. Import i konfiguracja modelu lub API LLM



3. Przygotowanie promptu

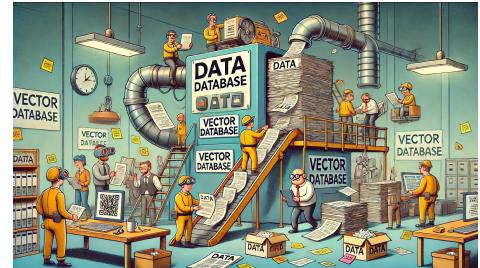


4. Opcjonalnie:
fine tuning modelu



Jak wdrożyć RAG w projekcie?

1. Pozyskanie danych do bazy wektorowej



2. Import i konfiguracja modelu lub API LLM



3. Przygotowanie promptu



4. Opcjonalnie:
fine tuning modelu



Nie stosuj modelu uczenia maszynowego, gdy...

*Nie stosuj modelu uczenia
maszynowego, jeżeli w danym
problemie nie ma wyraźnej
przewagi nad “tradycyjnym”
podejściem.*

Nie stosuj modelu uczenia maszynowego, gdy...

*Nie stosuj modelu uczenia
maszynowego, jeżeli w danym
problemie nie ma wyraźnej
przewagi nad "tradycyjnym"
podejściem.*



by DALL-E

Zadanie

Zadanie:

realizacja w 3-osobowych grupach

1. Uruchom projekt w dockerze <https://github.com/rzarno/php-rag>
2. Zadaj kilka pytań związanych ze zbiorem stron internetowych
3. Podmień zbiór dokumentów w folderze php-rag/app/src/documents. Usuń pliki tekstowe stron internetowych i wstaw 3-5 dowolne pliki tekstowe. Uruchom ponownie aplikację i sprawdź, czy potrafi odpowiedzieć na pytania dotyczące informacji zawartych w nowych tekstuach.