

シナリオデータに基づくLLMを用いた動画の生成手法の提案

創発ソフトウェア研究室

B4 河地 駿太郎

1. はじめに
2. 要素技術
3. 提案手法
4. 実験
5. 考察
6. まとめと今後の課題

発表の流れ

1. はじめに
2. 要素技術
3. 提案手法
4. 実験
5. 考察
6. まとめと今後の課題



はじめに

動画コンテンツの重要性

公知のミームの使用

- 必要な素材の数を抑制
- 容易に動画制作が可能

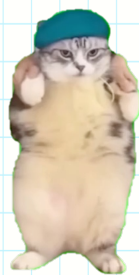
はじめに

ミーム

- インターネット上で Web サイトや SNS を通して拡散され, 話題になった文章や画像
- 進化生物学者リチャード・ドーキンスが命名
- 人から人へ広がる行動やアイデアの概念

はじめに

猫ミームの例



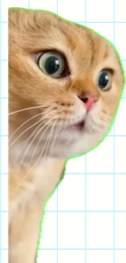
Girlfriend 猫



しょんぼり猫



ハア猫



怒られる猫



驚き猫



ハッピー猫

はじめに

猫ミーム動画例



はじめに

本研究の目的

テキストから必要な情報を抽出し、
ミームを用いた動画の生成

1. テキストの解析
2. 背景画像とミーム素材の決定
3. 素材を使用した動画の生成

はじめに

様々な動画生成AIが存在

高度な映像を使用せず, シナリオを重視

猫ミーム: 背景画像 + ミーム素材 + テキスト

構成要素が**単純**で実装が**容易**

1. はじめに
2. 要素技術
3. 提案手法
4. 実験
5. 考察
6. まとめと今後の課題

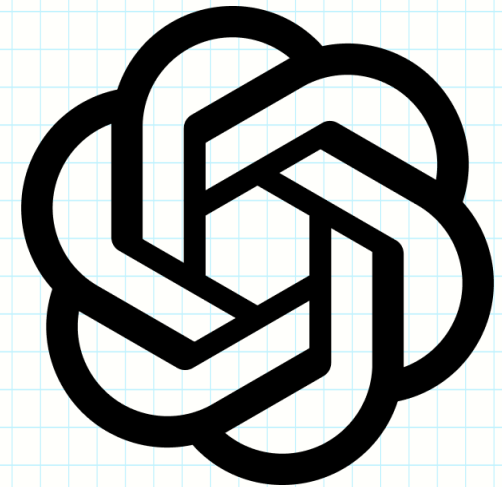


LLM (Large Language Models)

- 大規模なデータセットを使用して訓練
- 自然言語処理モデル
- ゼロショットや少数ショット学習により
様々なタスクを実現
 - 自然言語理解
 - 感情分析
 - 文章生成

GPT (Generative Pre-trained Transformer)

- OpenAI の提供する
強力な自然言語処理モデル
- 高速で効率的な処理を実現
- 複雑な文脈を理解

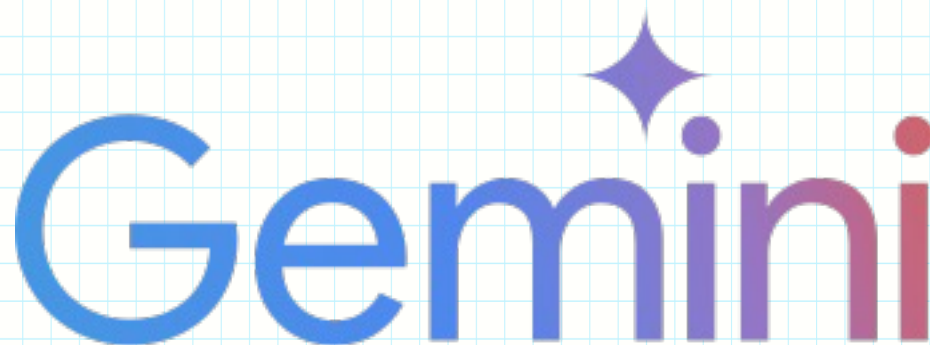


Faiss (Facebook AI Similarity Search)

- 高次元ベクトルの類似性探索,
クラスタリングのためのライブラリ
- 高速かつスケーラブルな処理を実現
 - メモリ内での検索
 - インデックスの圧縮

Gemini

- Google が開発した大規模言語モデル
- テキストや画像などの複数のデータを処理
- 文脈理解や応答精度が優秀



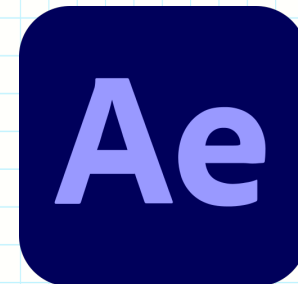
SQL (Structured Query Language)

- リレーショナルデータベースのデータを操作するための言語
- データの検索, 追加, 更新, 削除等が可能



Adobe After Effects (AE)

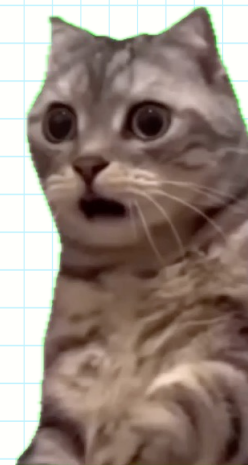
- Adobe 社が提供する映像制作ソフト
- 高品質なモーショングラフィックやビジュアルエフェクトを動画に加えるツール
- 外部エディタからの実行が可能



ExtendScript

- Adobe 製品向けのスクリプト言語
- JavaScript ベース
- Adobe 製品の自動化や拡張に使用
- After Effects へのプログラムの実行に使用

1. はじめに
2. 要素技術
3. 提案手法
4. 実験
5. 考察
6. まとめと今後の課題



シナリオの解析→動画の生成

1. テキストから情報を抽出
2. データベースを使用して, 背景画像とミーム素材のファイルパスを入手
3. ファイルパスとテキスト情報をもとに ExtendScript を作成, 実行

1. テキストからの情報の抽出

- プロンプトを使用して, 入力文から時間, 場所, 登場人物の状態, テキスト情報を入手
出力を正規表現で処理し, 使用
- モデルとして Gemini-1.5-flash を使用
temperature = 1.0

2. ファイルパスの決定

- 背景画像のファイルパスの決定には Faiss を使用
- ミーム素材のファイルパスの決定には モデルとして GPT-4o-mini を使用
temperature = 0.5

2. ファイルパスの決定(背景画像)

表1. 表 images の構成

image_id	location_id	time_condition_id	file_path
1	1	3	/Users/...
2	1	2	/Users/...
3	1	1	/Users/...
4	2	3	/Users/...

2. ファイルパスの決定(背景画像)

表2. 表 locations の構成

location_id	location_name
1	ATM コーナー
2	アーケード商店街
3	アイランドキッチン
4	アジト

2. ファイルパスの決定(背景画像)

表3. 表 time_conditions の構成

time_condition_id	time_and_condition
1	日中
2	夜
3	夕方
4	夜・照明 OFF

2. ファイルパスの決定(背景画像)

表1. 表 images の構成

image_id	location_id	time_condition_id	file_path
1	1	3	/Users/...
2	1	2	/Users/...
3	1	1	/Users/...
4	2	3	/Users/...

2. ファイルパスの決定(ミーム素材)

表4. 表 Meme_Features の構成

meme_id	feature_id	file_path
1	1	/Users/...
2	2	/Users/...
2	3	/Users/...
3	3	/Users/...

2. ファイルパスの決定(ミーム素材)

表5. 表 Memes の構成

meme_id	meme_name
1	DJ 猫
2	EDM 猫
3	Girlfriend 猫
4	oia 猫

2. ファイルパスの決定(ミーム素材)

表6. 表 Features の構成

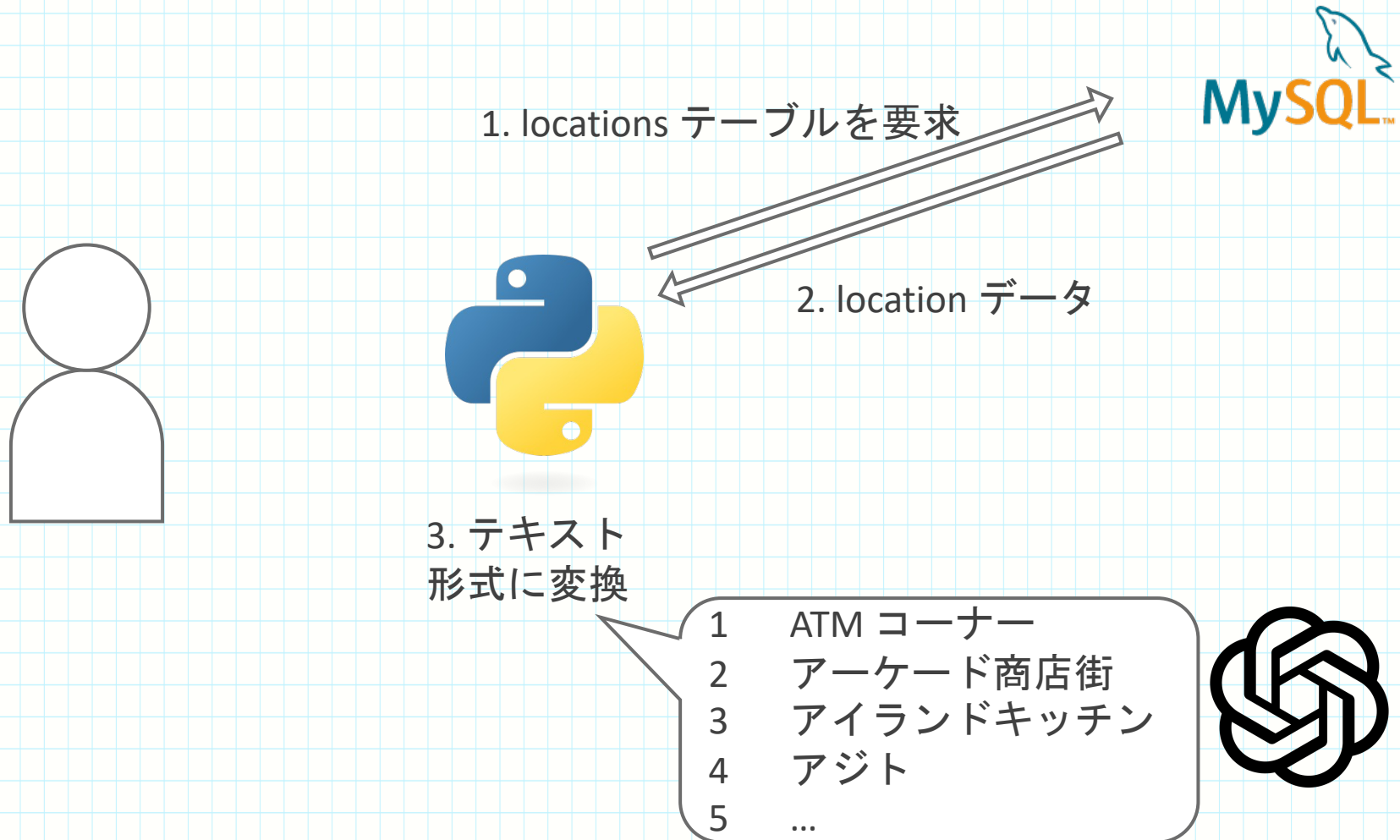
feature_id	feature_name
37	1 匹の猫を叩く
2	EDM
9	いびきをかく
5	キーボード

2. ファイルパスの決定(ミーム素材)

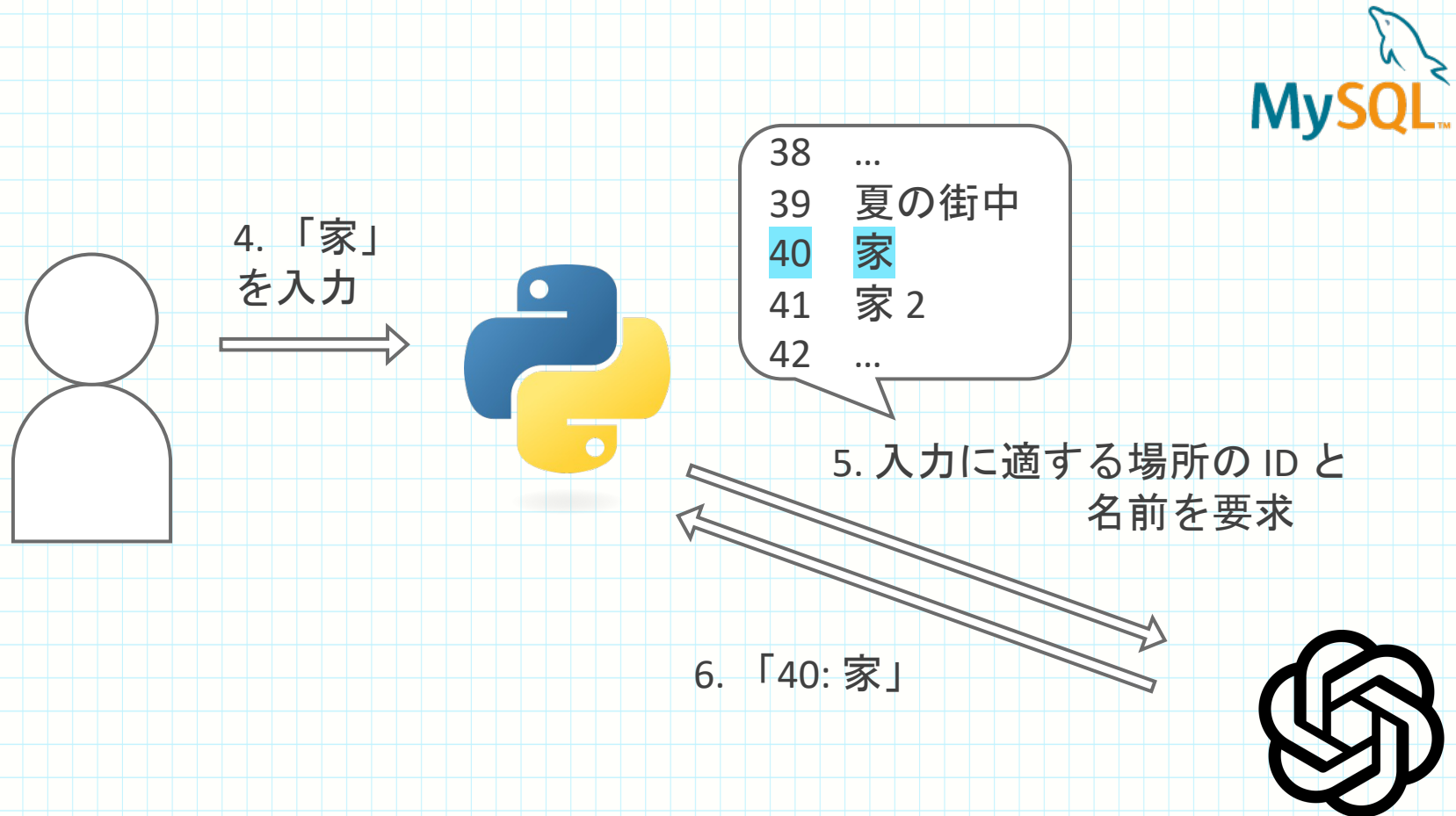
表4. 表 Meme_Features の構成

meme_id	feature_id	file_path
1	1	/Users/...
2	2	/Users/...
2	3	/Users/...
3	3	/Users/...

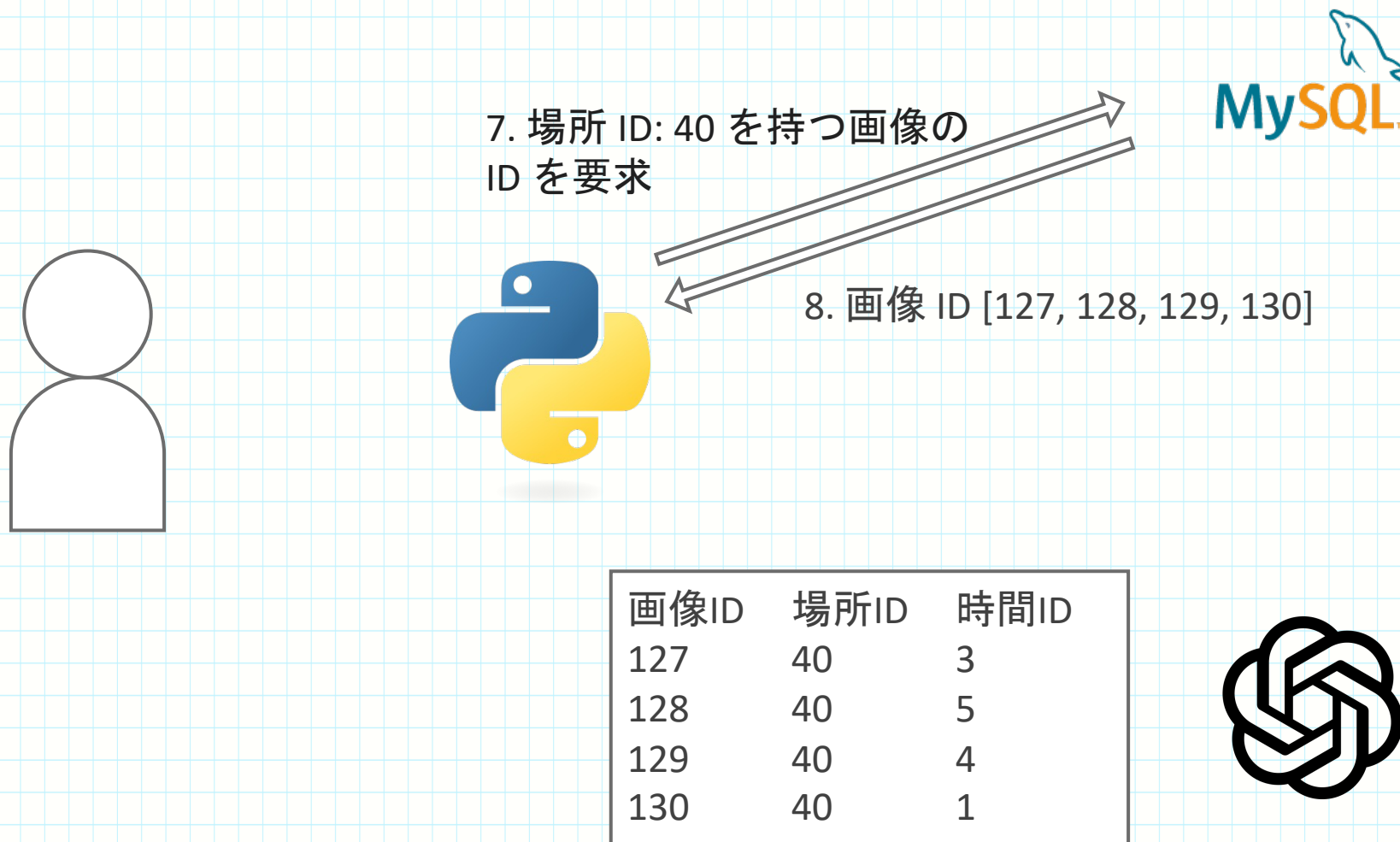
システムの概要(1)



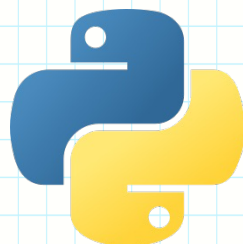
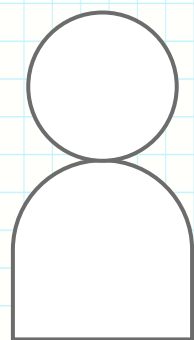
システムの概要(2)



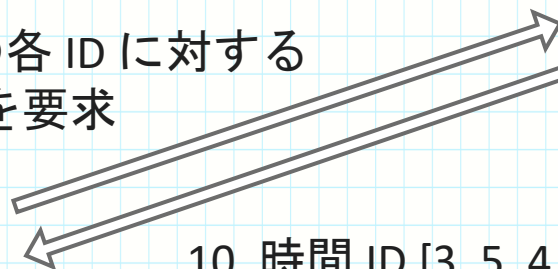
システムの概要(3)



システムの概要(4)

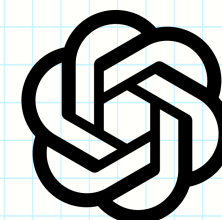


9. 画像の各 ID に対する
時間 ID を要求

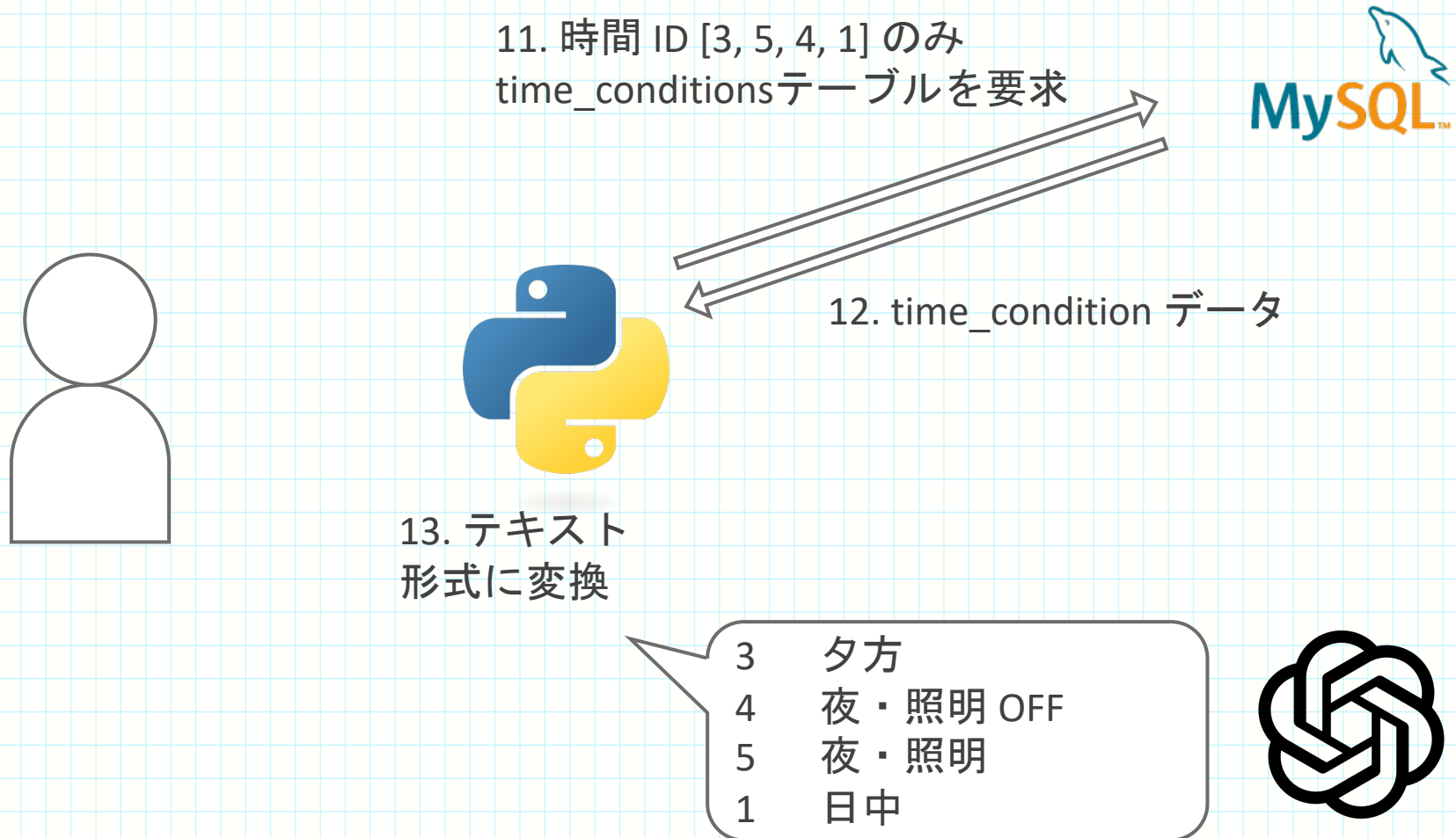


10. 時間 ID [3, 5, 4, 1]

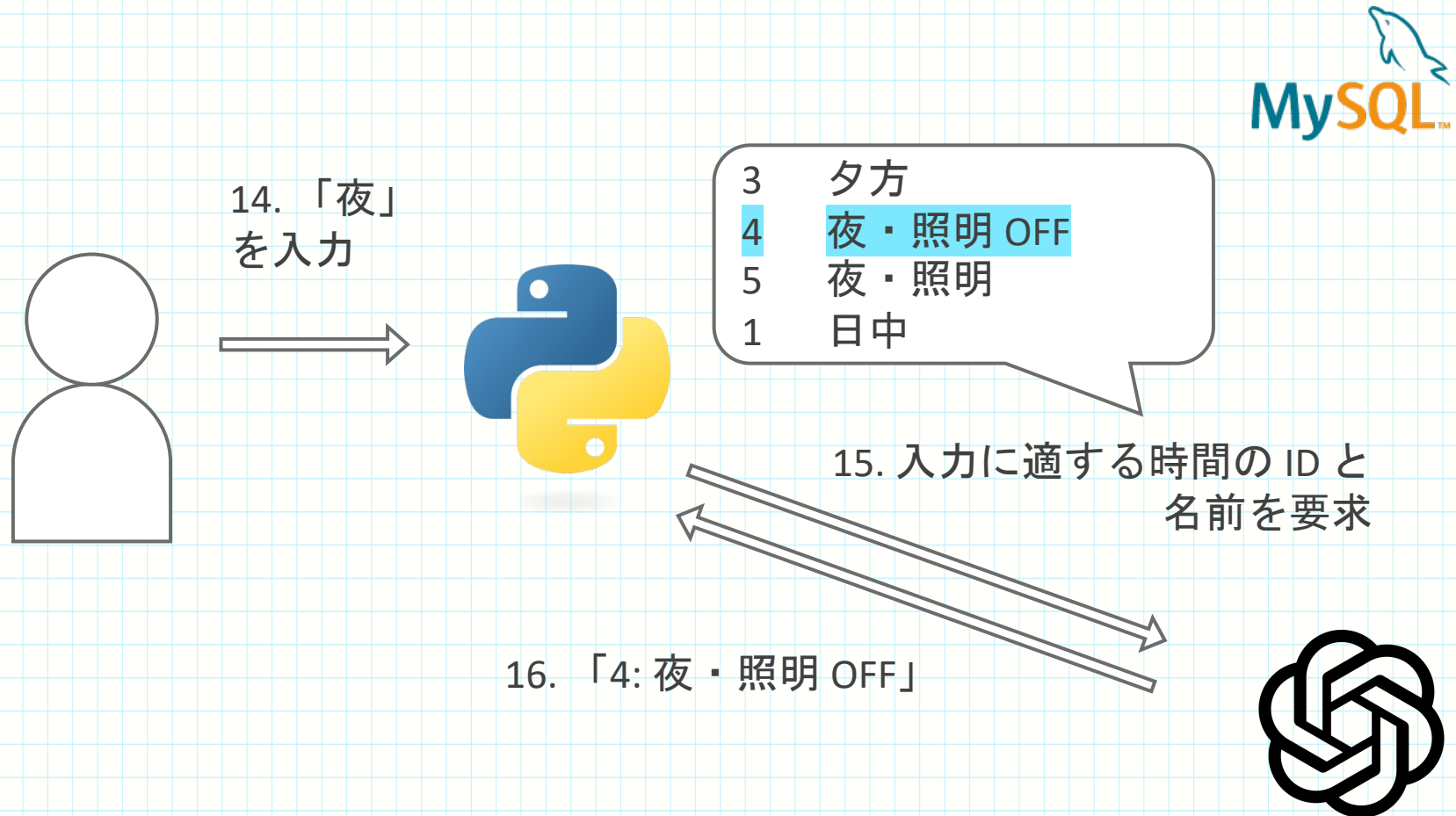
画像ID	場所ID	時間ID
127	40	3
128	40	5
129	40	4
130	40	1



システムの概要(5)



システムの概要(6)



システムの概要(7)

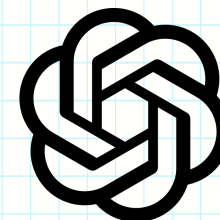
17. 場所 ID と時間 ID をもとに、
画像 ID を要求



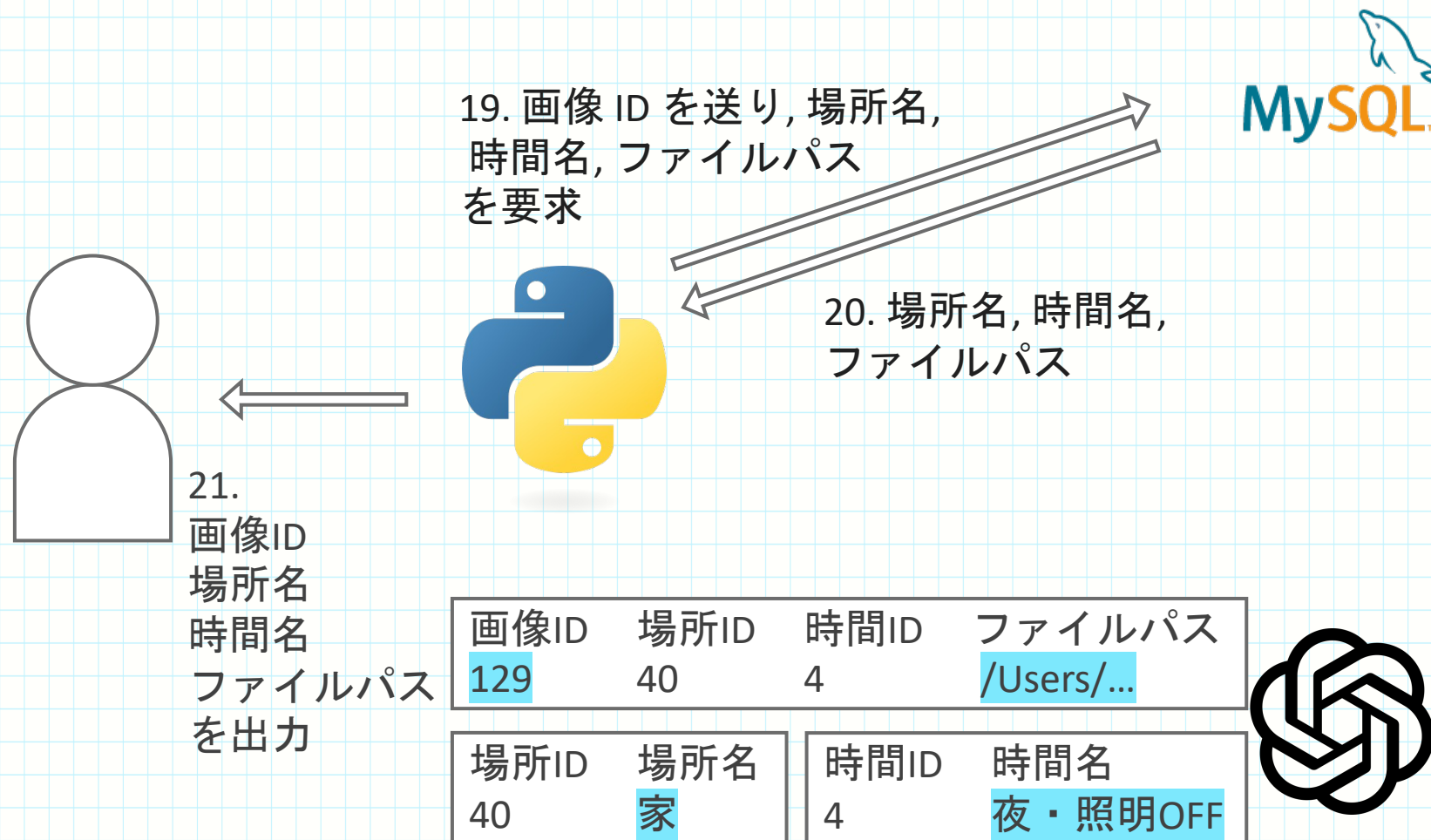
18. 「画像 ID: 129」



画像ID	場所ID	時間ID
127	40	3
128	40	5
129	40	4
130	40	1



システムの概要(8)



3. ExtendScriptの生成, 実行

- 動画の長さ, 素材の位置やサイズを指定したテンプレートにテキスト情報とファイルパスを入力

ExtendScript ファイルを出力, これを実行

1. はじめに
2. 要素技術
3. 提案手法
4. 実験
5. 考察
6. まとめと今後の課題



実験方法

- 実験 1

- テキストからの情報の抽出の際,
GPT と Gemini を使用して結果を比較(5 回)
- 入力文: 勉強している時は、一休みすると気分が
楽になる。

- 実験 2

- 背景画像のファイルパスの決定の際,
GPT と Faiss を使用して結果を比較
- 異なる 5 つの時間と場所を入力

実験 1:

表 7. GPT を使用して抽出した結果

時間	場所	登場人物の状態	テキスト情報
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる

実験 1:

表 8. Gemini を使用して抽出した結果

時間	場所	登場人物の状態	テキスト情報
勉強中	不明	疲れている	気分転換が必要
勉強中	不明	疲れている	気分転換必要
勉強中	不明	疲れている	気分転換が必要
勉強中	不明	疲れている	気分転換必要
勉強中	不明	疲れている	一休みで気分転換

実験

43

実験 1: Geminiを使用



実験 2:

表 9. GPT を使用して得られた結果

入力場所	入力時間	出力場所	出力時間
ファミレス	昼	レストラン	日中
図書館	夜	市立図書館	照明ON
学校	昼休み	学校のベンチ	日中
街中	夕方	街中のビル	夕方
自宅	夜	家	夜・照明OFF

実験 2:

表 10. Faiss を使用して得られた結果

入力場所	入力時間	出力場所	出力時間
ファミレス	昼	アジト	照明ON
図書館	夜	図書室	夕方
学校	昼休み	学校のベンチ	日中
街中	夕方	都会の街中	夕方
自宅	夜	家	夕方

1. はじめに
2. 要素技術
3. 提案手法
4. 実験
5. 考察
6. まとめと今後の課題



実験 1 の考察

- GPT, Gemini のいずれも時間と場所は同様の結果
- 登場人物の状態とテキスト情報には明確な違い
 - GPT: 「一休み中」, 「気分が楽になる」
入力文から直接情報を抽出
 - Gemini: 「疲れている」, 「気分転換が必要」
入力文から, 登場人物が疲れていると想定

実験 1 の考察

表 7. GPT を使用して抽出した結果

時間	場所	登場人物の状態	テキスト情報
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる
勉強中	不明	一休み中	気分が楽になる

実験 1 の考察

表 8. Gemini を使用して抽出した結果

時間	場所	登場人物の状態	テキスト情報
勉強中	不明	疲れている	気分転換が必要
勉強中	不明	疲れている	気分転換必要
勉強中	不明	疲れている	気分転換が必要
勉強中	不明	疲れている	気分転換必要
勉強中	不明	疲れている	一休みで気分転換

実験 2 の考察

- 「ファミレス」のようなデータベースに直接存在しない単語を使用
 - GPT: 正確に解釈
LLM を使用しているため, 高度な検索が可能
 - Faiss: 存在していない単語や略称の処理に失敗
一方でオフラインでも実行可能

実験 2:

表 9. GPT を使用して得られた結果

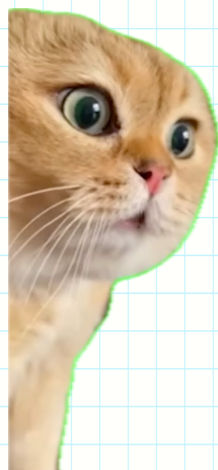
入力場所	入力時間	出力場所	出力時間
ファミレス	昼	レストラン	日中
図書館	夜	市立図書館	照明ON
学校	昼休み	学校のベンチ	日中
街中	夕方	街中のビル	夕方
自宅	夜	家	夜・照明OFF

実験 2:

表 10. Faiss を使用して得られた結果

入力場所	入力時間	出力場所	出力時間
ファミレス	昼	アジト	照明ON
図書館	夜	図書室	夕方
学校	昼休み	学校のベンチ	日中
街中	夕方	都会の街中	夕方
自宅	夜	家	夕方

1. はじめに
2. 要素技術
3. 提案手法
4. 実験
5. 考察
6. まとめと今後の課題



まとめと今後の課題

本研究の成果

1. 正確なテキストの解析
2. データベースを使用した, 背景画像とミーム素材の正確なファイルパスの決定
3. 素材を使用した動画の生成

を自動化するスクリプトの作成

まとめと今後の課題

今後の課題

- 背景画像の自動生成
- ミーム素材の生成の検討