Creating a Board Game Chatbot with Postgres, AI, and RAG

Matt Cornillon

Customer Engineer, Database specialist

Google Cloud

Creating a Board Game Chatbot with Postgres, AI, and RAG

Matt Cornillon

Customer Engineer, Database specialist

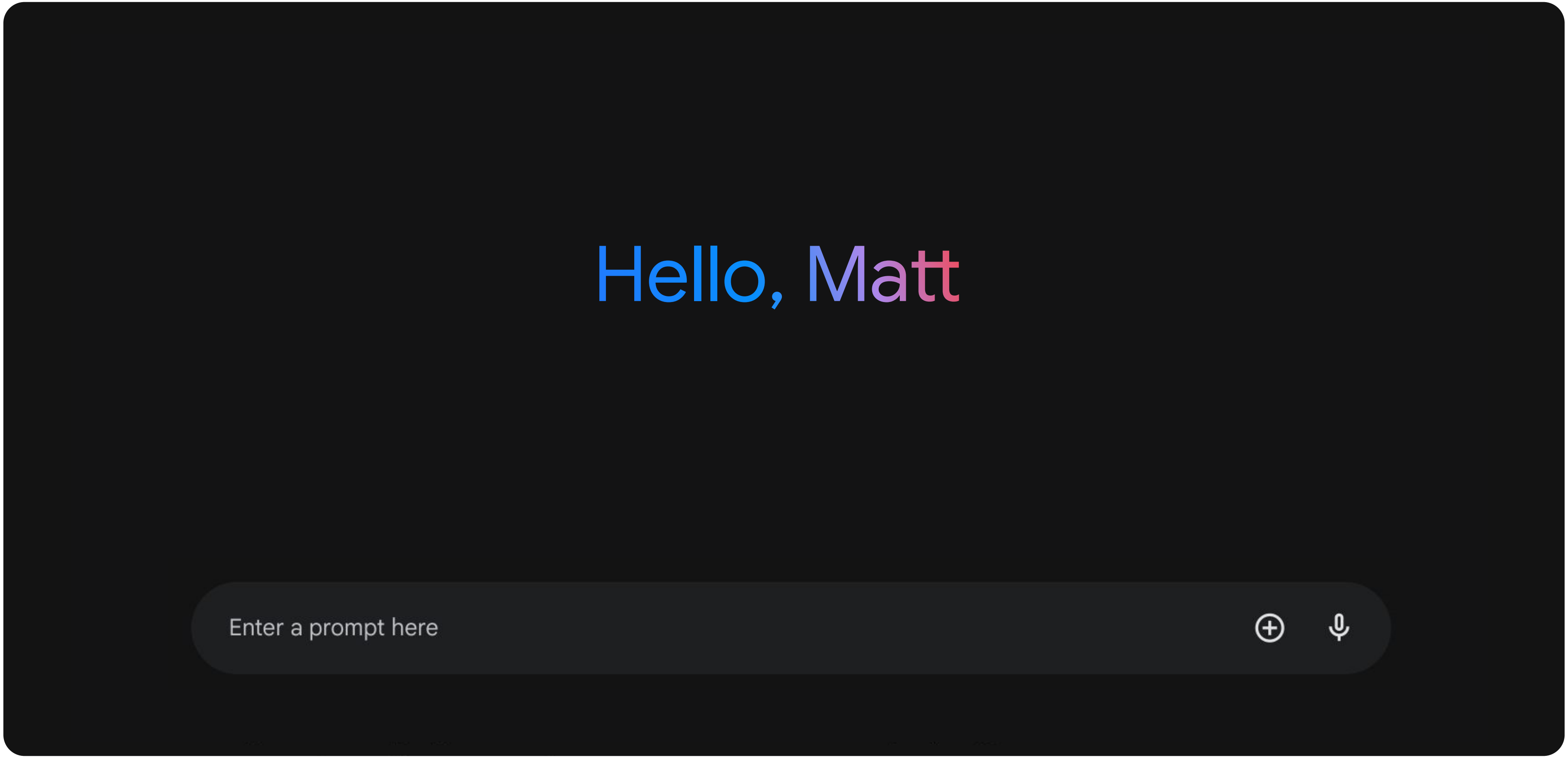
Google Cloud

Creating a Board Game Chatbot with Postgres, AI, and RAG

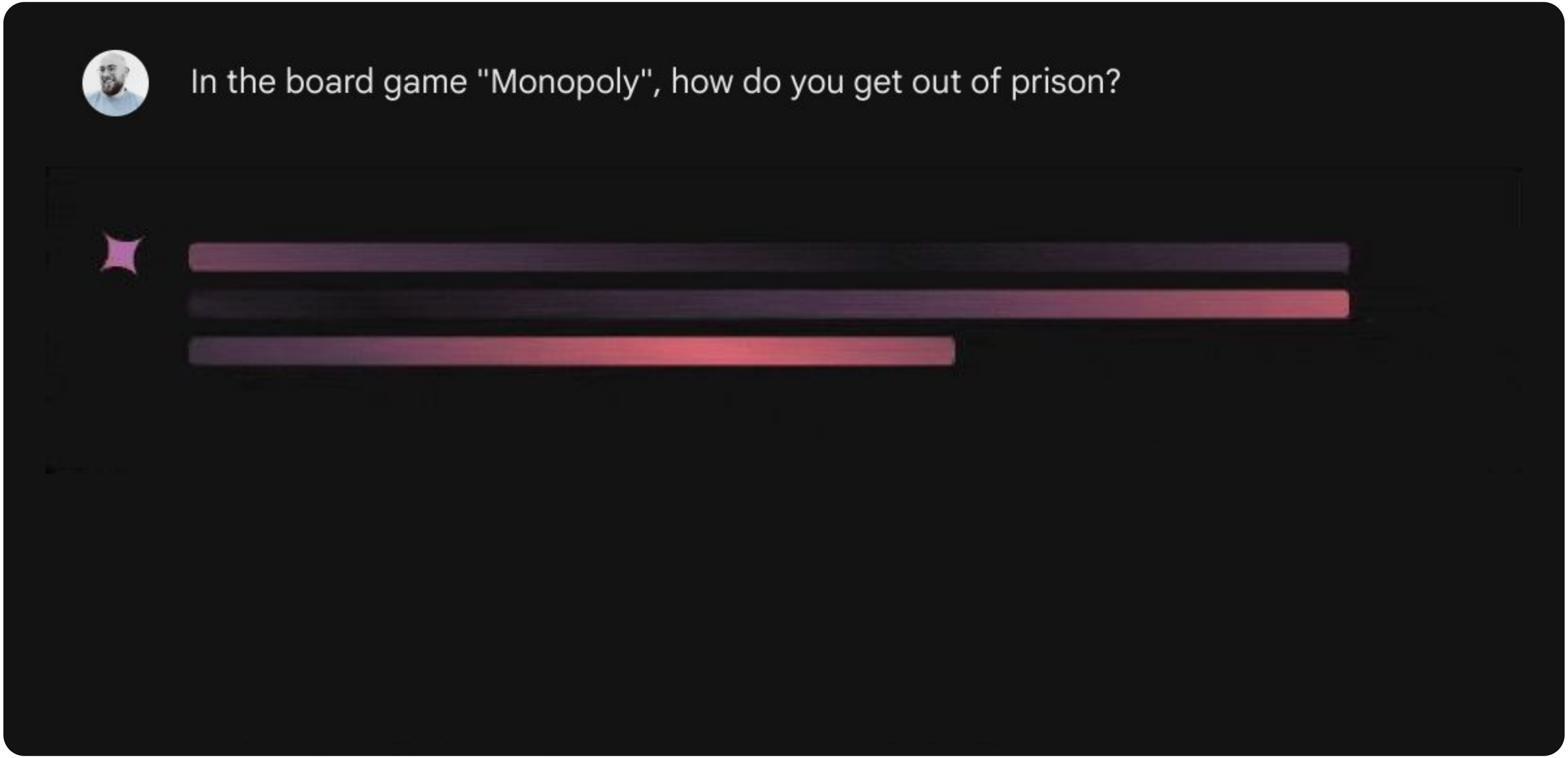
Matt Cornillon

Customer Engineer, Database specialist

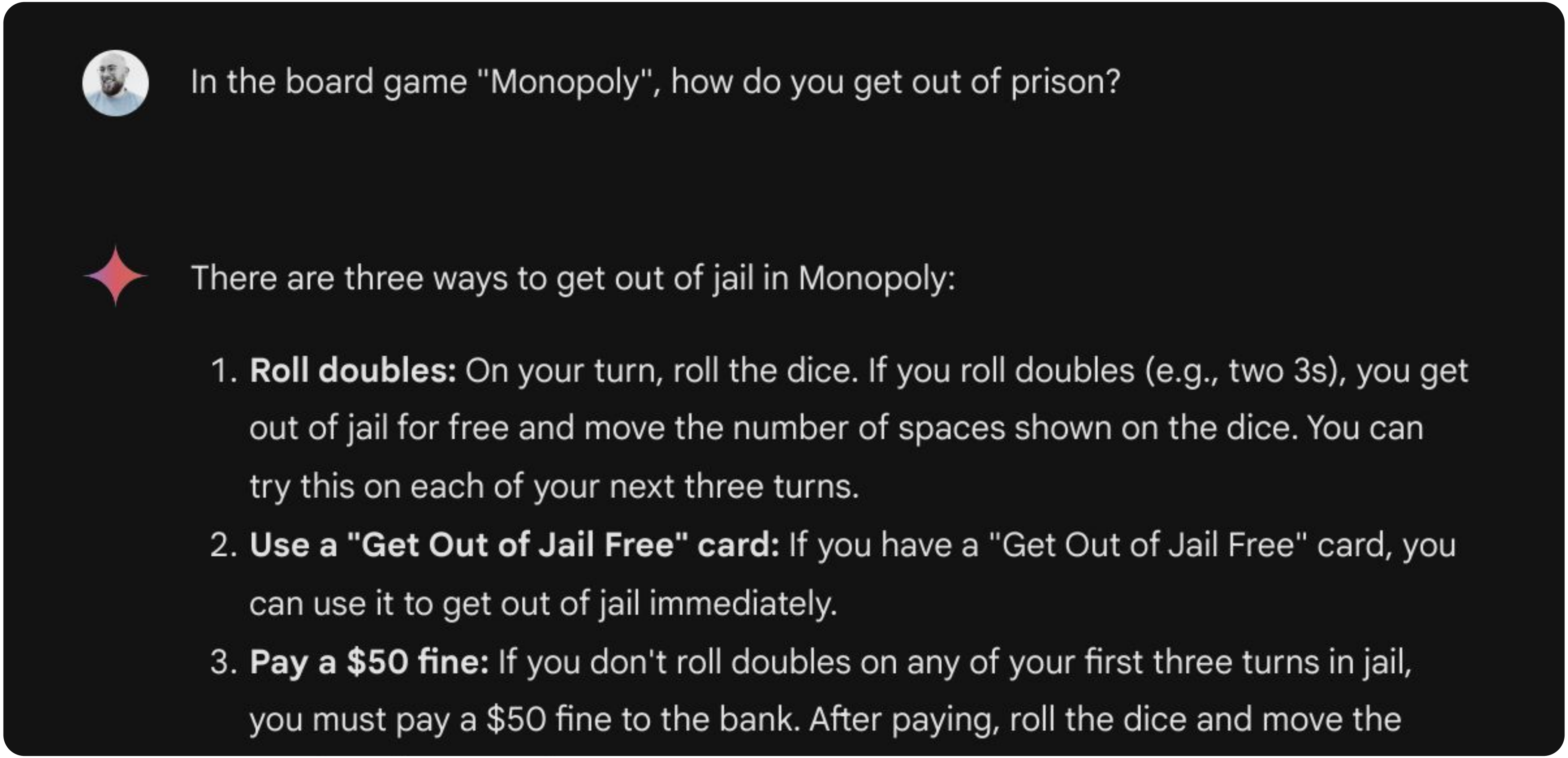
Google Cloud



Hello, Matt



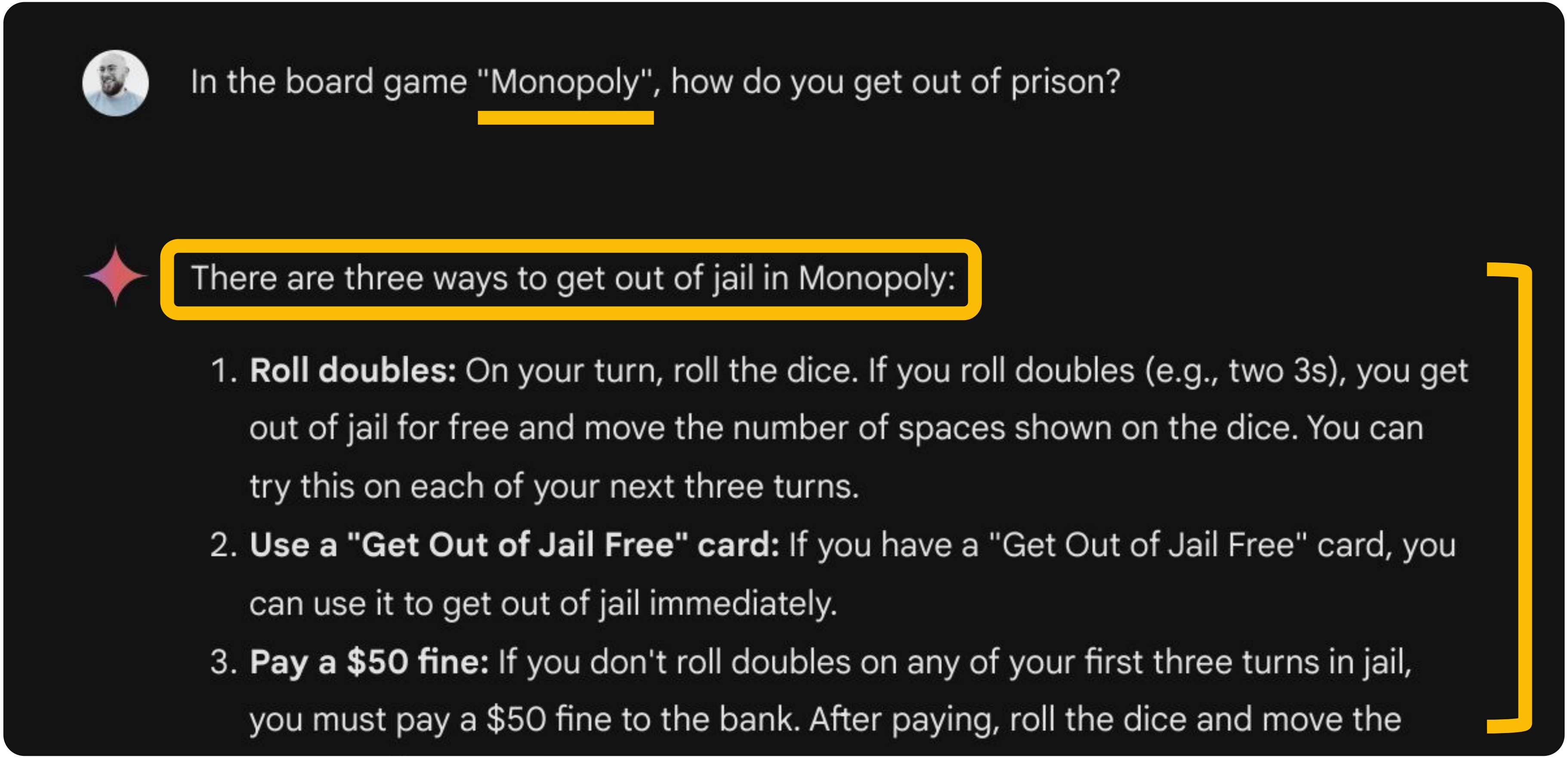
In the board game "Monopoly", how do you get out of prison?



In the board game "Monopoly", how do you get out of prison?

There are three ways to get out of jail in Monopoly:

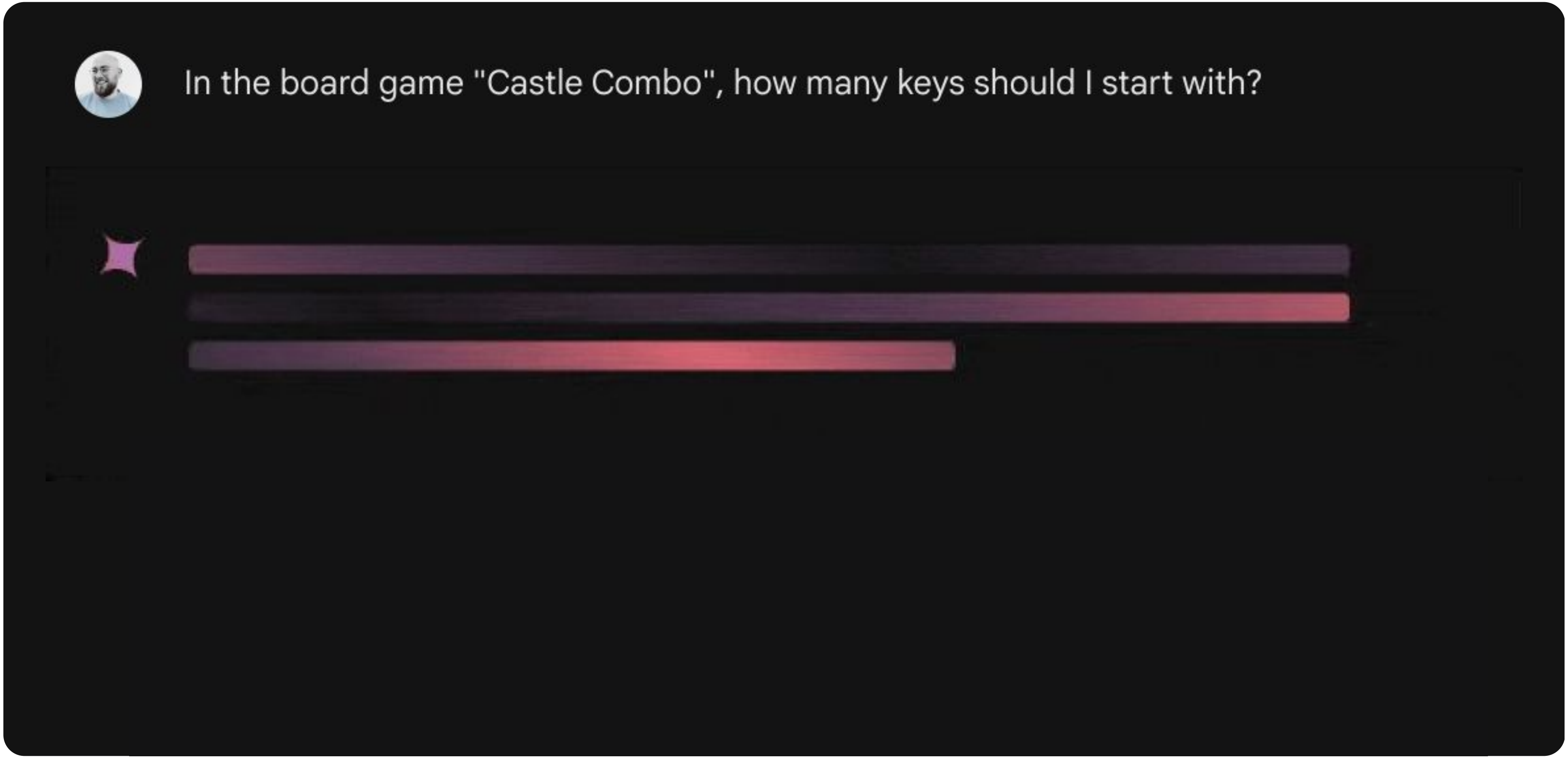
1. **Roll doubles:** On your turn, roll the dice. If you roll doubles (e.g., two 3s), you get out of jail for free and move the number of spaces shown on the dice. You can try this on each of your next three turns.
2. **Use a "Get Out of Jail Free" card:** If you have a "Get Out of Jail Free" card, you can use it to get out of jail immediately.
3. **Pay a $50 fine:** If you don't roll doubles on any of your first three turns in jail, you must pay a $50 fine to the bank. After paying, roll the dice and move the



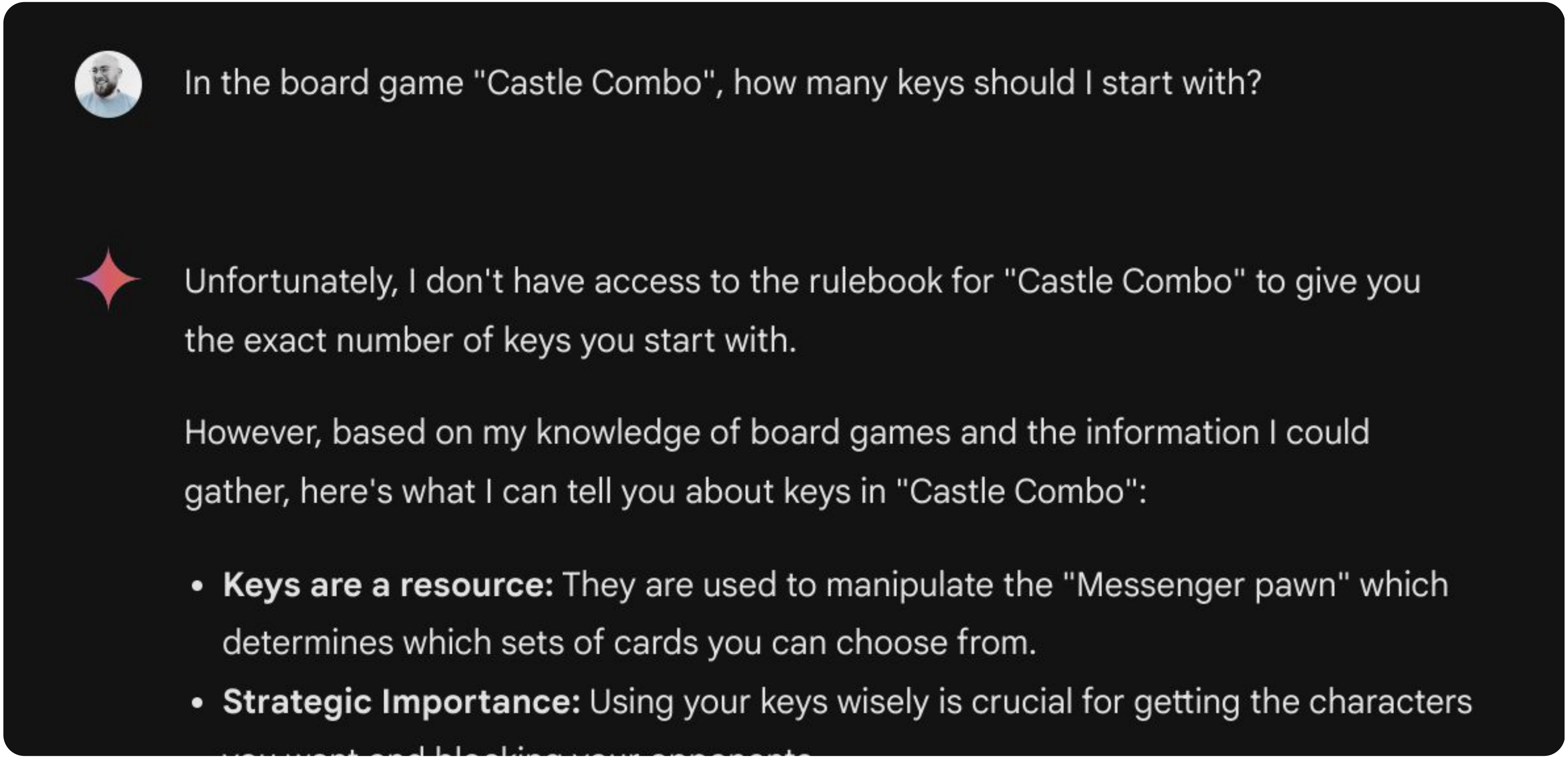
In the board game "Monopoly", how do you get out of prison?

There are three ways to get out of jail in Monopoly:

1. **Roll doubles:** On your turn, roll the dice. If you roll doubles (e.g., two 3s), you get out of jail for free and move the number of spaces shown on the dice. You can try this on each of your next three turns.
2. **Use a "Get Out of Jail Free" card:** If you have a "Get Out of Jail Free" card, you can use it to get out of jail immediately.
3. **Pay a $50 fine:** If you don't roll doubles on any of your first three turns in jail, you must pay a $50 fine to the bank. After paying, roll the dice and move the



In the board game "Castle Combo", how many keys should I start with?

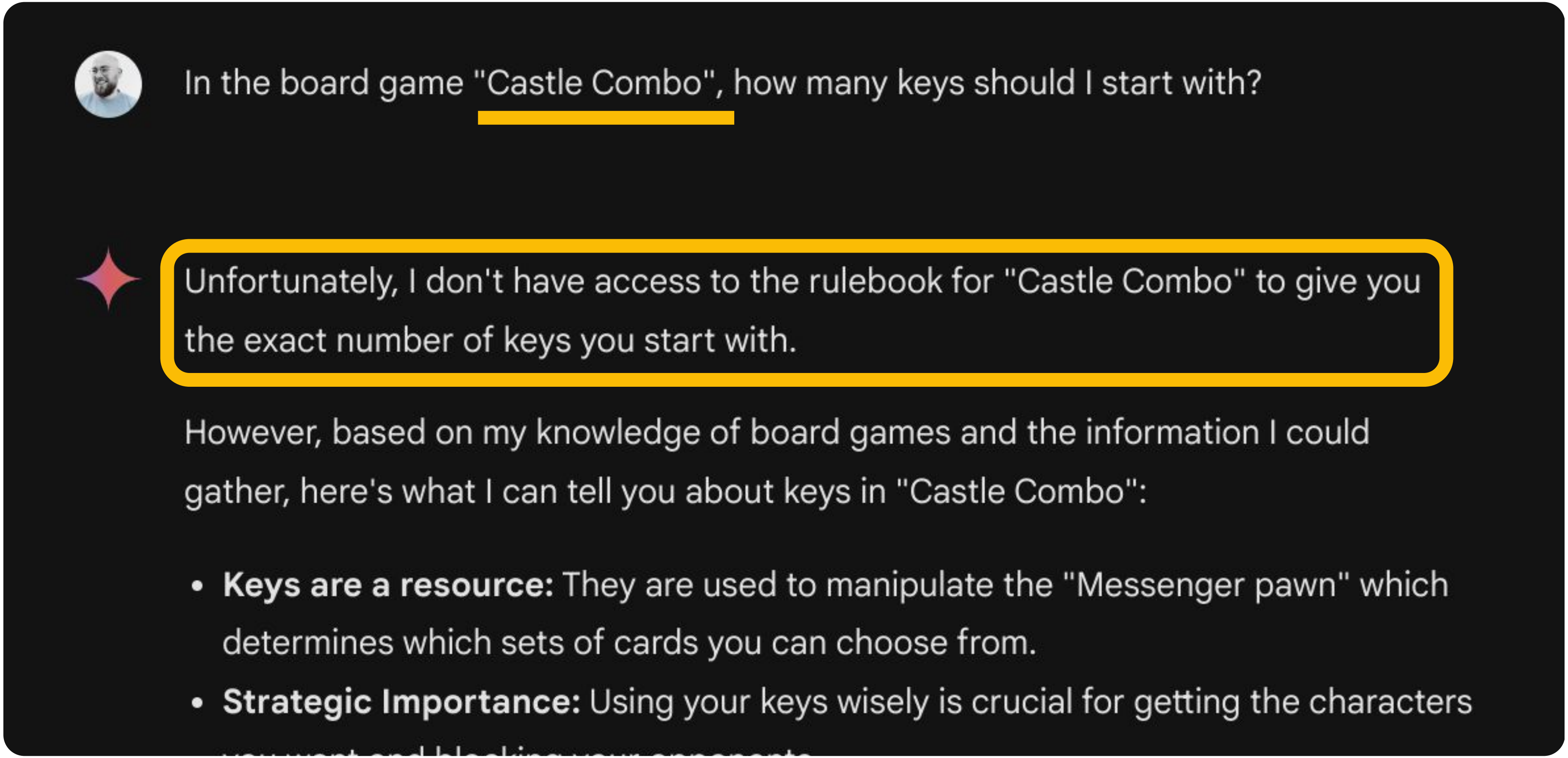


In the board game "Castle Combo", how many keys should I start with?

Unfortunately, I don't have access to the rulebook for "Castle Combo" to give you the exact number of keys you start with.

However, based on my knowledge of board games and the information I could gather, here's what I can tell you about keys in "Castle Combo":

- **Keys are a resource:** They are used to manipulate the "Messenger pawn" which determines which sets of cards you can choose from.
- **Strategic Importance:** Using your keys wisely is crucial for getting the characters

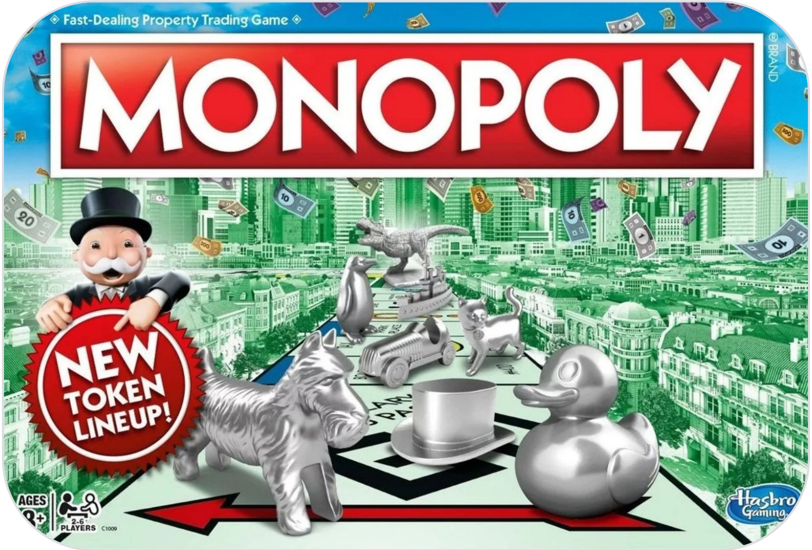


In the board game "Castle Combo", how many keys should I start with?

Unfortunately, I don't have access to the rulebook for "Castle Combo" to give you the exact number of keys you start with.

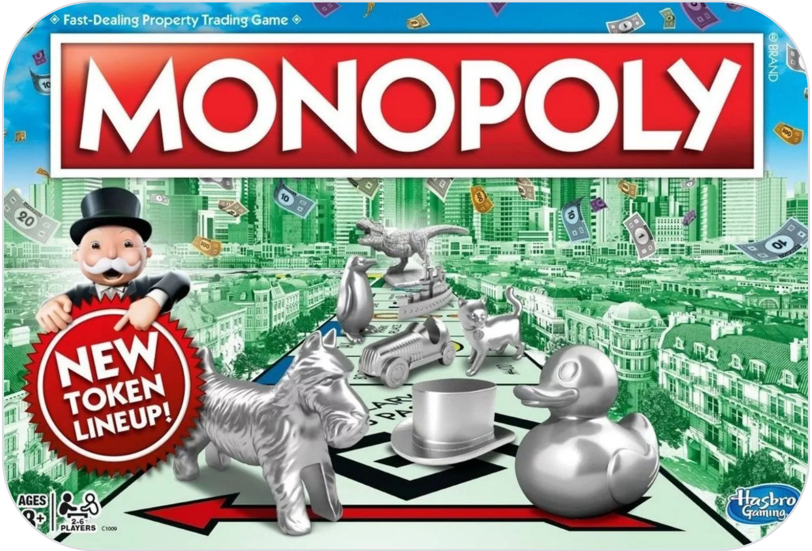
However, based on my knowledge of board games and the information I could gather, here's what I can tell you about keys in "Castle Combo":

- **Keys are a resource:** They are used to manipulate the "Messenger pawn" which determines which sets of cards you can choose from.
- **Strategic Importance:** Using your keys wisely is crucial for getting the characters


Fast-Dealing Property Trading Game
MONOPOLY
®BRAND
NEW TOKEN LINEUP!
AGES 8+
2-6 PLAYERS
C1009
Hasbro Gaming

Grégory Grard & Mathieu Roussel
Castle Combo
CATCH UP GAMES
Stéphane Escapa

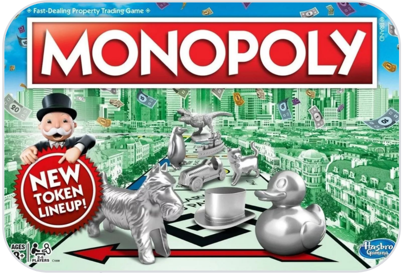
Time flies fast when you're having fun.



Release date: **1935**

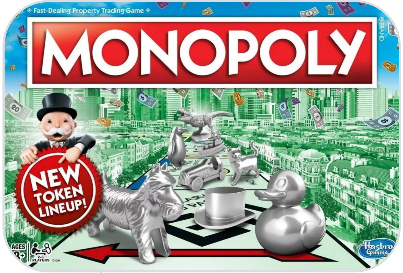
Release date: **September 2024**

Knowledge Cutoff Dates matter



1935 1995 2011 2019 September 2024

Knowledge Cutoff Dates matter



Gemini 1.5 cutoff date

The point at which Gemini's training data ends.

1935

1995

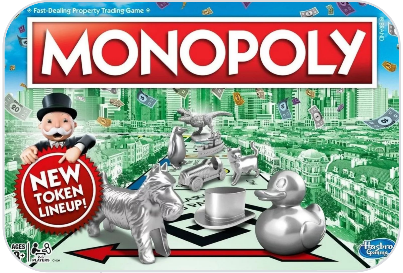
2011

2019

Late 2023

September 2024

Knowledge Cutoff Dates matter



1935

1995

2011

2019

Gemini 1.5 cutoff date

The point at which Gemini's training data ends.

Late 2023

September 2024

How to solve this?

Choose wisely

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

How to solve this?

Choose wisely

How to solve this?

Choose wisely

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

How to solve this?

Choose wisely

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros

Cons

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros **Cons**

- Can be done quickly

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros	Cons
Can be done quickly	Forces you to change your LLM and impact your existing prompts Might not always be possible (no LLM up to your desired date, technical blockers) Time cannot be stopped. You will have to do such switch continuously.

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros

Cons

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros **Cons**

- LLM tuned with your data directly to provide the most accurate answer

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros	**Cons**
LLM tuned with your data directly to provide the most accurate answer	Can be computationally expensive, especially for large-scale fine-tuning Requires a significant amount of high-quality data for effective fine-tuning.

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

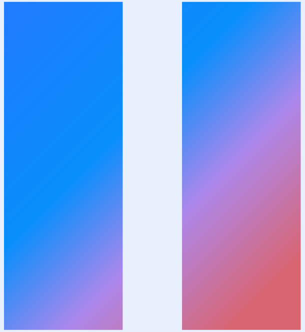
Because "No one can serve you better than yourself", create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely



Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

RAG?

Use a more up-to-date LLM

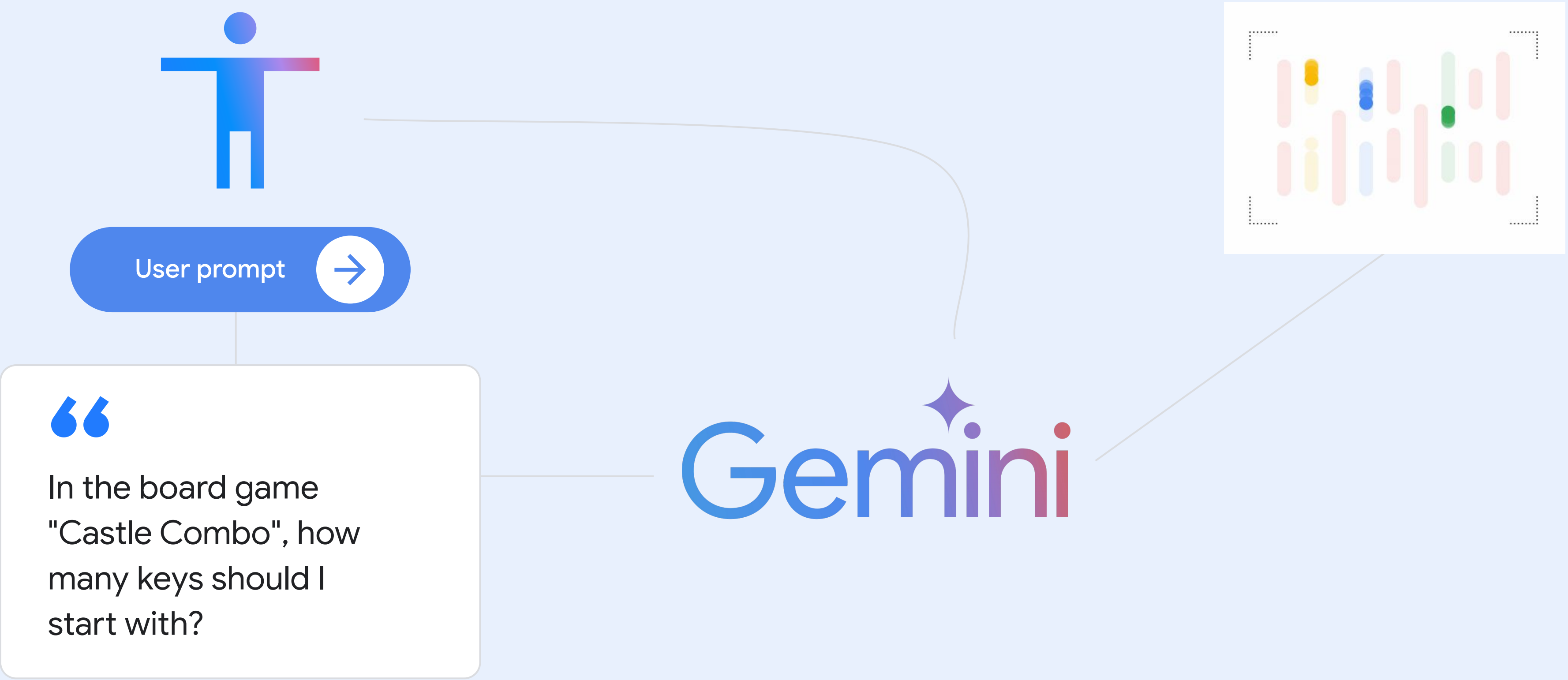
LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

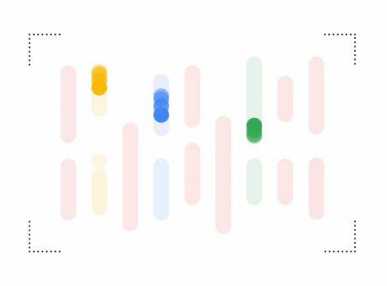
Combine your favorite and powerful LLM with your own data to always be up to date.



User prompt

> In the board game "Castle Combo", how many keys should I start with?

Gemini

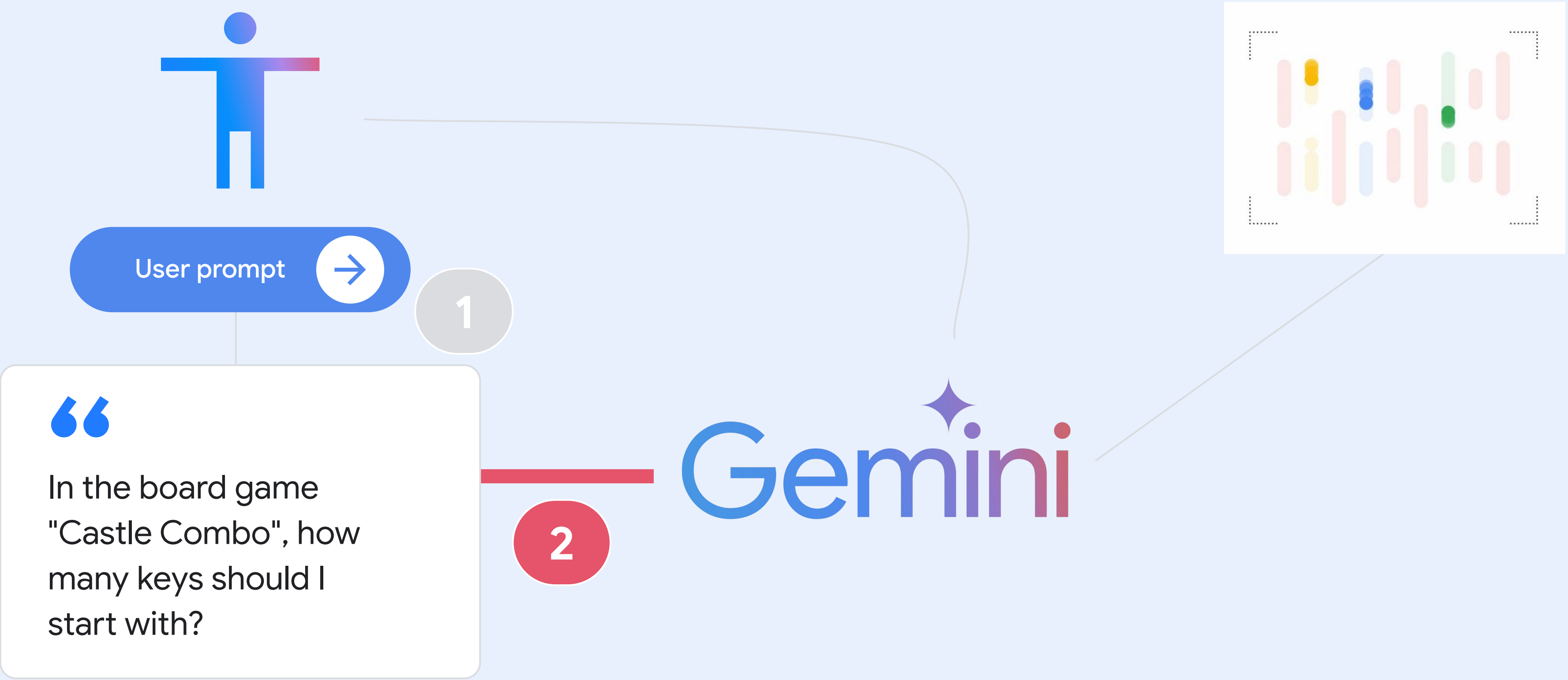


User prompt

1

> In the board game "Castle Combo", how many keys should I start with?

Gemini



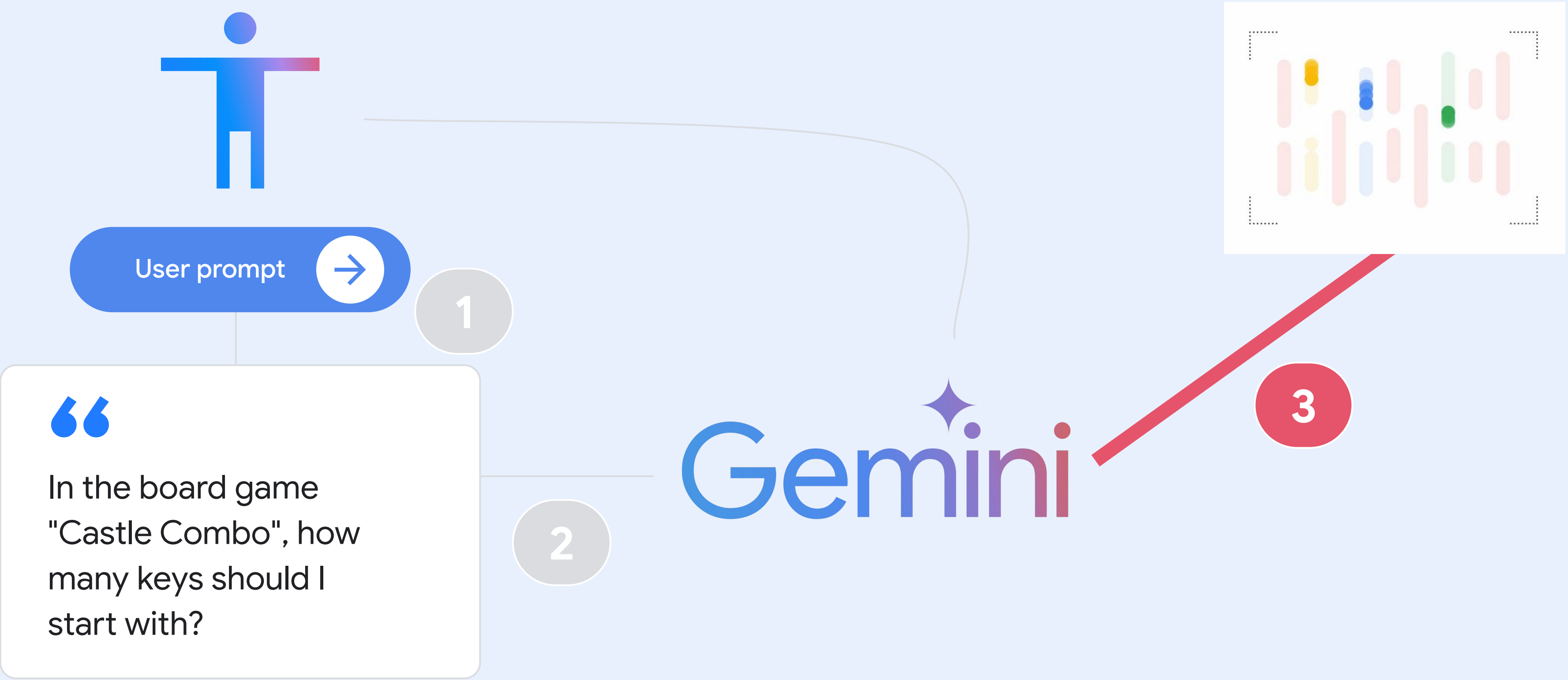
User prompt

1

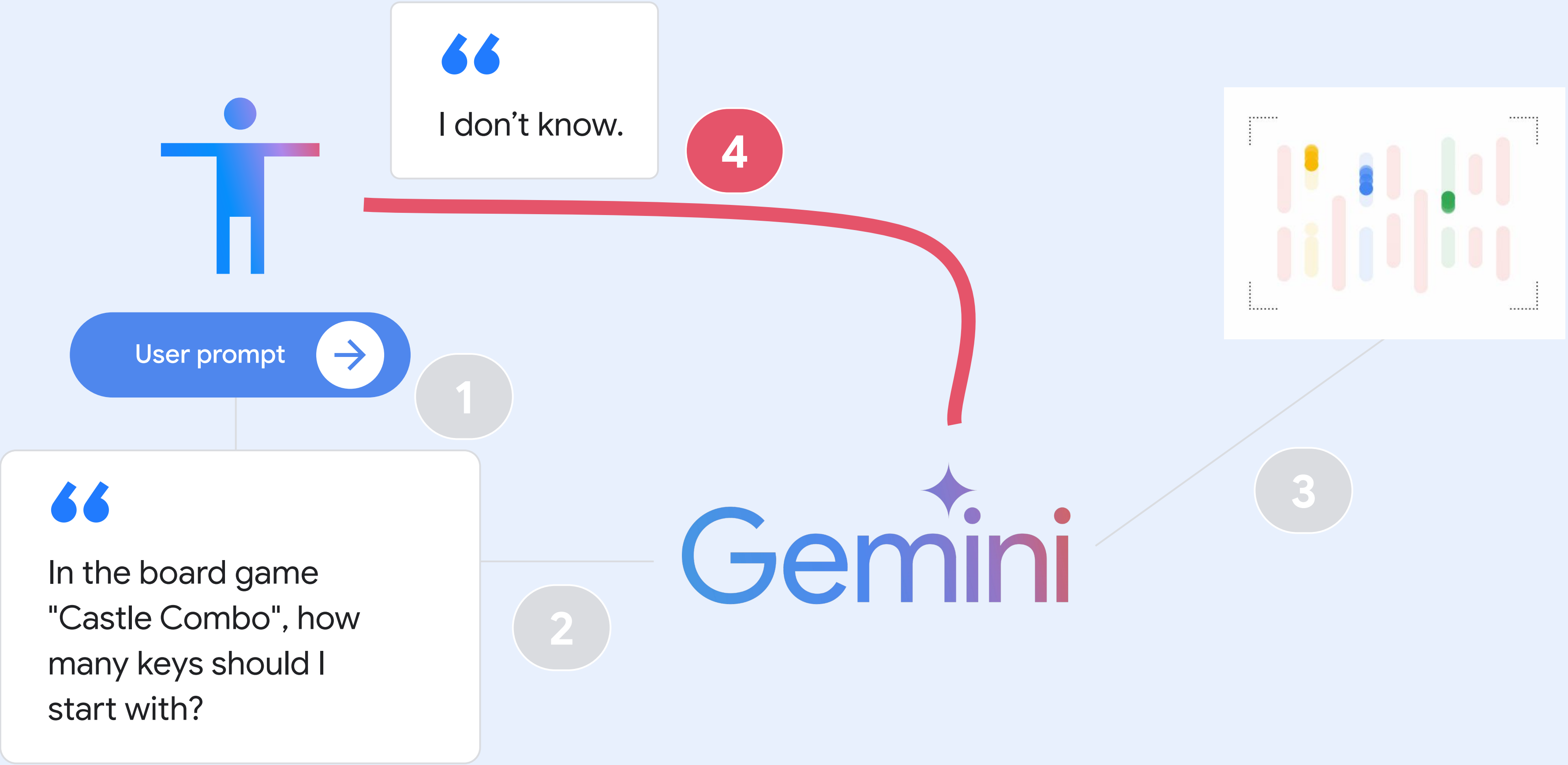
2

Gemini

> In the board game "Castle Combo", how many keys should I start with?



In the board game "Castle Combo", how many keys should I start with?



> I don't know.

> In the board game "Castle Combo", how many keys should I start with?

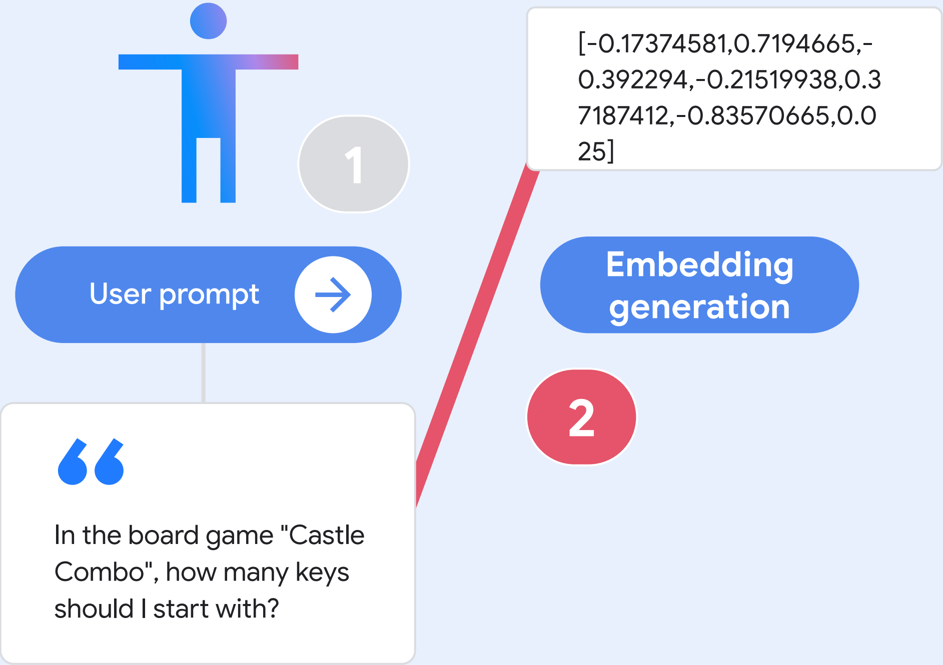
User prompt

> In the board game "Castle Combo", how many keys should I start with?

1

User prompt

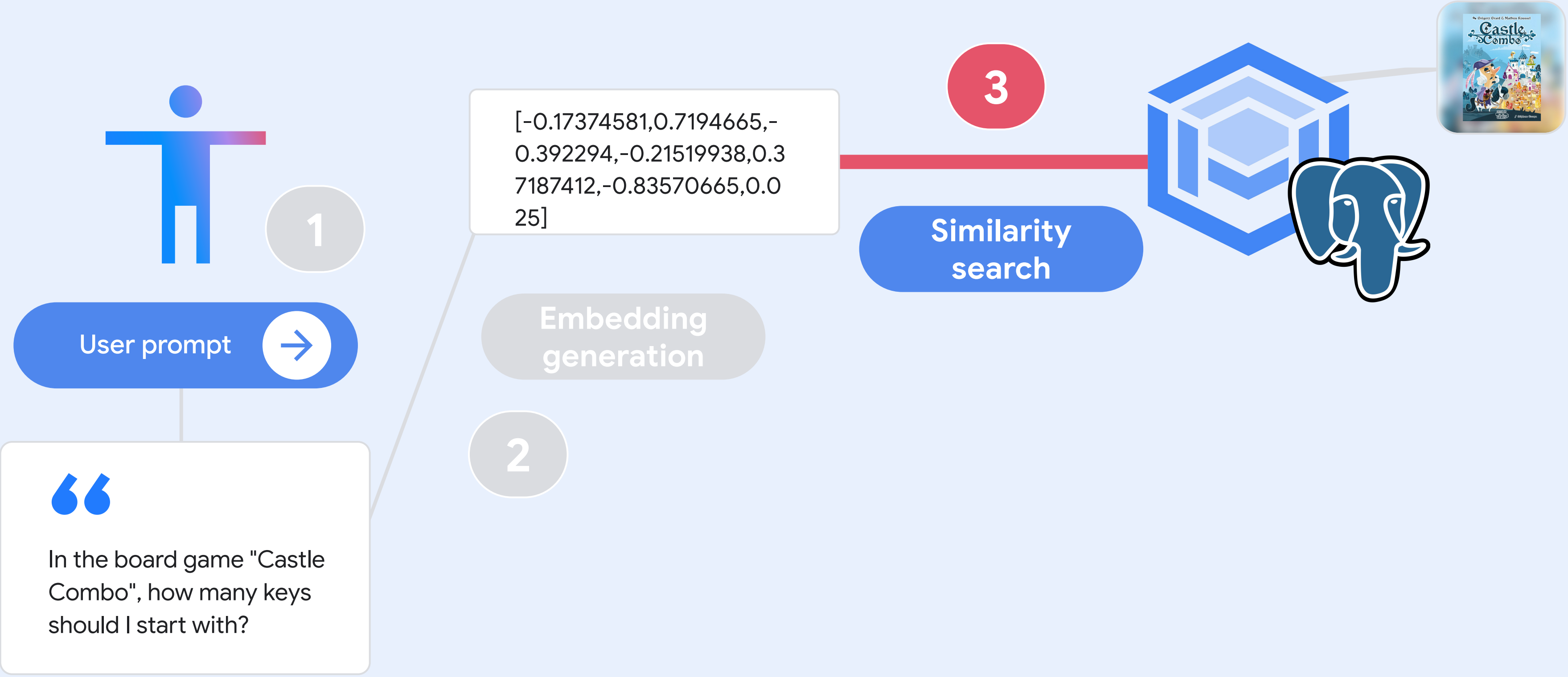
> In the board game "Castle Combo", how many keys should I start with?



> In the board game "Castle Combo", how many keys should I start with?

[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025]

Gemini



User prompt

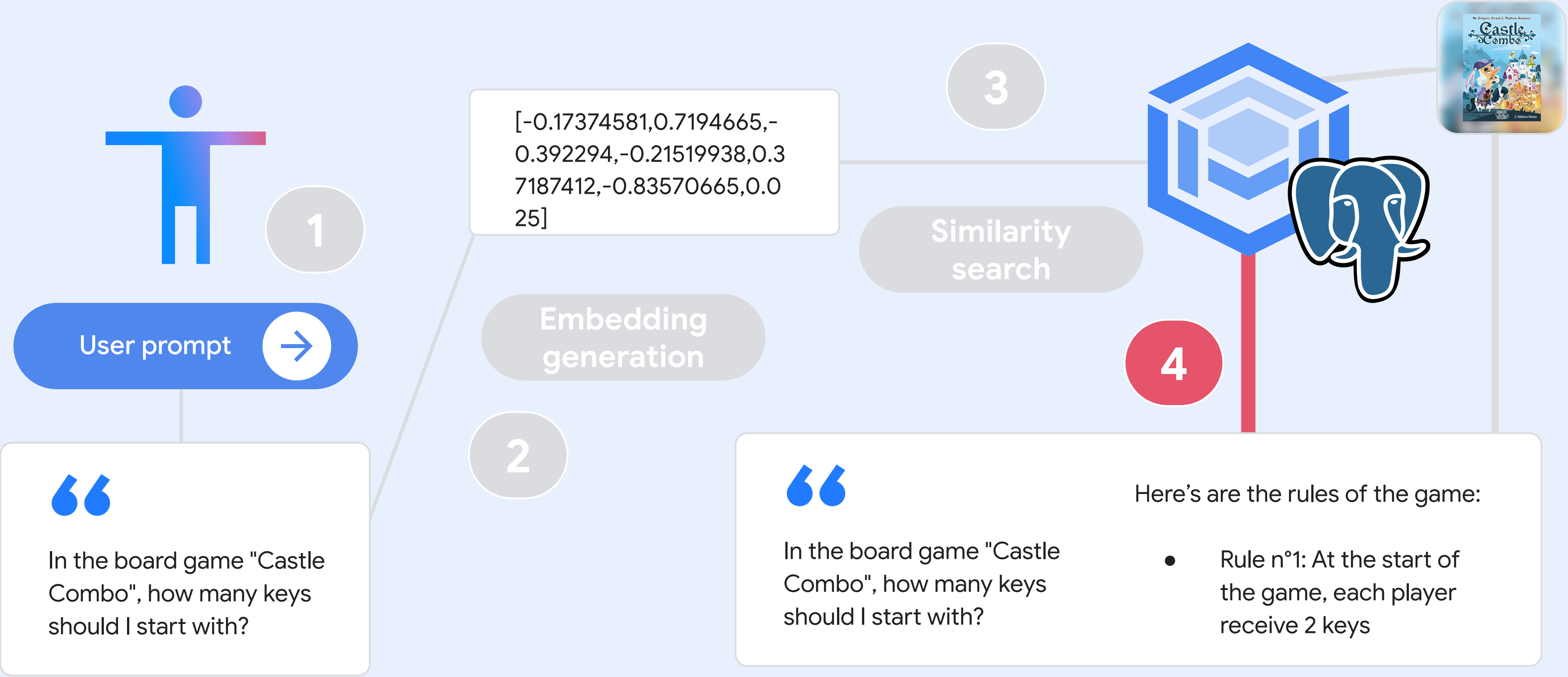
> In the board game "Castle Combo", how many keys should I start with?

Embedding generation

[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025]

Similarity search

Gemini



1

User prompt

In the board game "Castle Combo", how many keys should I start with?

2

Embedding generation

[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025]

3

Similarity search

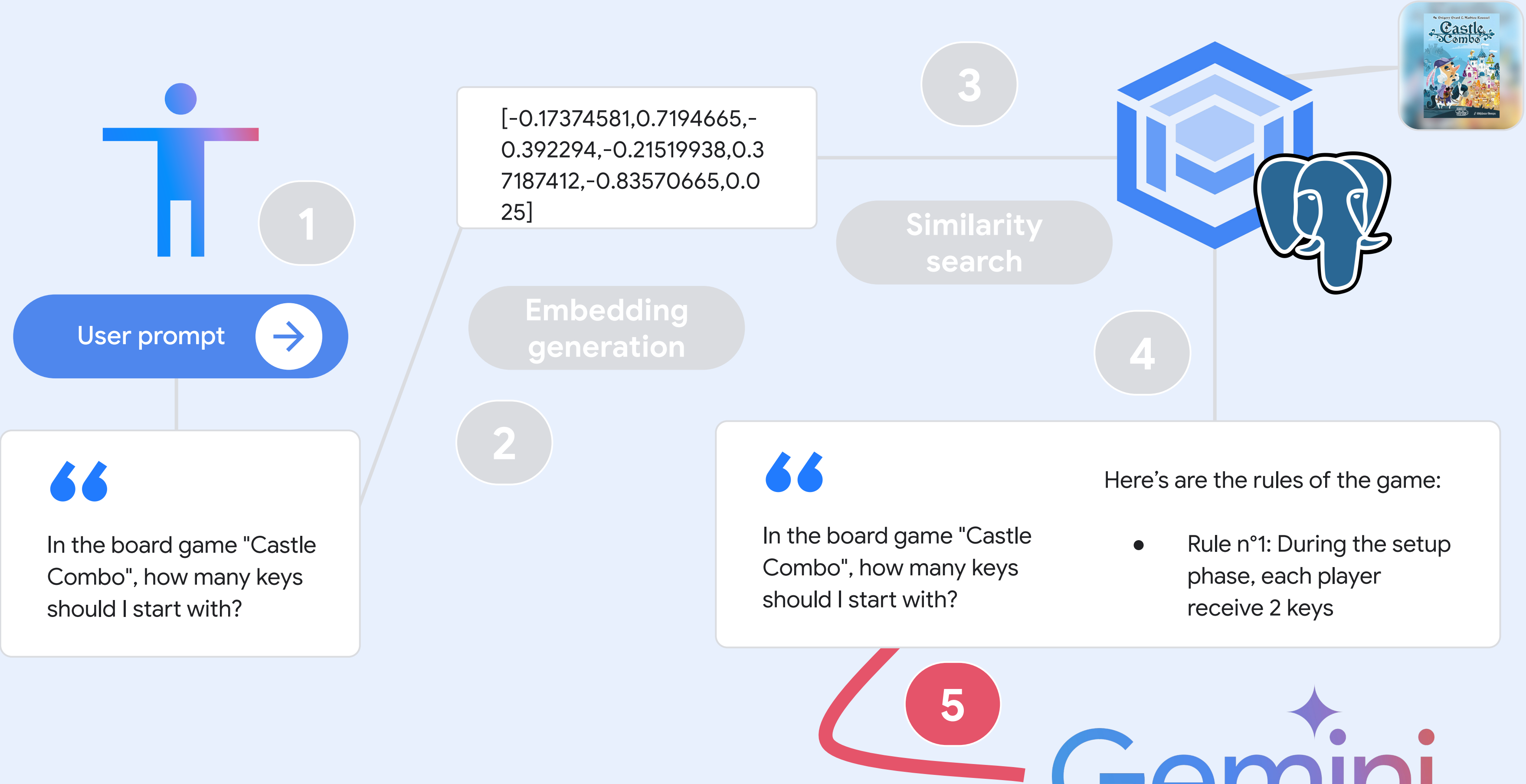
4

In the board game "Castle Combo", how many keys should I start with?

Here’s are the rules of the game:

- Rule n°1: At the start of the game, each player receive 2 keys

Gemini



1

User prompt

“In the board game "Castle Combo", how many keys should I start with?

2

Embedding generation

[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025]

3

Similarity search

4

“In the board game "Castle Combo", how many keys should I start with?

Here’s are the rules of the game:

- Rule n°1: During the setup phase, each player receive 2 keys

5

Gemini

User prompt

In the board game "Castle Combo", how many keys should I start with?

6

Gemini

In Castle combo, each player start the game with 2 keys.
See Rule n°1: (...)

How to solve this?

Choose wisely

Pros **Cons**

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because "No one can serve you better than yourself", create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros **Cons**

- Flexibility: Can be easily adapted to different knowledge base.
- Real time updates
- Easy to implement on existing and evolving data

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Pros	Cons
Flexibility: Can be easily adapted to different knowledge base. Real time updates Easy to implement on existing and evolving data	Your data needs to be clean and qualitative

Use a more up-to-date LLM

LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation

Combine your favorite and powerful LLM with your own data to always be up to date.

How to solve this?

Choose wisely

Use a more up-to-date LLM

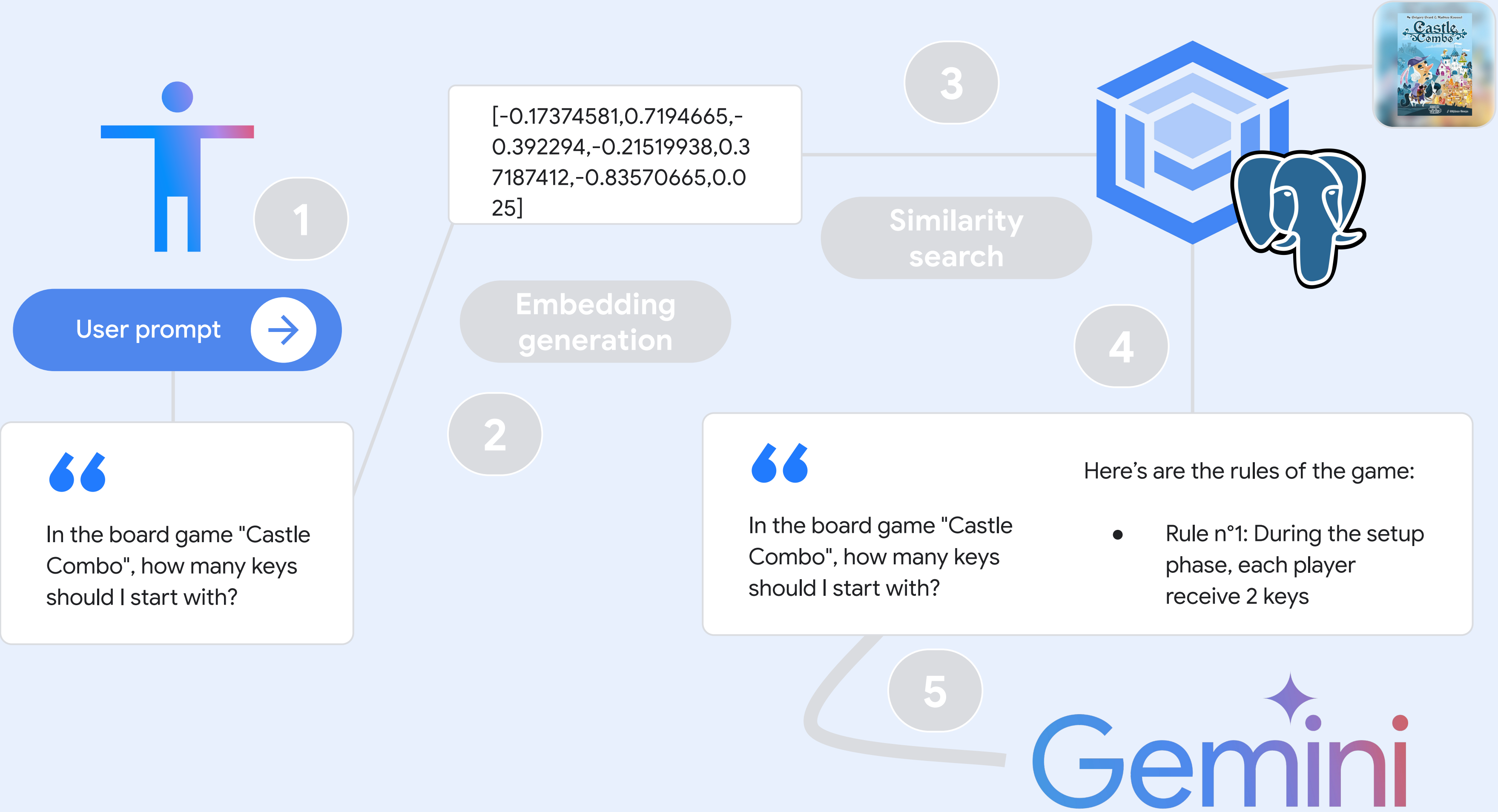
LLMs are updated continuously. Maybe you could find one that has the information you require?

Finetune your model on specific data

Because “No one can serve you better than yourself”, create the LLM you really dream of.

Implement Retrieval Augmented Generation ✓

Combine your favorite and powerful LLM with your own data to always be up to date.



1

User prompt

> In the board game "Castle Combo", how many keys should I start with?

2

Embedding generation

[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025]

3

Similarity search

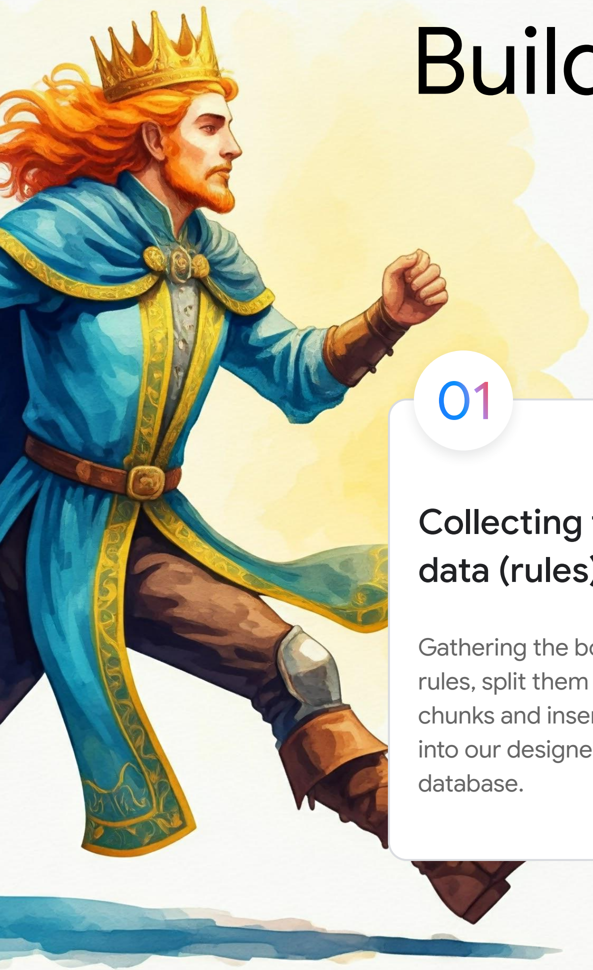
4

> In the board game "Castle Combo", how many keys should I start with?

Here’s are the rules of the game:

- Rule n°1: During the setup phase, each player receive 2 keys

5



Building a board game chatbot with Postgres and RAG

01

Collecting the data (rules)

Gathering the board game rules, split them into chunks and insert them into our designed Postgres database.

02

Generating the embeddings of the rules

Transforming those texts into usable vectors for similarity searches.

03

Finding the rules using similarity searches

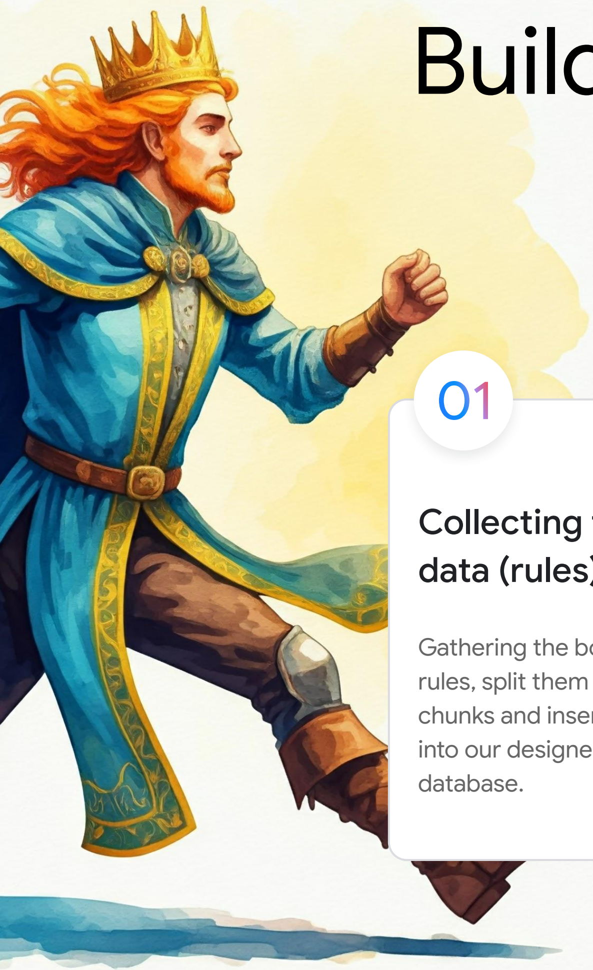
Explore similarity search with Postgres and pgvector to find the most appropriate rules.

04

Prompt engineering with input and rules

Building a new prompt with the user question and the rules we found to feed the LLM.

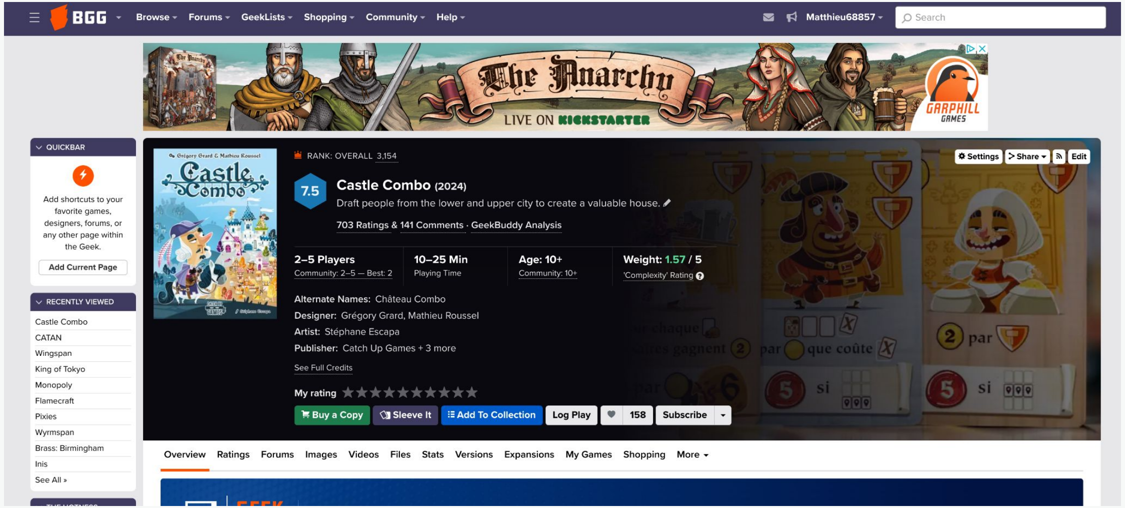
Building a board game chatbot with Postgres and RAG



01

Collecting the data (rules)

Gathering the board game rules, split them into chunks and insert them into our designed Postgres database.



RANK: OVERALL 3,154

7.5 **Castle Combo** (2024)

Draft people from the lower and upper city to create a valuable house.

703 Ratings & 141 Comments · GeekBuddy Analysis

2–5 Players Community: 2–5 — Best: 2

10–25 Min Playing Time

Age: 10+ Community: 10+

Weight: 1.57 / 5 'Complexity' Rating

Alternate Names: Château Combo

Designer: Grégory Grard, Mathieu Roussel

Artist: Stéphane Escapa

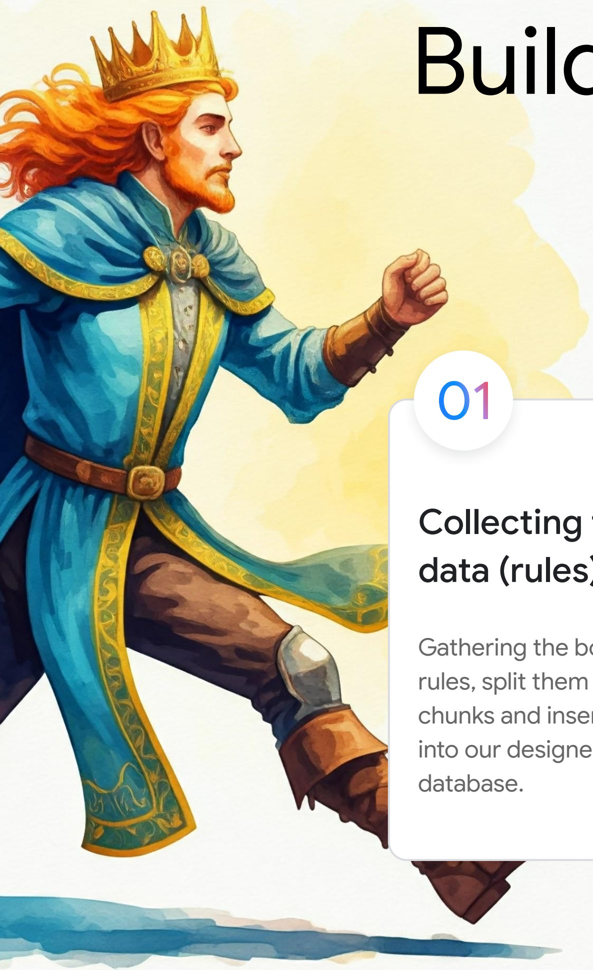
Publisher: Catch Up Games + 3 more

See Full Credits

My rating

https://boardgamegeek.com/

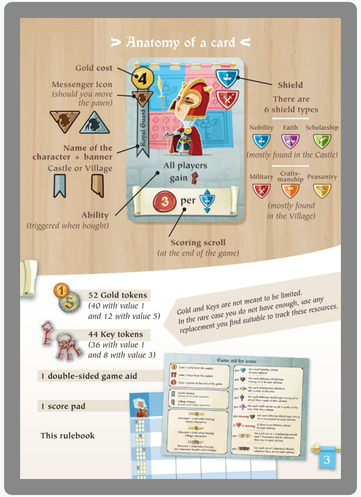
Building a board game chatbot with Postgres and RAG



01

Collecting the data (rules)

Gathering the board game rules, split them into chunks and insert them into our designed Postgres database.



> Anatomy of a card <

Gold cost

Messenger Icon (should you move the pawn)

Name of the character + banner

Castle or Village

Ability (triggered when bought)

All players gain

per

Scoring scroll (at the end of the game)

Shield

There are 6 shield types

Nobility Faith Scholarship (mostly found in the Castle)

Military Crafts-manship Peasantry (mostly found in the Village)

52 Gold tokens (40 with value 1 and 12 with value 5)

44 Key tokens (36 with value 1 and 8 with value 3)

Gold and Keys are not meant to be limited. In the rare case you do not have enough, use any replacement you find suitable to track these resources.

1 double-sided game aid

1 score pad

This rulebook

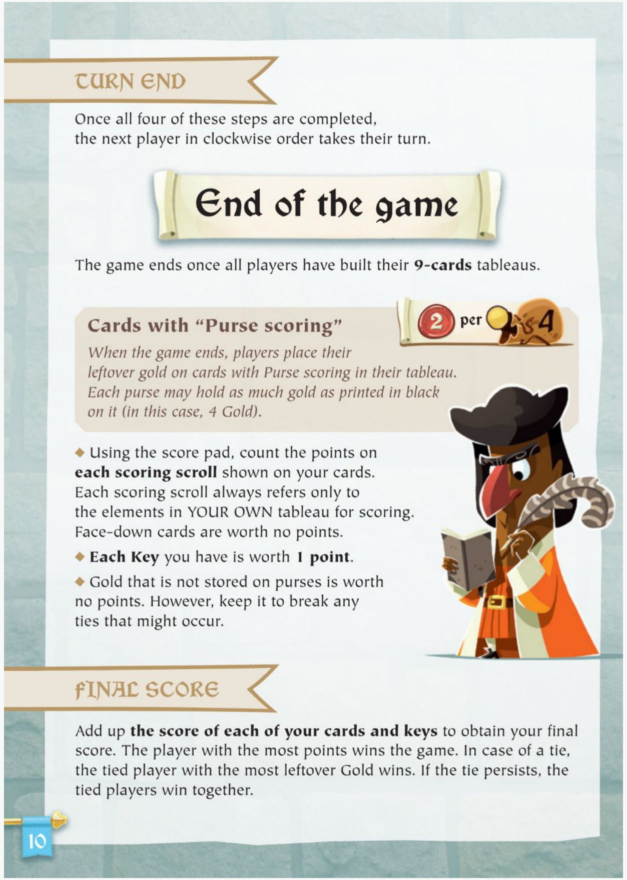
Game aid for icons

3

pdf

- Extracting texts from PDF files
- Splitting texts into chunks
- Inserting chunks into Postgres

Extracting texts from PDF files



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

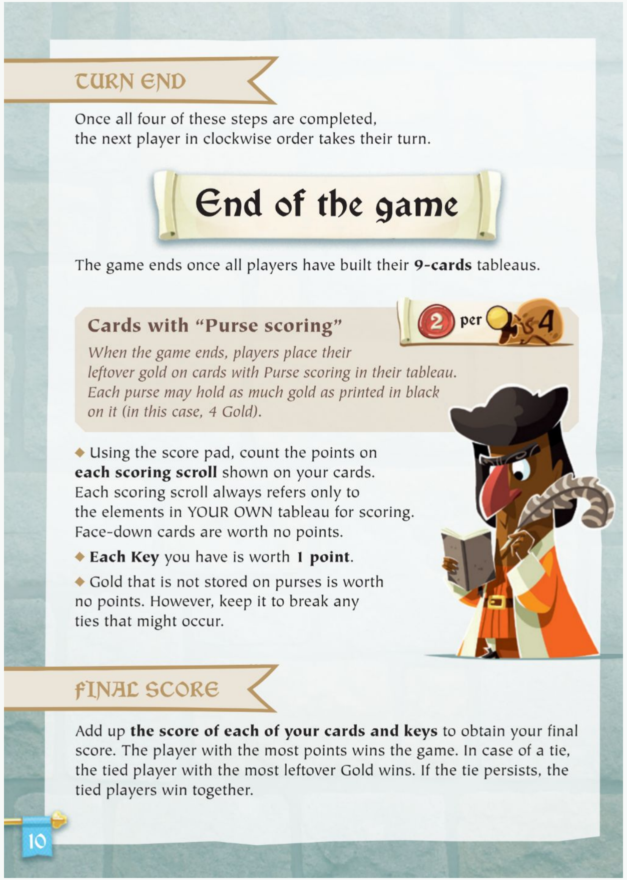
- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

pdf

Extracting texts from PDF files



TURN END

Once all four of these steps are completed, the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

◆ Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.

◆ **Each Key** you have is worth **1 point**.

◆ Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

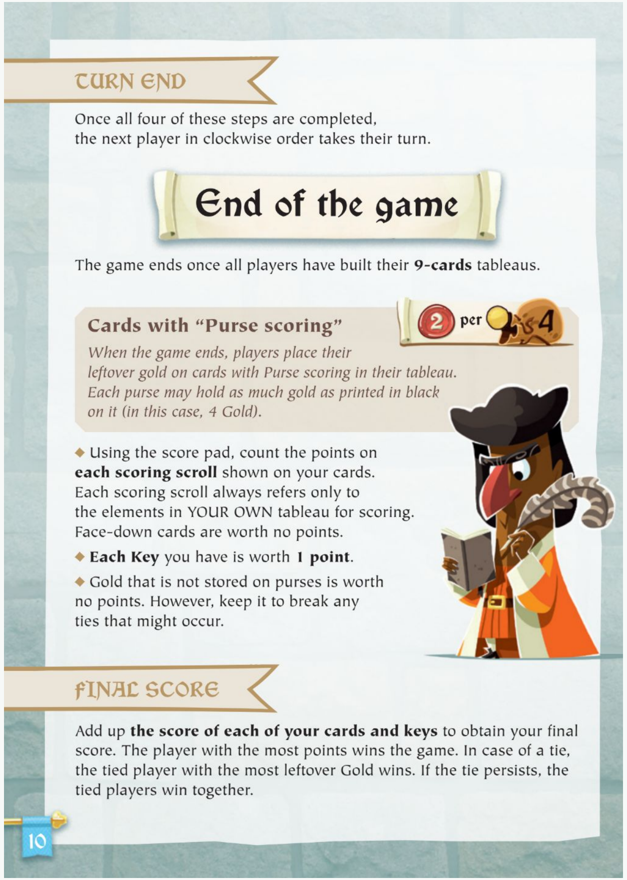
10

pdf

```
from PyPDF2 import PdfReader

reader =
PdfReader("castlecombo_rules.pdf")
page = reader.pages[9]
print(page.extract_text())
```

PyPDF2

Extracting texts from PDF files



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with “Purse scoring”

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

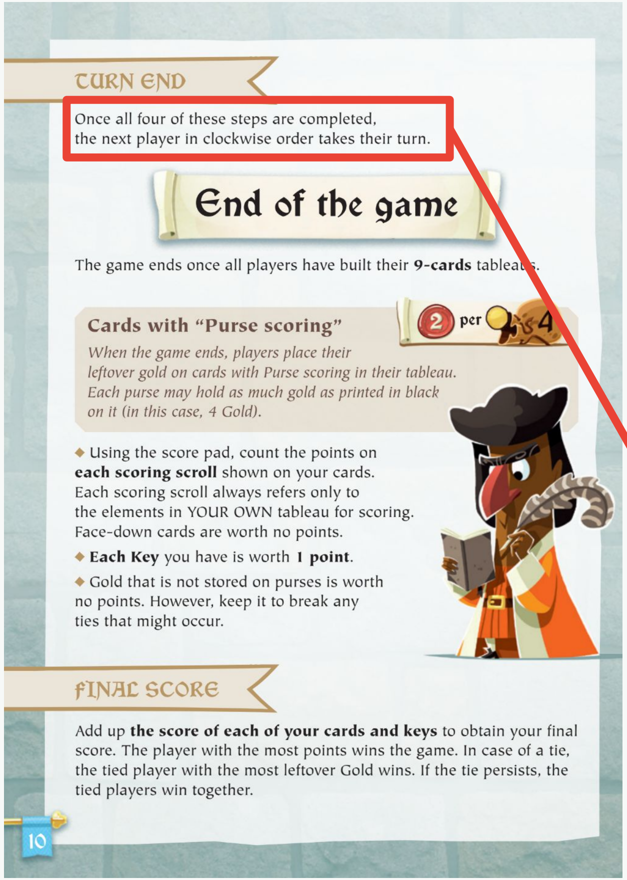
10

pdf

```
10 Using the score pad, count the points on
each scoring scroll  shown on your cards.
Each scoring scroll always refers only to
the elements in YOUR OWN tableau for scoring.
Face-down cards are worth no points.
 Each Key  you have is worth 1 point .
 Gold that is not stored on purses is worth
no points. However, keep it to break any
ties that might occur.
End of the game
The game ends once all players have built their 9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in their tableau.
Each purse may hold as much gold as printed in black
on it (in this case, 4 Gold).
per
Once all four of these steps are completed,
the next player in clockwise order takes their turn.TURN END
FINAL SCORE
Add up  the score of each of your cards and keys  to obtain your final
score. The player with the most points wins the game. In case of a tie,
the tied player with the most leftover Gold wins. If the tie persists, the
tied players win together.
```

Results from PyPDF2

Extracting texts from PDF files



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

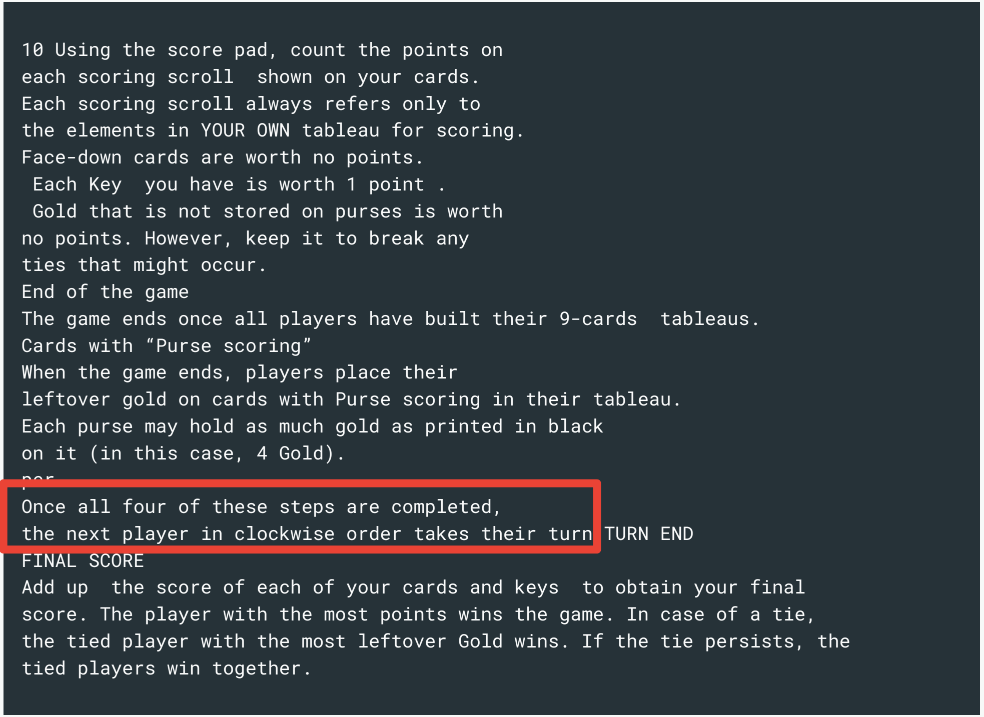
When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

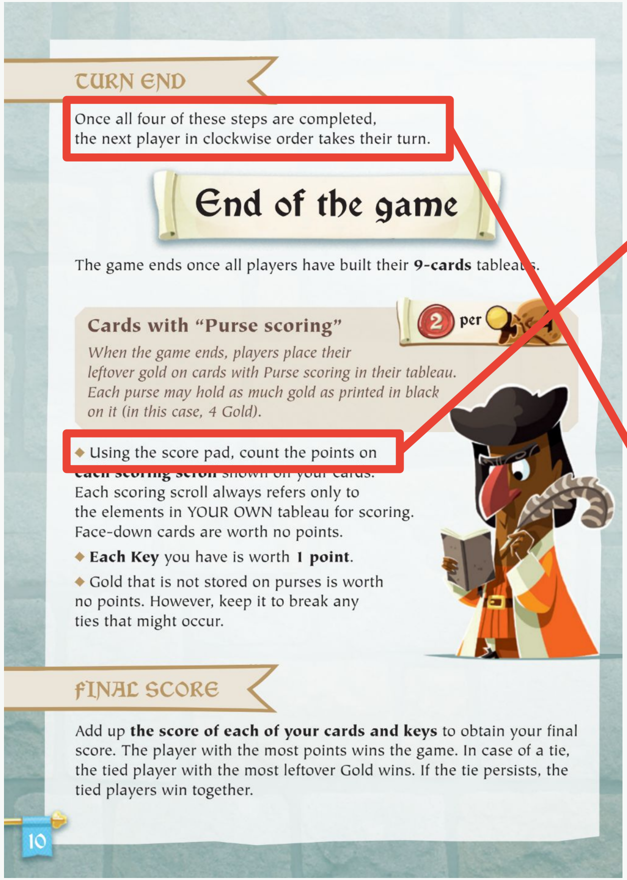
pdf



```
10 Using the score pad, count the points on
each scoring scroll  shown on your cards.
Each scoring scroll always refers only to
the elements in YOUR OWN tableau for scoring.
Face-down cards are worth no points.
 Each Key  you have is worth 1 point .
 Gold that is not stored on purses is worth
no points. However, keep it to break any
ties that might occur.
End of the game
The game ends once all players have built their 9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in their tableau.
Each purse may hold as much gold as printed in black
on it (in this case, 4 Gold).
per
Once all four of these steps are completed,
the next player in clockwise order takes their turn TURN END
FINAL SCORE
Add up  the score of each of your cards and keys  to obtain your final
score. The player with the most points wins the game. In case of a tie,
the tied player with the most leftover Gold wins. If the tie persists, the
tied players win together.
```

Results from PyPDF2

Extracting texts from PDF files



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

2 per

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

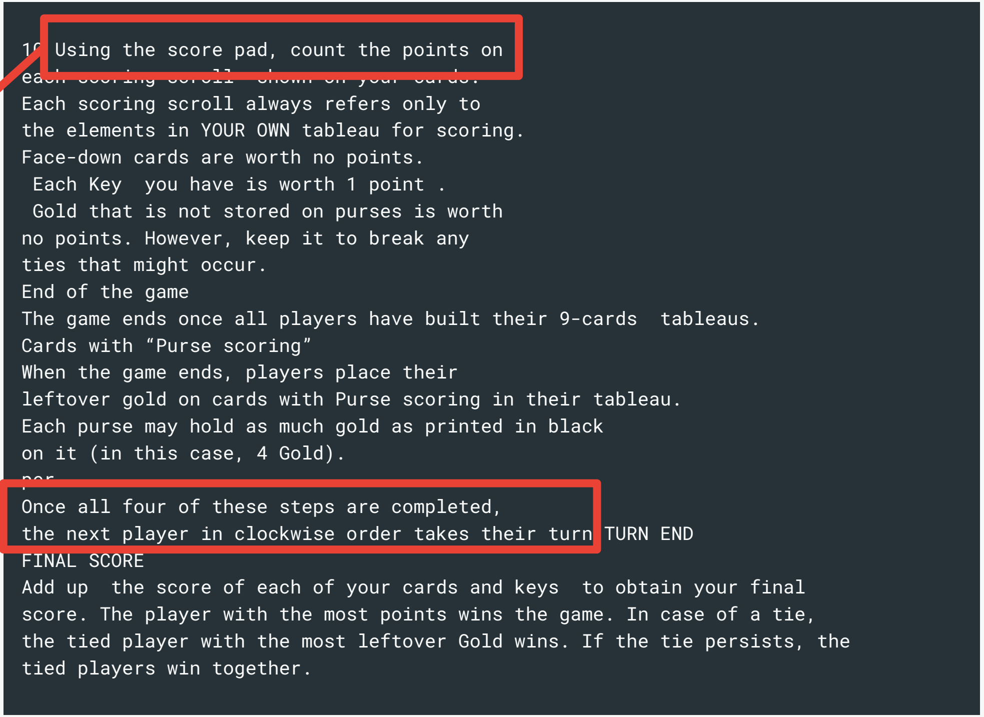
- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

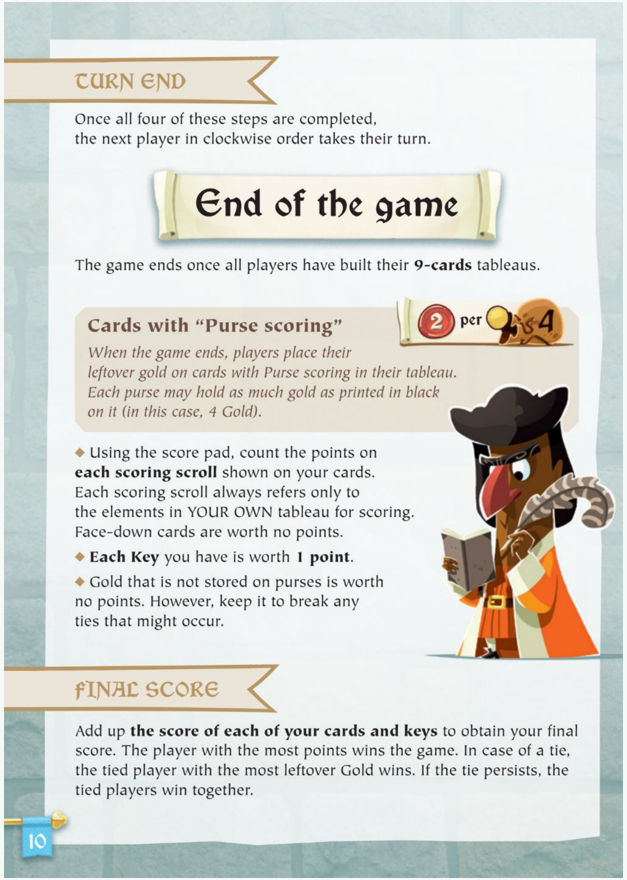
pdf



```
10 Using the score pad, count the points on
each scoring scroll  shown on your cards.
Each scoring scroll always refers only to
the elements in YOUR OWN tableau for scoring.
Face-down cards are worth no points.
 Each Key  you have is worth 1 point .
 Gold that is not stored on purses is worth
no points. However, keep it to break any
ties that might occur.
End of the game
The game ends once all players have built their 9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in their tableau.
Each purse may hold as much gold as printed in black
on it (in this case, 4 Gold).
per
Once all four of these steps are completed,
the next player in clockwise order takes their turn TURN END
FINAL SCORE
Add up  the score of each of your cards and keys  to obtain your final
score. The player with the most points wins the game. In case of a tie,
the tied player with the most leftover Gold wins. If the tie persists, the
tied players win together.
```

Results from PyPDF2

Extracting texts from PDF files



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with “Purse scoring”

2 per 4

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

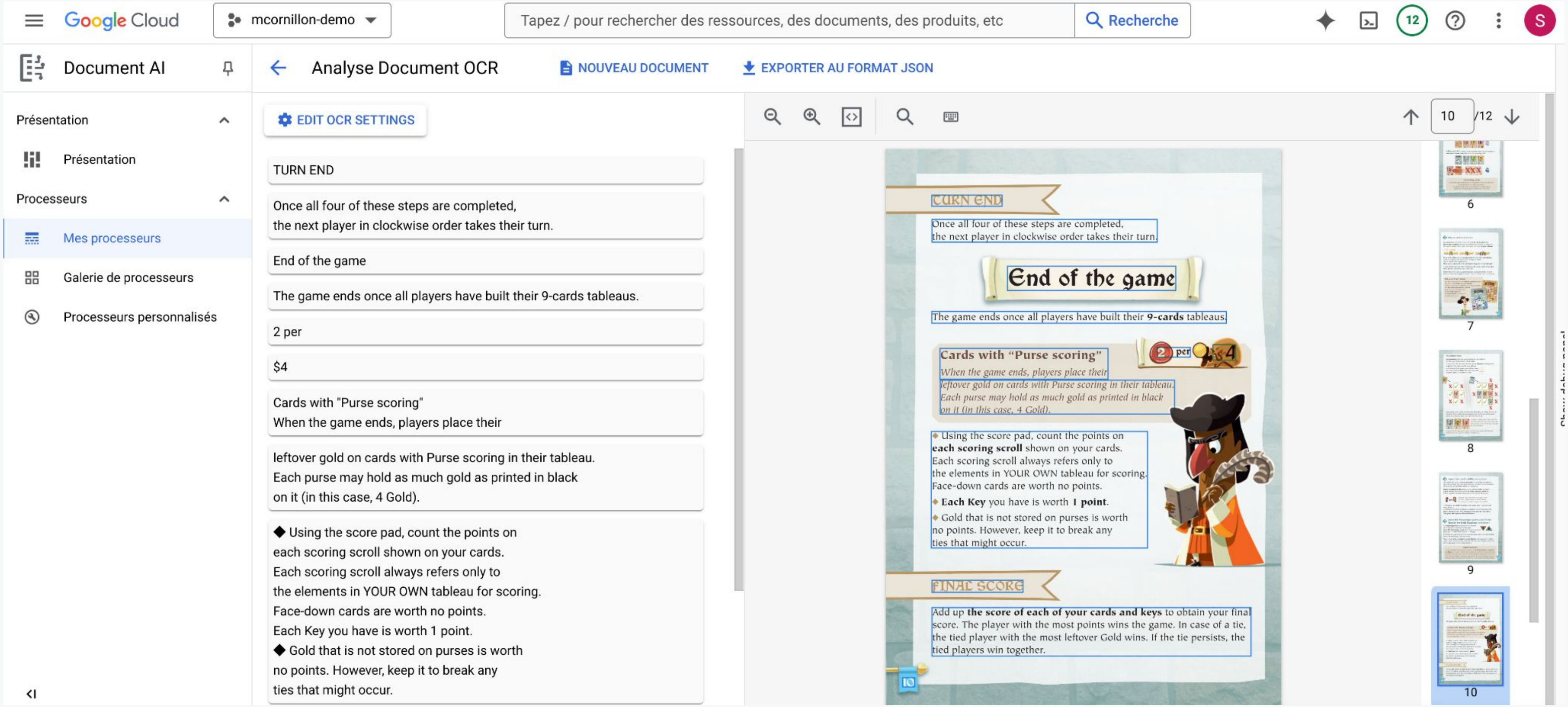
Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

pdf

AI

Extracting texts from PDF files



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their 9-cards tableaus.

2 per

$4

Cards with "Purse scoring"
When the game ends, players place their

leftover gold on cards with Purse scoring in their tableau.
Each purse may hold as much gold as printed in black
on it (in this case, 4 Gold).

◆ Using the score pad, count the points on
each scoring scroll shown on your cards.
Each scoring scroll always refers only to
the elements in YOUR OWN tableau for scoring.
Face-down cards are worth no points.
Each Key you have is worth 1 point.
◆ Gold that is not stored on purses is worth
no points. However, keep it to break any
ties that might occur.

TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

2 per 4

◆ Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.

◆ **Each Key** you have is worth **1 point**.

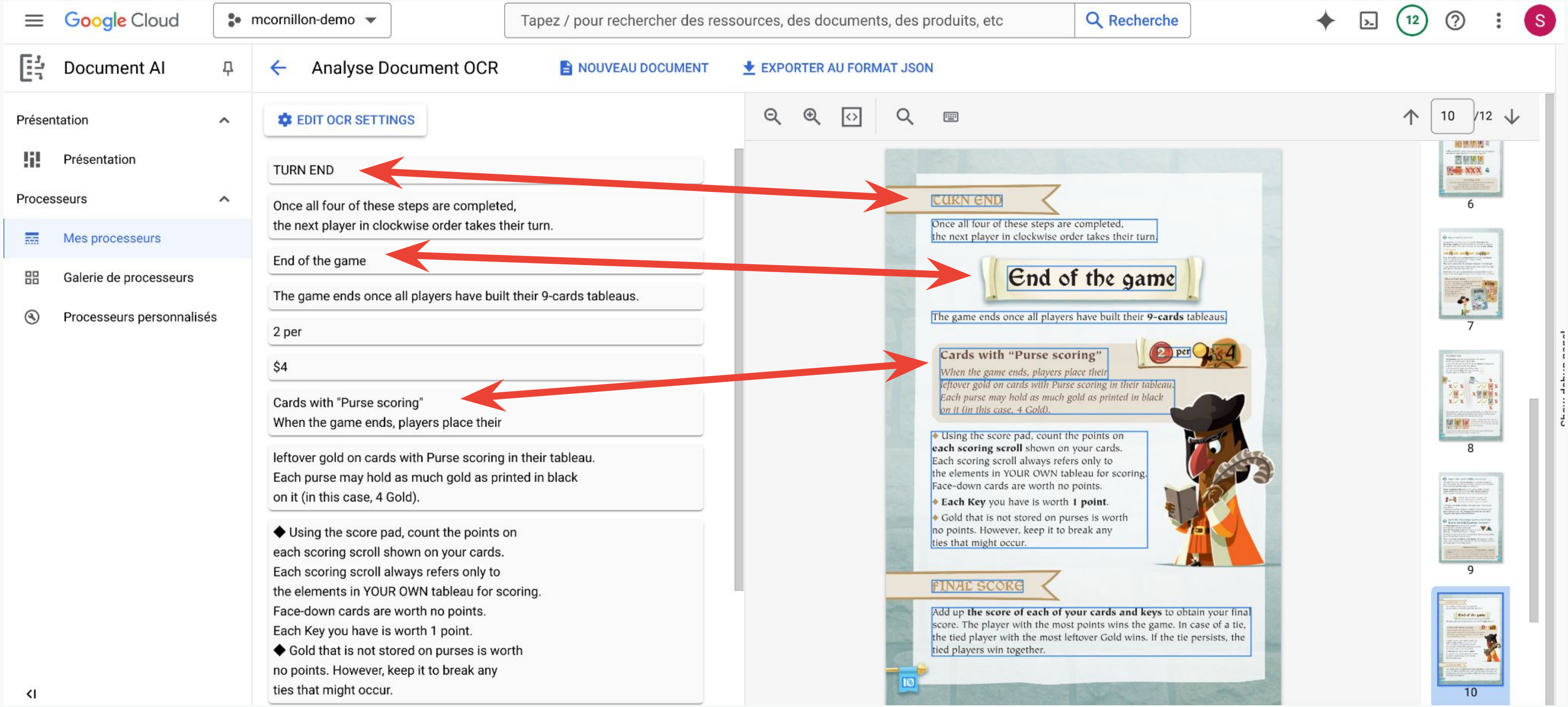
◆ Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

Extracting texts from PDF files



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their 9-cards tableaus.

2 per

$4

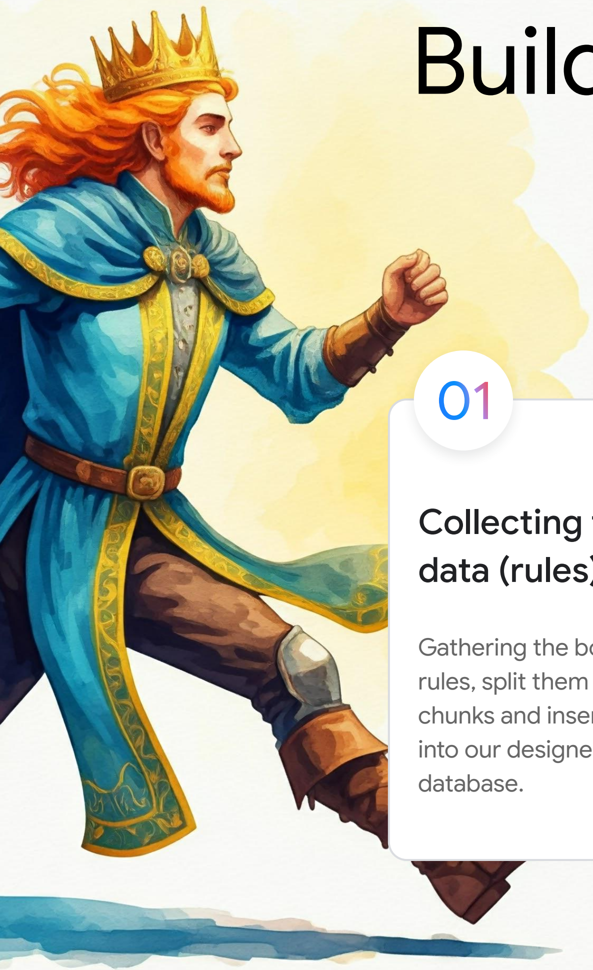
Cards with "Purse scoring"
When the game ends, players place their

leftover gold on cards with Purse scoring in their tableau.
Each purse may hold as much gold as printed in black
on it (in this case, 4 Gold).

◆ Using the score pad, count the points on
each scoring scroll shown on your cards.
Each scoring scroll always refers only to
the elements in YOUR OWN tableau for scoring.
Face-down cards are worth no points.
Each Key you have is worth 1 point.
◆ Gold that is not stored on purses is worth
no points. However, keep it to break any
ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

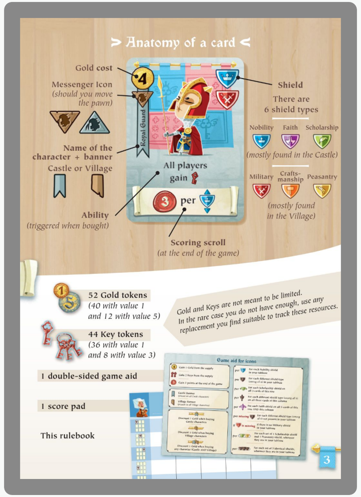


Building a board game chatbot with Postgres and RAG

01

Collecting the data (rules)

Gathering the board game rules, split them into chunks and insert them into our designed Postgres database.



> Anatomy of a card <

Gold cost

Messenger Icon
(should you move the pawn)

Name of the character + banner

Castle or Village

Ability
(triggered when bought)

Royal Guard

All players gain

3 per

Scoring scroll
(at the end of the game)

Shield

There are 6 shield types

Nobility Faith Scholarship
(mostly found in the Castle)

Military Craftsmanship Peasantry
(mostly found in the Village)

52 Gold tokens
(40 with value 1 and 12 with value 5)

44 Key tokens
(36 with value 1 and 8 with value 3)

Gold and Keys are not meant to be limited. In the rare case you do not have enough, use any replacement you find suitable to track these resources.

1 double-sided game aid

1 score pad

This rulebook

Game aid for icons

3

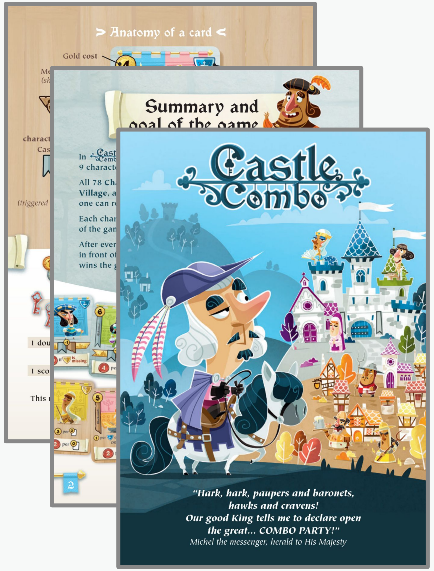
pdf

- Extracting texts from PDF files
- Splitting texts into chunks
- Inserting chunks into Postgres

In an ideal world

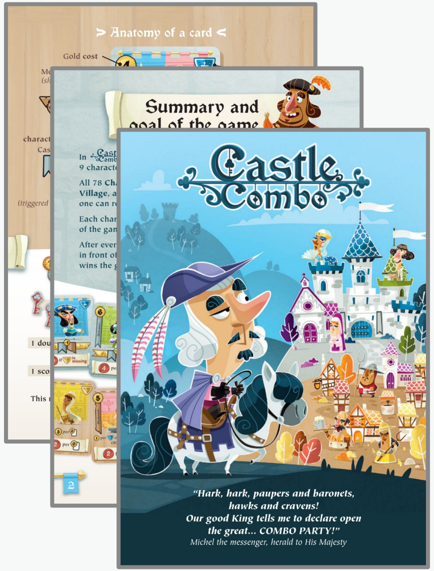
Splitting texts into chunks

In an ideal world



"Hark, hark, paupers and baronets,
hawks and cravens!
Our good King tells me to declare open
the great... COMBO PARTY!"
Michel the messenger, herald to His Majesty

In an ideal world

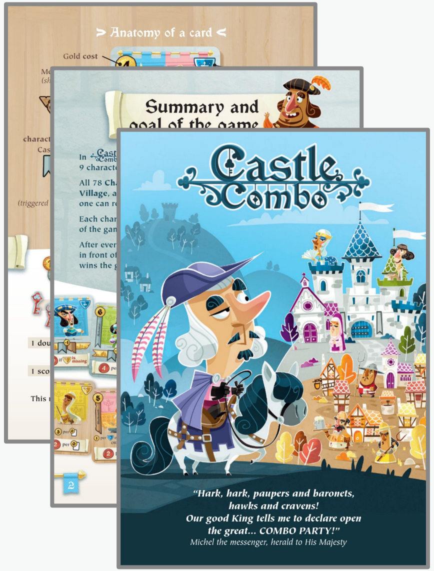


"Hark, hark, paupers and baronets, hawks and cravens! Our good King tells me to declare open the great... COMBO PARTY!"
Michel the messenger, herald to His Majesty

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

Splitting texts into chunks

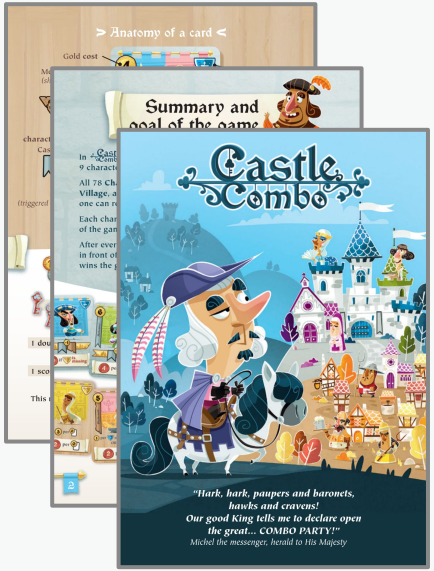
In an ideal world


> Anatomy of a card <
Summary and goal of the game
Castle Combo
"Hark, hark, paupers and baronets, hawks and cravens! Our good King tells me to declare open the great... COMBO PARTY!"
Michel the messenger, herald to His Majesty

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

Splitting texts into chunks

In an ideal world

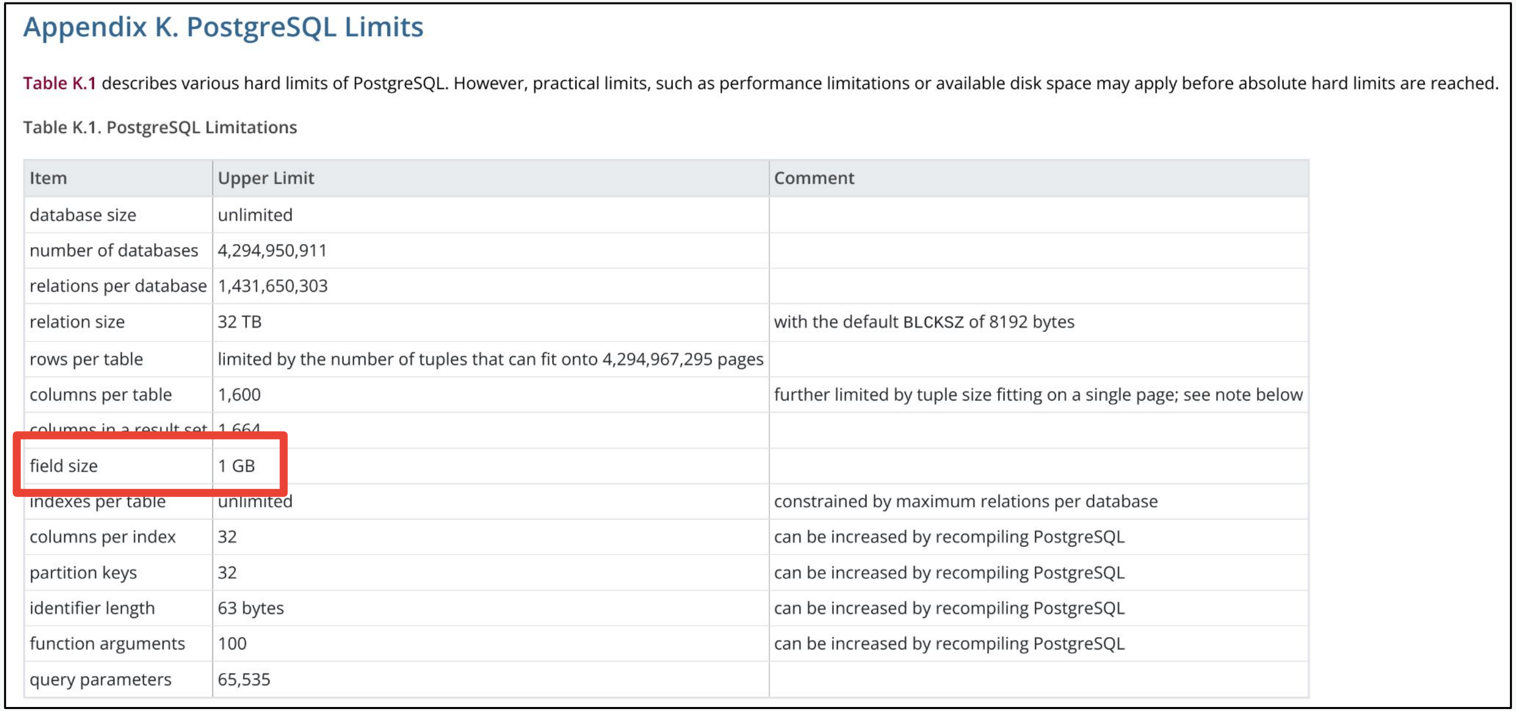

Anatomy of a card
Summary and goal of the game
Castle Combo
"Hark, hark, paupers and baronets, hawks and cravens! Our good King tells me to declare open the great... COMBO PARTY!"
Michel the messenger, herald to His Majesty

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

Gemini

But the world is not perfect (yet)

But the world is not perfect (yet)



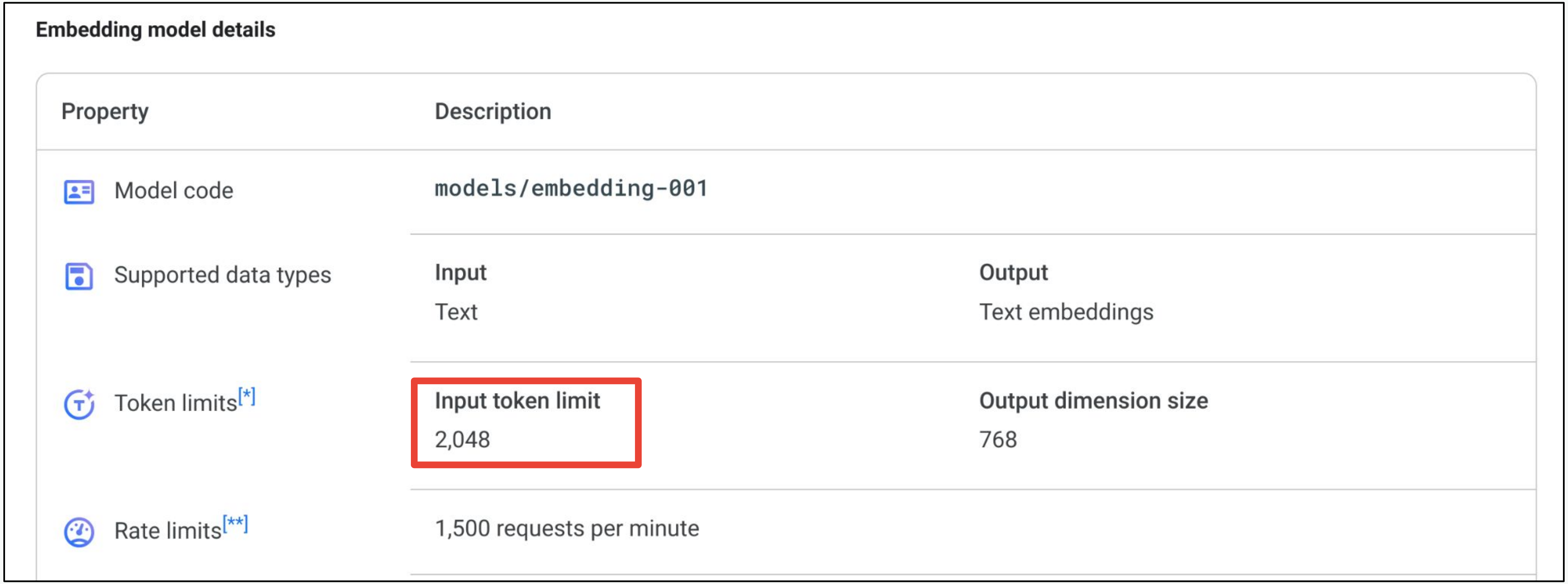
Appendix K. PostgreSQL Limits

Table K.1 describes various hard limits of PostgreSQL. However, practical limits, such as performance limitations or available disk space may apply before absolute hard limits are reached.

Table K.1. PostgreSQL Limitations

Item	Upper Limit	Comment
database size	unlimited	
number of databases	4,294,950,911	
relations per database	1,431,650,303	
relation size	32 TB	with the default `BLCKSZ` of 8192 bytes
rows per table	limited by the number of tuples that can fit onto 4,294,967,295 pages	
columns per table	1,600	further limited by tuple size fitting on a single page; see note below
columns in a result set	1,664	
field size	1 GB	
indexes per table	unlimited	constrained by maximum relations per database
columns per index	32	can be increased by recompiling PostgreSQL
partition keys	32	can be increased by recompiling PostgreSQL
identifier length	63 bytes	can be increased by recompiling PostgreSQL
function arguments	100	can be increased by recompiling PostgreSQL
query parameters	65,535	

But the world is not perfect (yet)



Embedding model details

Property	Description	
Model code	`models/embedding-001`	
Supported data types	**Input** Text	**Output** Text embeddings
Token limits[*]	**Input token limit** 2,048	**Output dimension size** 768
Rate limits[**]	1,500 requests per minute	

But the world is not perfect (yet)

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

But the world is not perfect (yet)

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

1 token = 4 characters

But the world is not perfect (yet)

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

1 token = 4 characters

100 tokens = 60-80 English words

But the world is not perfect (yet)

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

1 token = 4 characters

100 tokens = 60-80 English words

12007 tokens

But the world is not perfect (yet)

```
(...)
End of the game
The game ends once all players have built their
9-cards  tableaus.
Cards with “Purse scoring”
When the game ends, players place their
leftover gold on cards with Purse scoring in
their tableau.
Each purse may hold as much gold as printed in
black
on it (in this case, 4 Gold).
Once all four of these steps are completed,
the next player in clockwise order takes their
turn.
(...)
```

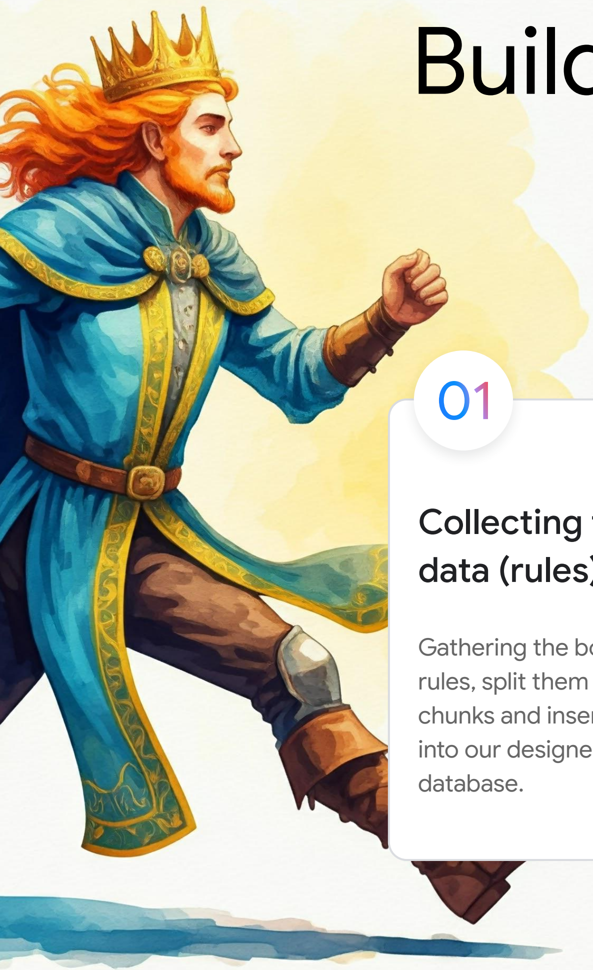
1 token = 4 characters

100 tokens = 60-80 English words

→ **12007 tokens**

$

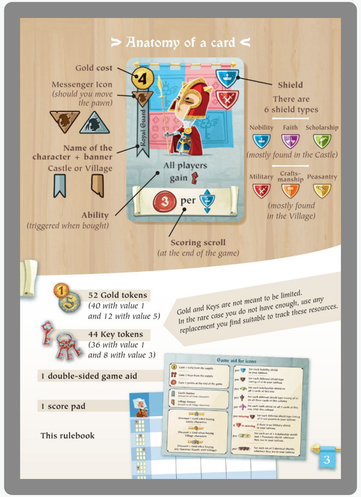
Building a board game chatbot
with Postgres and RAG



01

Collecting the data (rules)

Gathering the board game rules, split them into chunks and insert them into our designed Postgres database.



> Anatomy of a card <

Gold cost

Messenger Icon
(should you move the pawn)

Name of the character + banner

Castle or Village

Ability
(triggered when bought)

Shield

There are
6 shield types

Nobility Faith Scholarship
(mostly found in the Castle)

Military Craftsmanship Peasantry
(mostly found in the Village)

Scoring scroll
(at the end of the game)

52 Gold tokens
(40 with value 1 and 12 with value 5)

44 Key tokens
(36 with value 1 and 8 with value 3)

Gold and Keys are not meant to be limited. In the rare case you do not have enough, use any replacement you find suitable to track these resources.

1 double-sided game aid

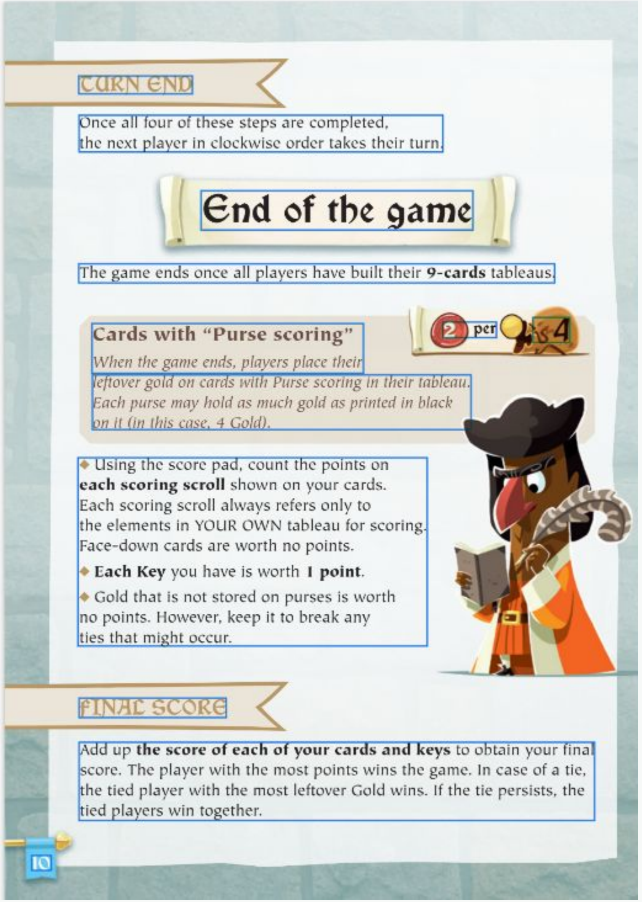
1 score pad

This rulebook

pdf

- Extracting texts from PDF files
- Splitting texts into chunks
- Inserting chunks into Postgres

Inserting chunks into Postgres



TURN END

Once all four of these steps are completed, the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

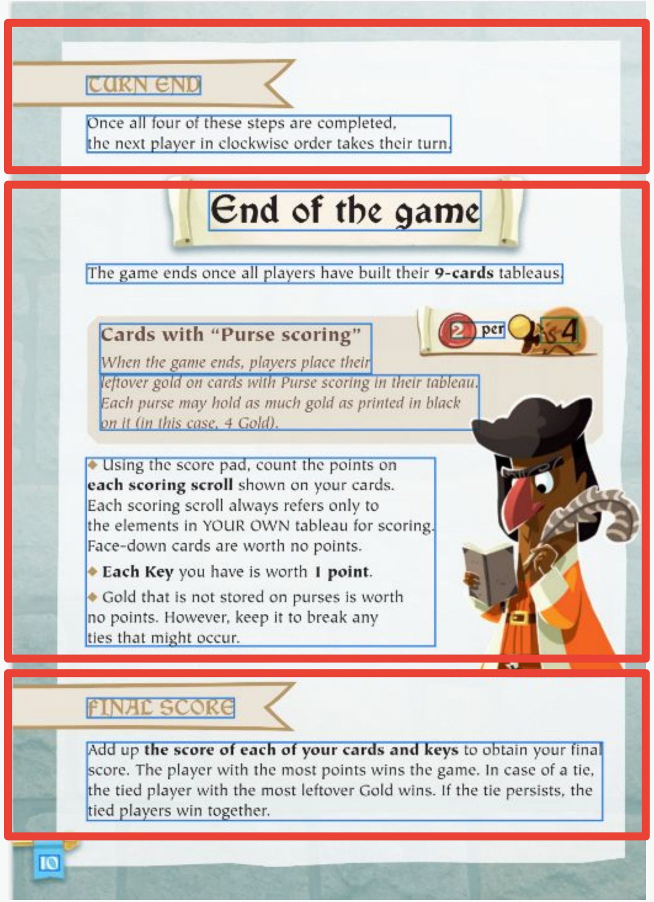
When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

Inserting chunks into Postgres



TURN END

Once all four of these steps are completed, the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

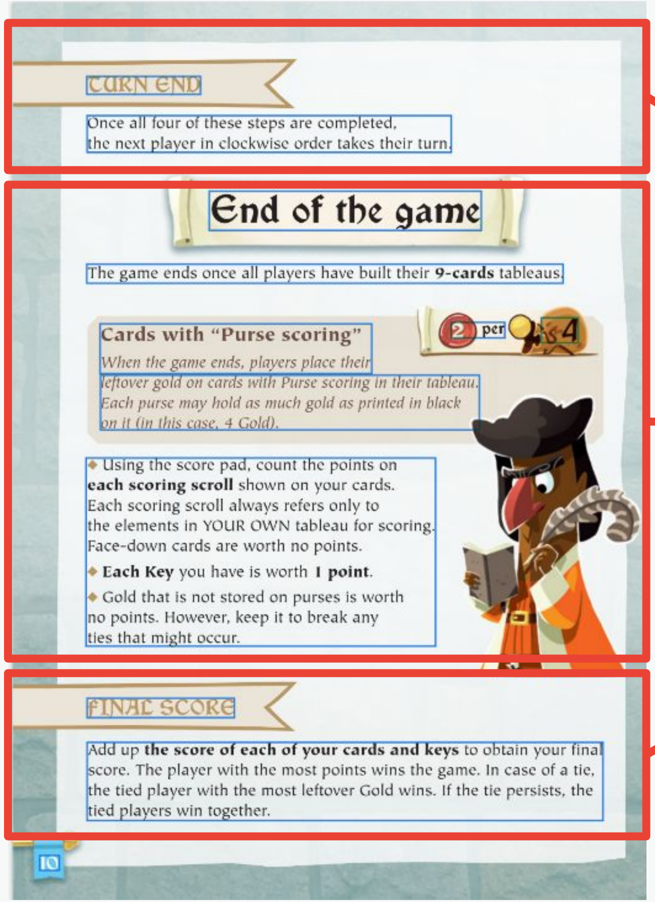
When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

Inserting chunks into Postgres



TURN END

Once all four of these steps are completed, the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

2 per 4

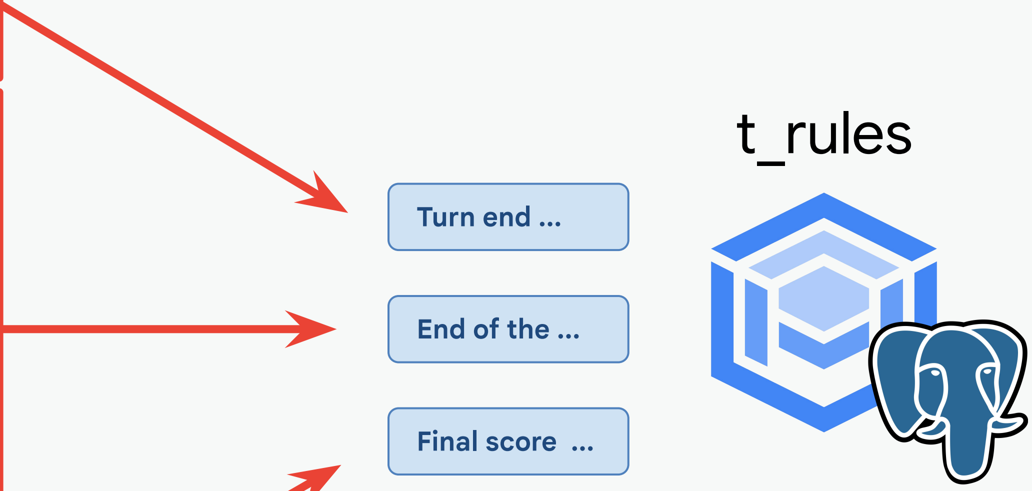
When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10



t_rules

Turn end ...

End of the ...

Final score ...

Building a board game chatbot with Postgres and RAG

02

Generating the embeddings of the rules and prompts

Transforming those texts into usable vectors for similarity searches.

Building a board game chatbot with Postgres and RAG

02

Generating the embeddings of the rules and prompts

Transforming those texts into usable vectors for similarity searches.

game_id	rule_id	content	embedding
Castle combo	1	Once all four of these steps are completed (...)	<null>
Castle combo	2	The game ends once all players (...)	<null>
Castle combo	3	Add up the score of each of your cards and keys (...)	<null>
Wingspan	4	Place the following items in the center of the table (...)	<null>

Building a board game chatbot with Postgres and RAG

02

Generating the embeddings of the rules and prompts

Transforming those texts into usable vectors for similarity searches.

game_id	rule_id	content	embedding
Castle combo	1	Once all four of these steps are completed (...)	<null>
Castle combo	2	The game ends once all players (...)	<null>
Castle combo	3	Add up the score of each of your cards and keys (...)	<null>
Wingspan	4	Place the following items in the center of the table (...)	<null>

2 Generating the embeddings of the rules and prompts

Gemini

text-embedding-004

Add up the score of each of your cards and keys to obtain your final score. The player with the most points wins.

2

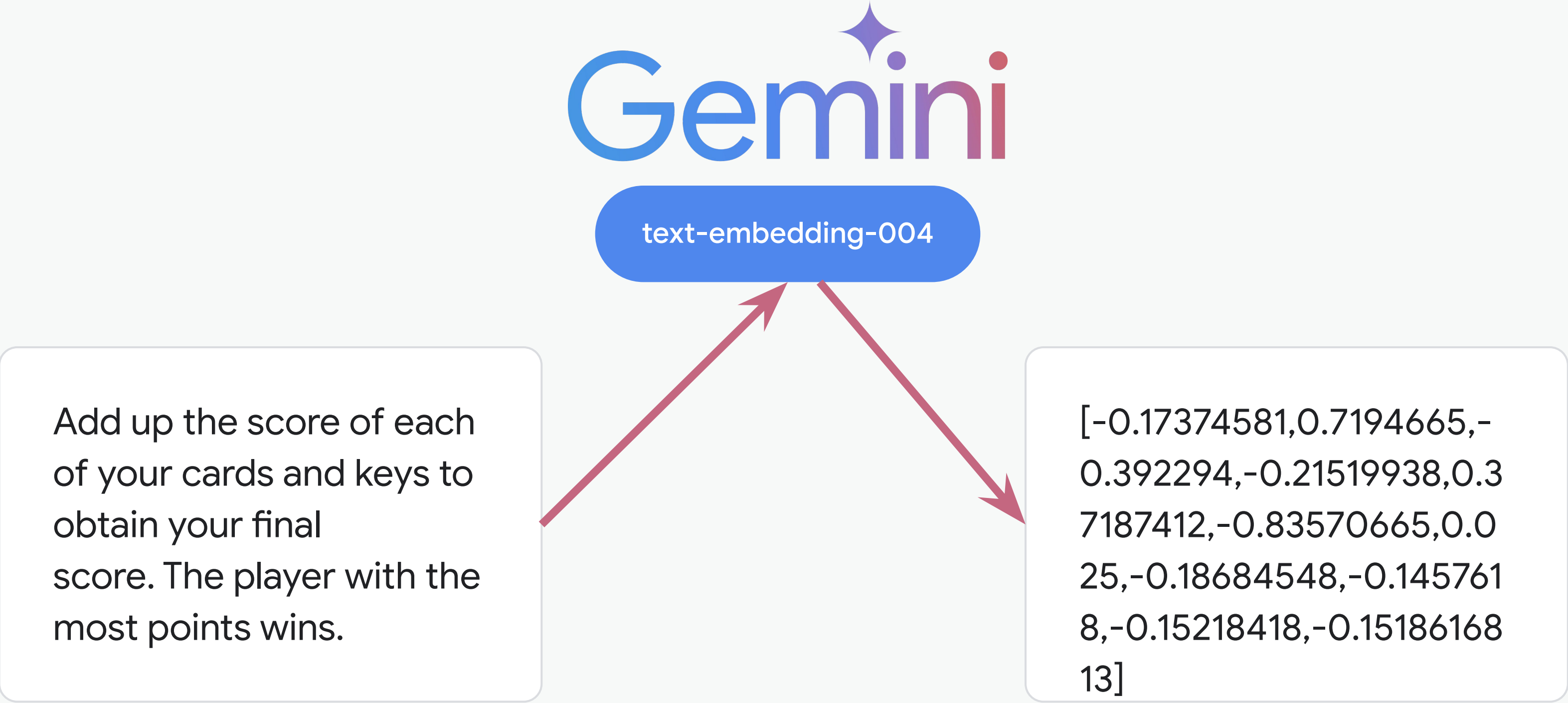
Generating the embeddings of the rules and prompts

Gemini

text-embedding-004

Add up the score of each of your cards and keys to obtain your final score. The player with the most points wins.

2 Generating the embeddings of the rules and prompts



Add up the score of each of your cards and keys to obtain your final score. The player with the most points wins.

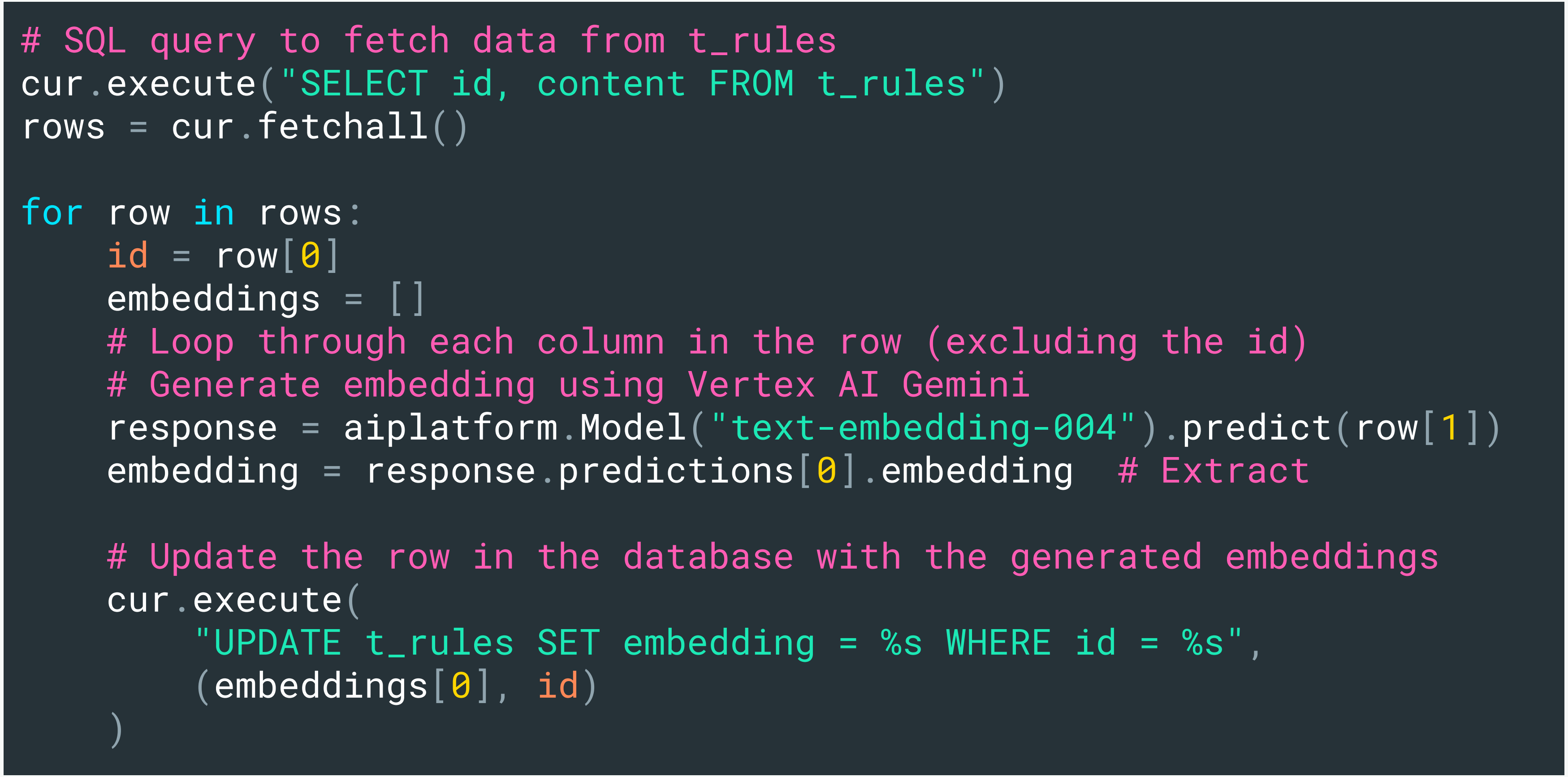
```
[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025,-0.18684548,-0.1457618,-0.15218418,-0.1518616813]
```

pgvector

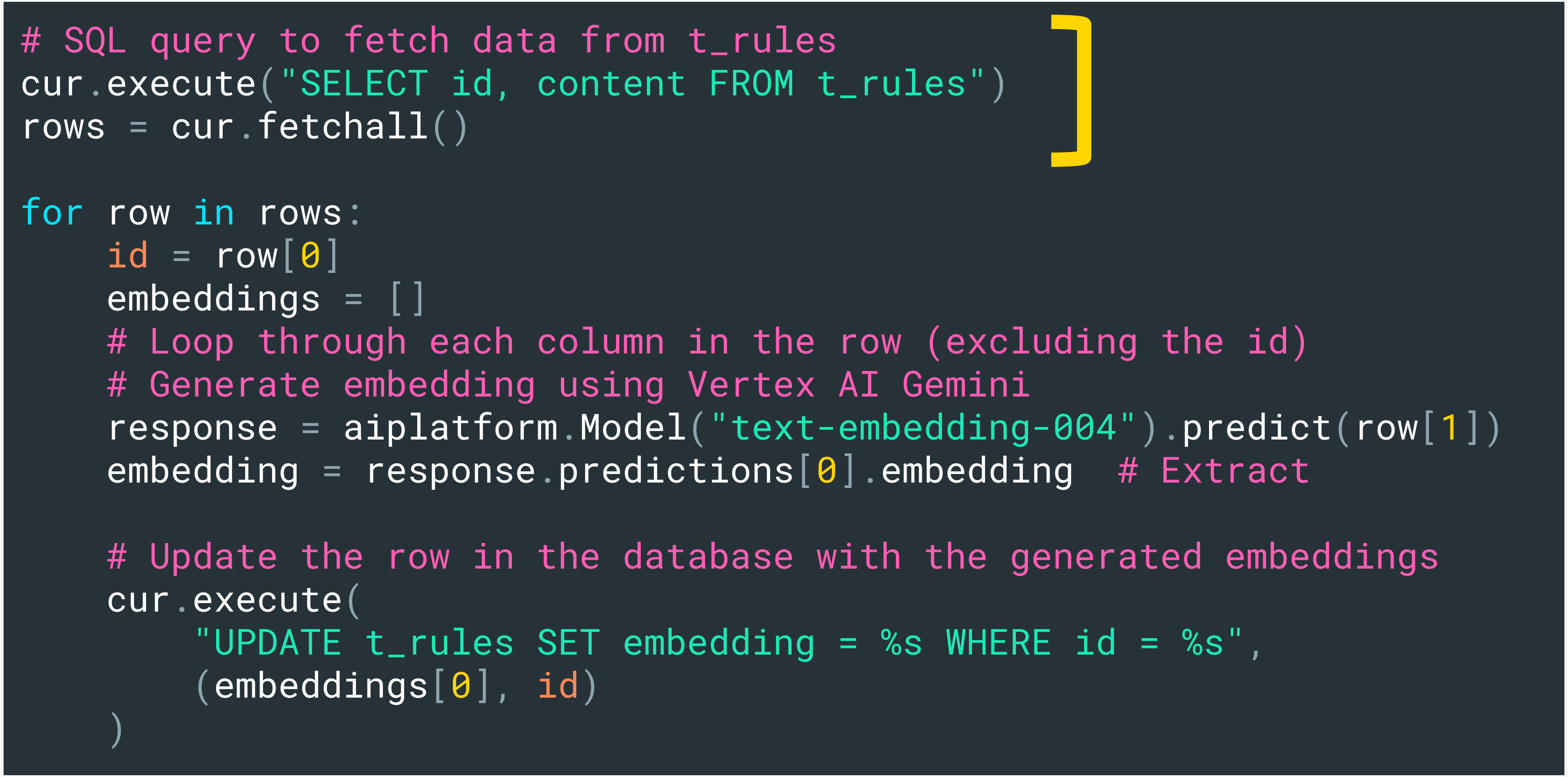
pgvector

```sql
CREATE EXTENSION vector;
```

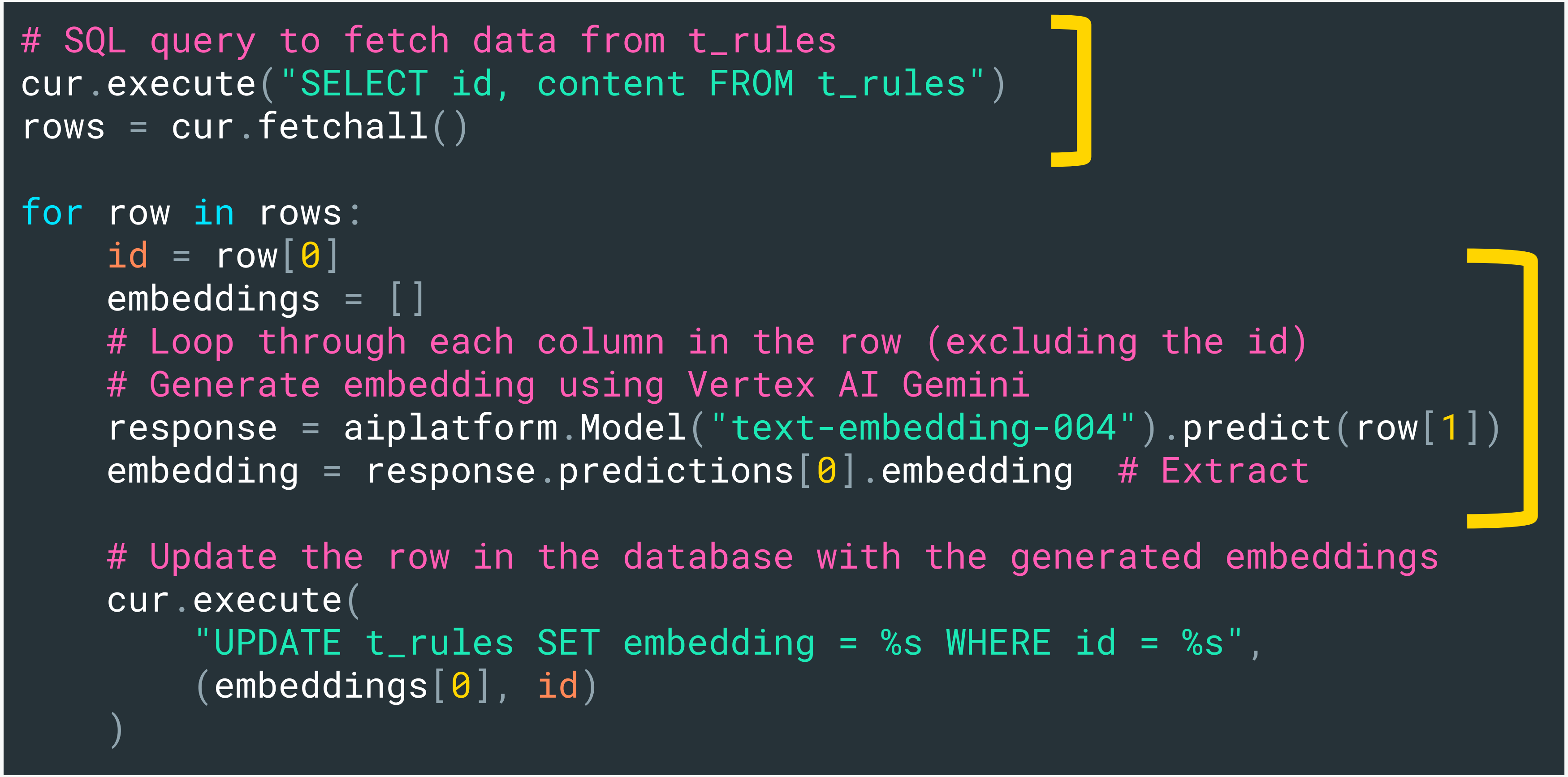
```sql
ALTER TABLE t_rules
ADD COLUMN embedding VECTOR(768);
```



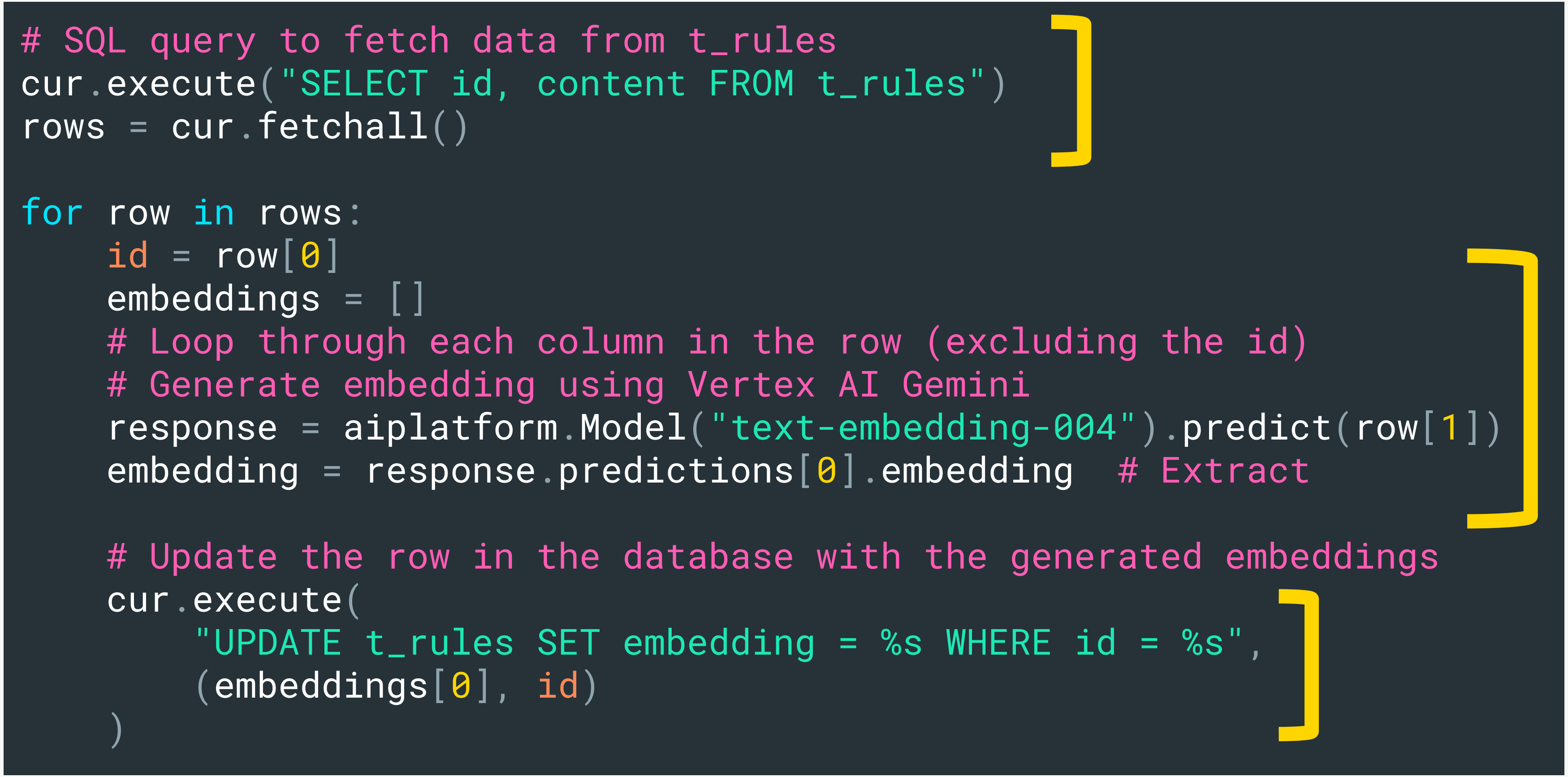
```
# SQL query to fetch data from t_rules
cur.execute("SELECT id, content FROM t_rules")
rows = cur.fetchall()

for row in rows:
    id = row[0]
    embeddings = []
    # Loop through each column in the row (excluding the id)
    # Generate embedding using Vertex AI Gemini
    response = aiplatform.Model("text-embedding-004").predict(row[1])
    embedding = response.predictions[0].embedding  # Extract

    # Update the row in the database with the generated embeddings
    cur.execute(
        "UPDATE t_rules SET embedding = %s WHERE id = %s",
        (embeddings[0], id)
    )
```



```python
# SQL query to fetch data from t_rules
cur.execute("SELECT id, content FROM t_rules")
rows = cur.fetchall()

for row in rows:
    id = row[0]
    embeddings = []
    # Loop through each column in the row (excluding the id)
    # Generate embedding using Vertex AI Gemini
    response = aiplatform.Model("text-embedding-004").predict(row[1])
    embedding = response.predictions[0].embedding  # Extract

    # Update the row in the database with the generated embeddings
    cur.execute(
        "UPDATE t_rules SET embedding = %s WHERE id = %s",
        (embeddings[0], id)
    )
```



```
# SQL query to fetch data from t_rules
cur.execute("SELECT id, content FROM t_rules")
rows = cur.fetchall()

for row in rows:
    id = row[0]
    embeddings = []
    # Loop through each column in the row (excluding the id)
    # Generate embedding using Vertex AI Gemini
    response = aiplatform.Model("text-embedding-004").predict(row[1])
    embedding = response.predictions[0].embedding  # Extract

    # Update the row in the database with the generated embeddings
    cur.execute(
        "UPDATE t_rules SET embedding = %s WHERE id = %s",
        (embeddings[0], id)
    )
```



```
# SQL query to fetch data from t_rules
cur.execute("SELECT id, content FROM t_rules")
rows = cur.fetchall()

for row in rows:
    id = row[0]
    embeddings = []
    # Loop through each column in the row (excluding the id)
    # Generate embedding using Vertex AI Gemini
    response = aiplatform.Model("text-embedding-004").predict(row[1])
    embedding = response.predictions[0].embedding  # Extract

    # Update the row in the database with the generated embeddings
    cur.execute(
        "UPDATE t_rules SET embedding = %s WHERE id = %s",
        (embeddings[0], id)
    )
```

Google Cloud

Cloud SQL

AlloyDB

Google Cloud

Cloud SQL

AlloyDB

```
CREATE EXTENSION google_ml_integration;
```

Google Cloud

Cloud SQL

AlloyDB

```
CREATE EXTENSION google_ml_integration;
```

```
UPDATE t_rules
SET embedding =
embedding('text-embedding-004', content);
```

Building a board game chatbot with Postgres and RAG

02

Generating the embeddings of the rules

Transforming those texts into usable vectors for similarity searches.

game_id	rule_id	content	embedding
Castle combo	1	Once all four of these steps are completed (...)	[-0.15,25,-0.151,-0.1458,-0.161168181]
Castle combo	2	The game ends once all players (...)	[-0.615,25,-0.151,-0.8484,-0.894986]
Castle combo	3	Add up the score of each of your cards and keys (...)	[-0.75,25,-0.151,-0.51157,-0.757575779]
Wingspan	4	Place the following items in the center of the table (...)	[-0.36,25,-0.151,-0.8181,-0.887452954]

Building a board game chatbot with Postgres and RAG

03

Finding the rules using similarity searches

Explore similarity search with Postgres and pgvector to find the most appropriate rules.

Building a board game chatbot with Postgres and RAG

03

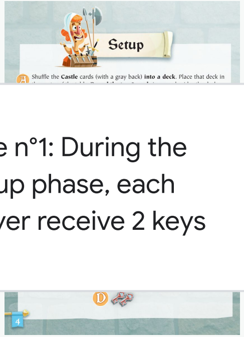
Finding the rules using similarity searches

Explore similarity search with Postgres and pgvector to find the most appropriate rules.

> In the board game "Castle Combo", how many keys should I start with?

[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025]

game_id	rule_id	content	embedding
Castle combo	1	Once all four of these steps are completed (...)	[-0.15,25,-0.151,-0.1458,-0.161168181]
Castle combo	2	The game ends once all players (...)	[-0.615,25,-0.151,-0.8484,-0.894986]
Castle combo	3	Add up the score of each of your cards and keys (...)	[-0.75,25,-0.151,-0.51157,-0.757575779]
Wingspan	4	Place the following items in the center of the table (...)	[-0.36,25,-0.151,-0.8181,-0.887452954]



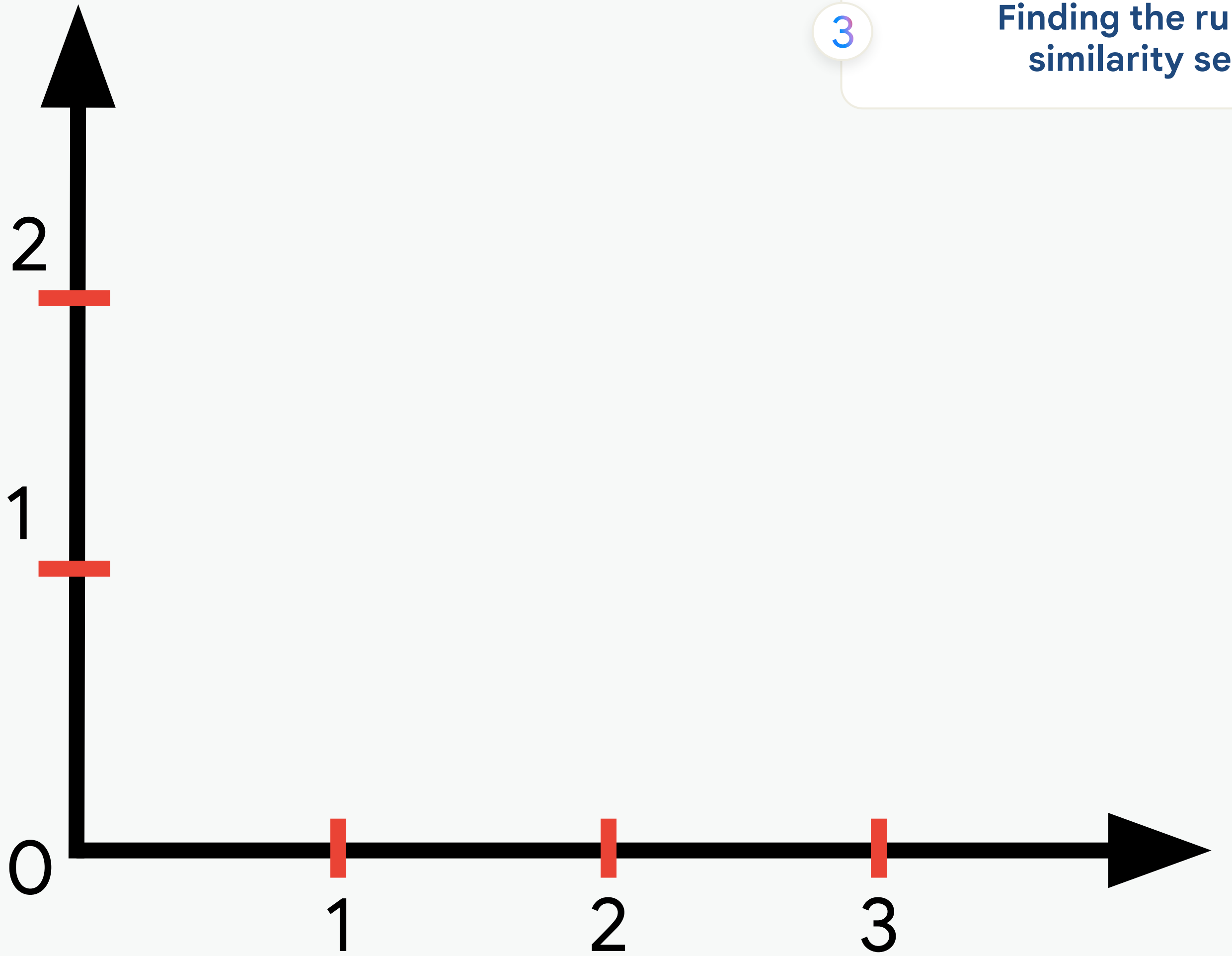
Setup

A Shuffle the **Castle** cards (with a gray back) **into a deck**. Place that deck in

Rule n°1: During the setup phase, each player receive 2 keys

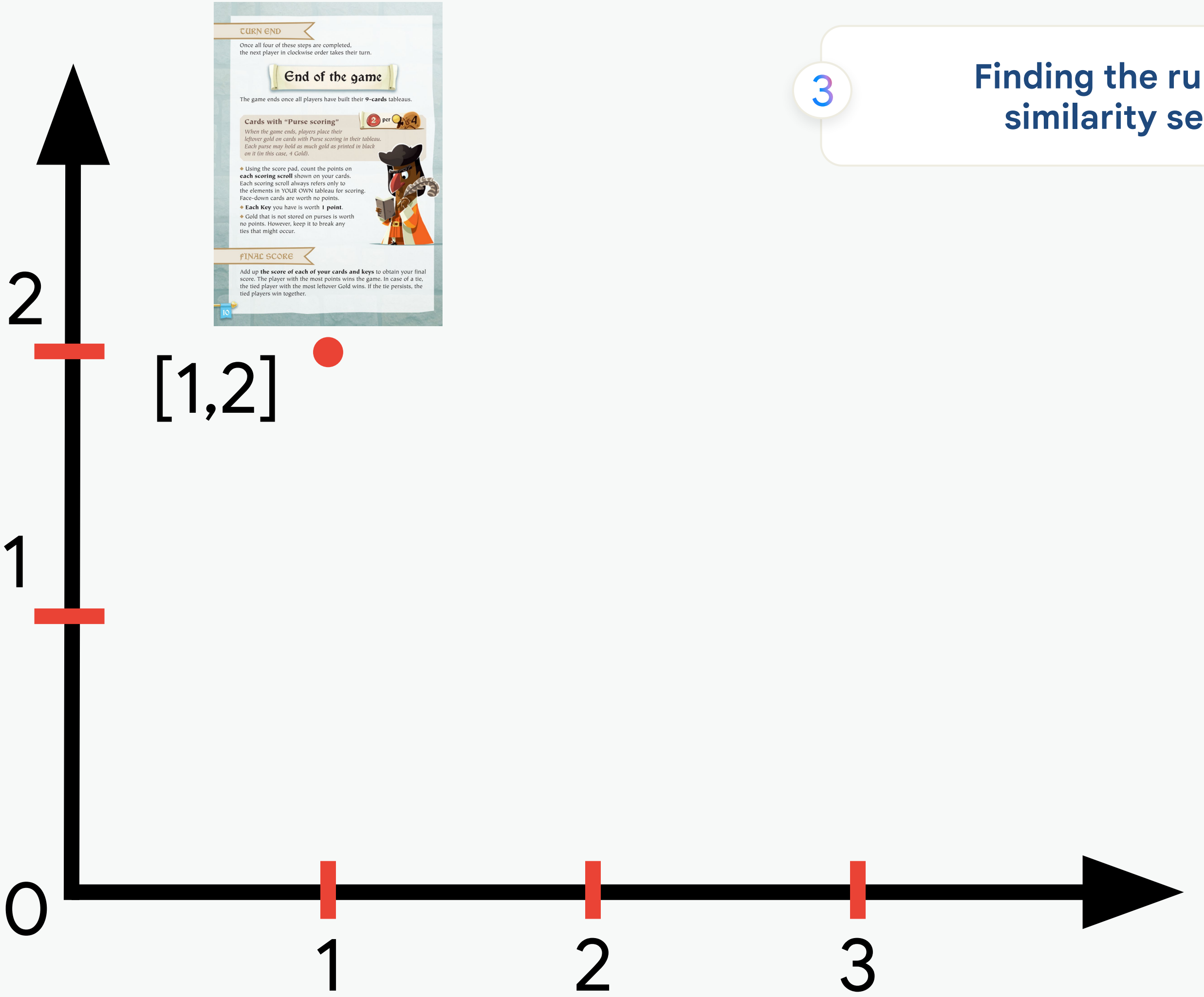
3

Finding the rules using similarity searches



3

Finding the rules using similarity searches



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

2 per 4

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

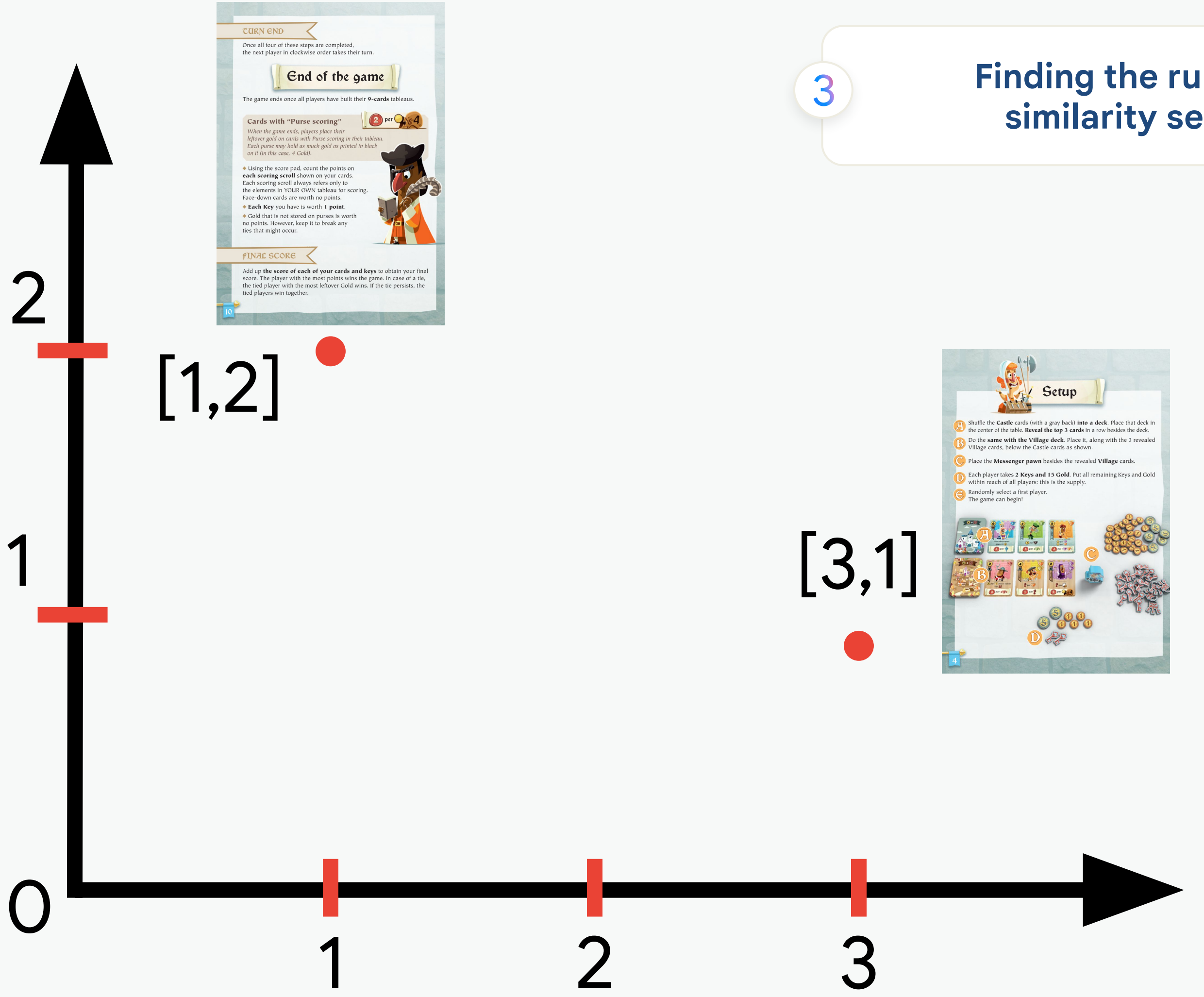
FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

3

Finding the rules using similarity searches



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

◆ Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.

◆ **Each Key** you have is worth **1 point**.

◆ Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

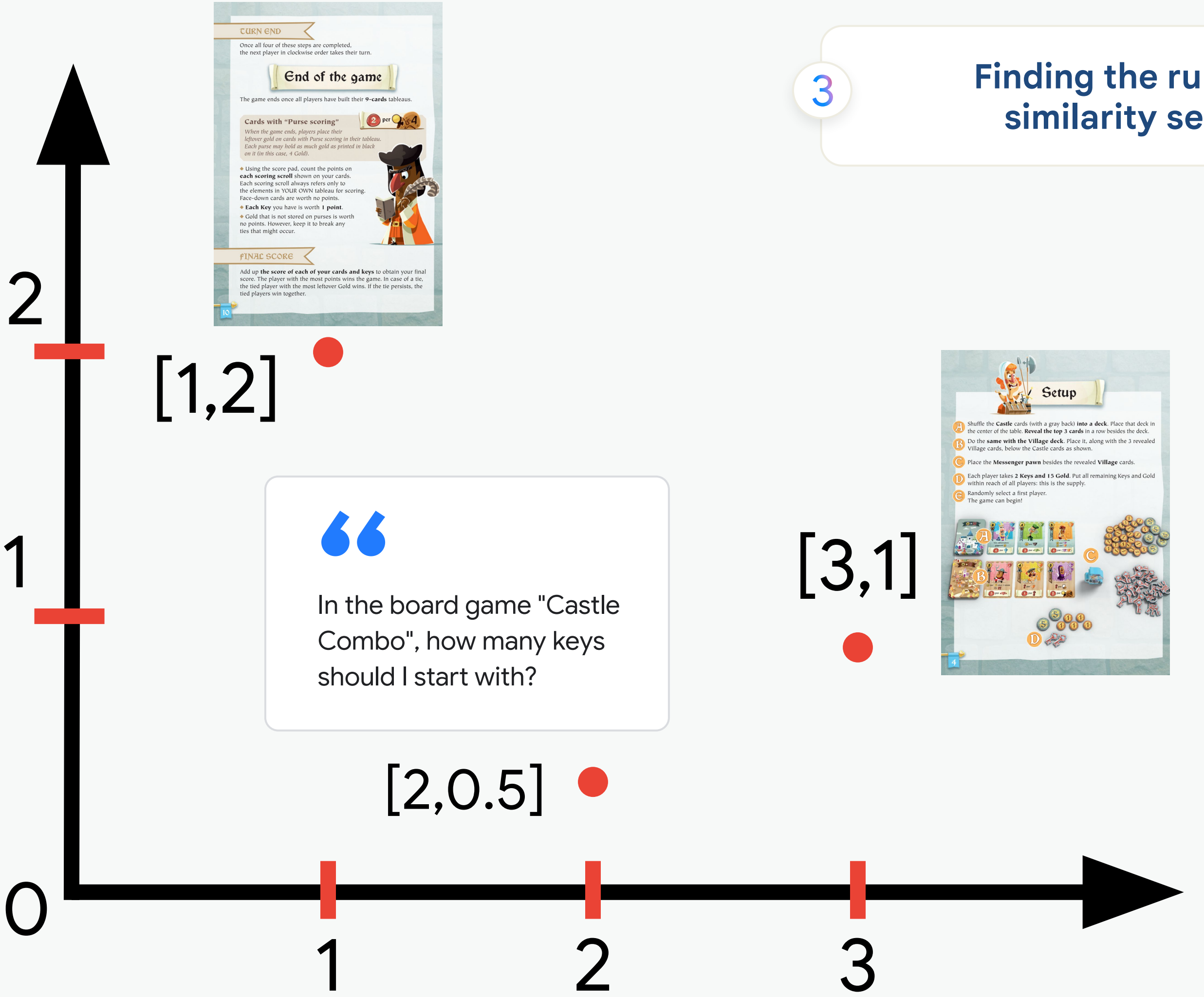
Setup

A Shuffle the **Castle** cards (with a gray back) **into a deck**. Place that deck in the center of the table. **Reveal the top 3 cards** in a row besides the deck.
B Do the **same with the Village deck**. Place it, along with the 3 revealed Village cards, below the Castle cards as shown.
C Place the **Messenger pawn** besides the revealed **Village** cards.
D Each player takes **2 Keys and 15 Gold**. Put all remaining Keys and Gold within reach of all players: this is the supply.
E Randomly select a first player.
The game can begin!

4

3

Finding the rules using similarity searches



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

2 per 4

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

Setup

Ⓐ Shuffle the **Castle** cards (with a gray back) **into a deck**. Place that deck in the center of the table. **Reveal the top 3 cards** in a row besides the deck.

Ⓑ Do the **same with the Village deck**. Place it, along with the 3 revealed Village cards, below the Castle cards as shown.

Ⓒ Place the **Messenger pawn** besides the revealed **Village** cards.

Ⓓ Each player takes **2 Keys and 15 Gold**. Put all remaining Keys and Gold within reach of all players: this is the supply.

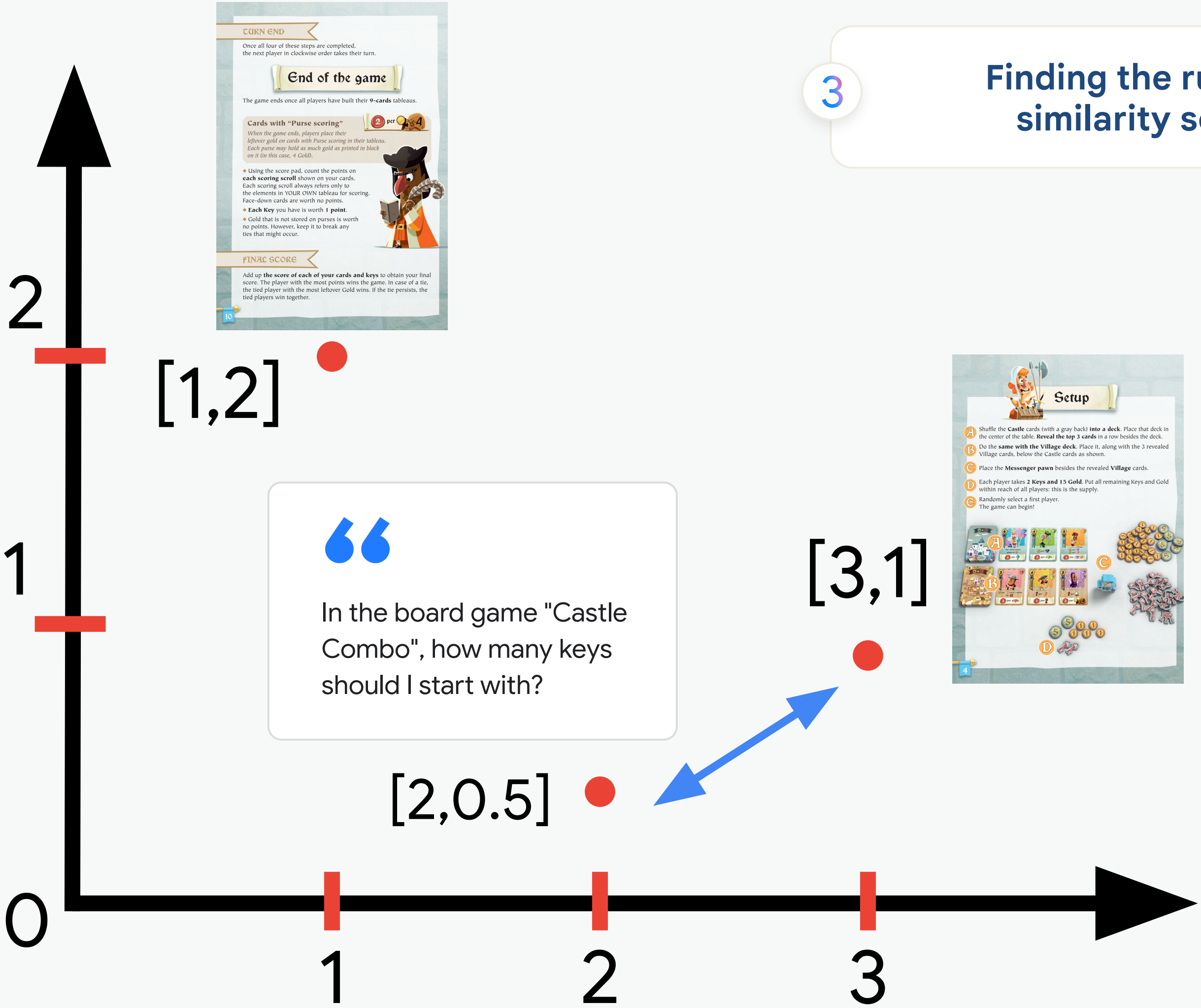
Ⓔ Randomly select a first player.
The game can begin!

4

> In the board game "Castle Combo", how many keys should I start with?

3

Finding the rules using similarity searches



TURN END

Once all four of these steps are completed, the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

2 per 4

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

◆ Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.

◆ **Each Key** you have is worth **1 point**.

◆ Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

Setup

A Shuffle the **Castle** cards (with a gray back) **into a deck**. Place that deck in the center of the table. **Reveal the top 3 cards** in a row besides the deck.

B Do the **same with the Village deck**. Place it, along with the 3 revealed Village cards, below the Castle cards as shown.

C Place the **Messenger pawn** besides the revealed **Village** cards.

D Each player takes **2 Keys and 15 Gold**. Put all remaining Keys and Gold within reach of all players: this is the supply.

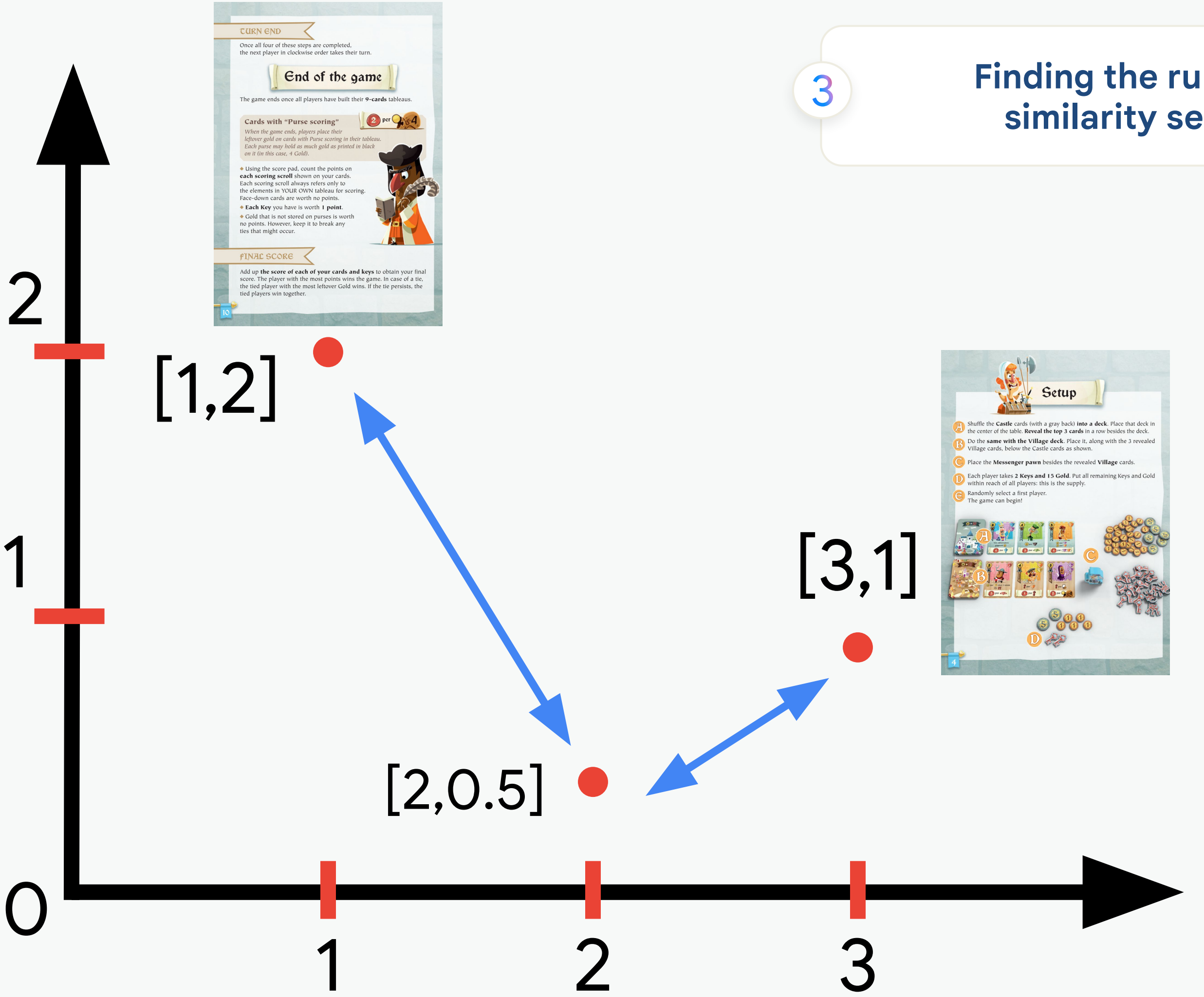
E Randomly select a first player.
The game can begin!

4

> In the board game "Castle Combo", how many keys should I start with?

3

Finding the rules using similarity searches



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

2 per 4

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

Setup

Ⓐ Shuffle the **Castle** cards (with a gray back) **into a deck**. Place that deck in the center of the table. **Reveal the top 3 cards** in a row besides the deck.

Ⓑ Do the **same with the Village deck**. Place it, along with the 3 revealed Village cards, below the Castle cards as shown.

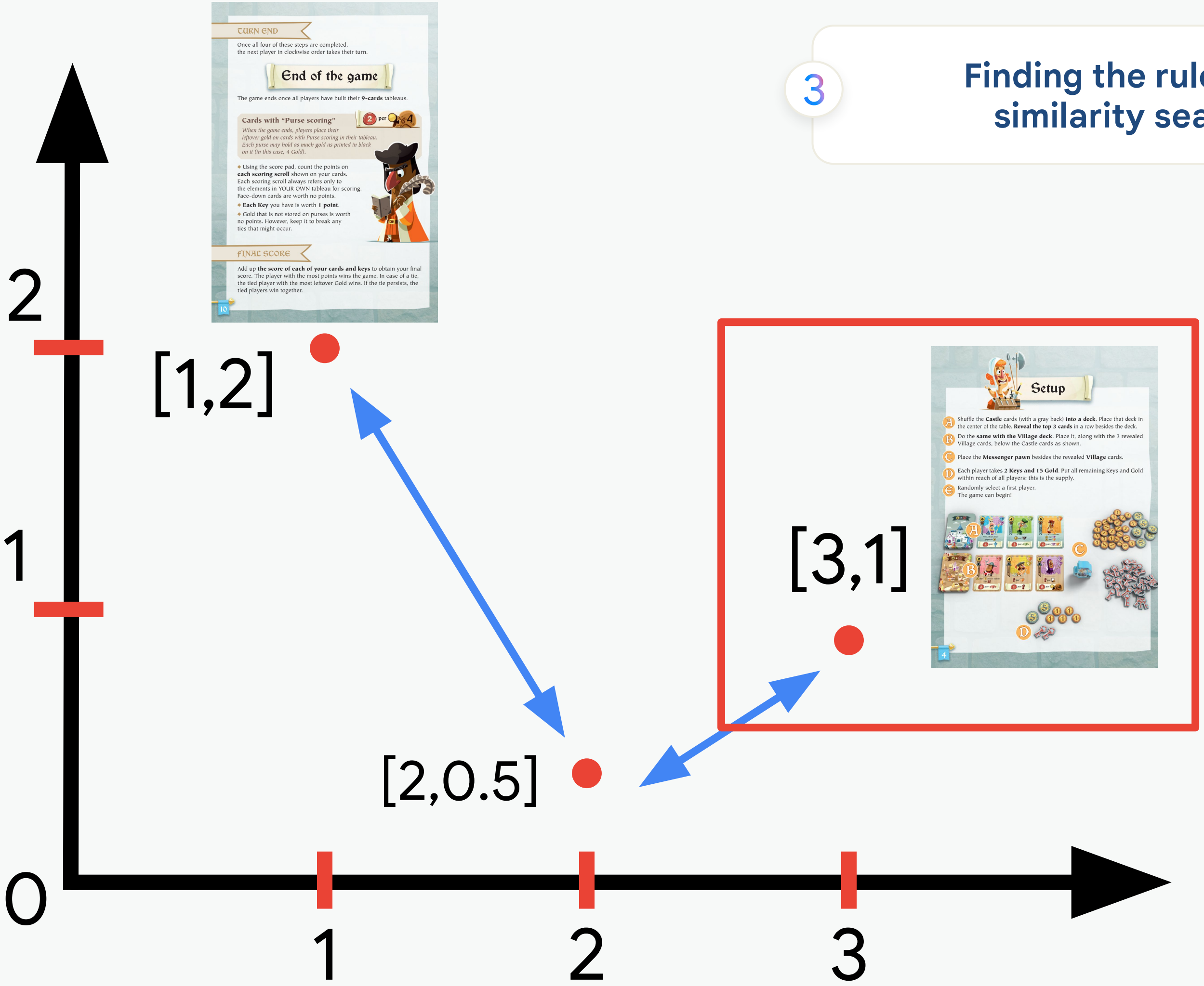
Ⓒ Place the **Messenger pawn** besides the revealed **Village** cards.

Ⓓ Each player takes **2 Keys and 15 Gold**. Put all remaining Keys and Gold within reach of all players: this is the supply.

Ⓔ Randomly select a first player.
The game can begin!

4

3 Finding the rules using similarity searches



TURN END

Once all four of these steps are completed,
the next player in clockwise order takes their turn.

End of the game

The game ends once all players have built their **9-cards** tableaus.

Cards with "Purse scoring"

When the game ends, players place their leftover gold on cards with Purse scoring in their tableau. Each purse may hold as much gold as printed in black on it (in this case, 4 Gold).

- Using the score pad, count the points on **each scoring scroll** shown on your cards. Each scoring scroll always refers only to the elements in YOUR OWN tableau for scoring. Face-down cards are worth no points.
- **Each Key** you have is worth **1 point**.
- Gold that is not stored on purses is worth no points. However, keep it to break any ties that might occur.

FINAL SCORE

Add up **the score of each of your cards and keys** to obtain your final score. The player with the most points wins the game. In case of a tie, the tied player with the most leftover Gold wins. If the tie persists, the tied players win together.

10

Setup

Ⓐ Shuffle the **Castle** cards (with a gray back) **into a deck**. Place that deck in the center of the table. **Reveal the top 3 cards** in a row besides the deck.

Ⓑ Do the **same with the Village deck**. Place it, along with the 3 revealed Village cards, below the Castle cards as shown.

Ⓒ Place the **Messenger pawn** besides the revealed **Village** cards.

Ⓓ Each player takes **2 Keys and 15 Gold**. Put all remaining Keys and Gold within reach of all players: this is the supply.

Ⓔ Randomly select a first player.
The game can begin!

4

pgvector

3

Finding the rules using similarity searches

3

Finding the rules using similarity searches

pgvector

```
SELECT *
FROM t_rules
ORDER BY
embedding <-> '[3,1,2]' LIMIT 5;
```

3 Finding the rules using similarity searches

pgvector

```
SELECT *
FROM t_rules
ORDER BY
embedding <-> '[3,1,2]' LIMIT 5;
```

3 Finding the rules using similarity searches

pgvector

<-> - L2 distance

<=> - cosine distance

<+> - L1 distance

<#> - (negative) inner product

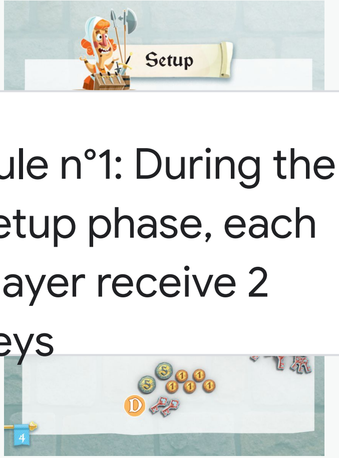
<~> - Hamming distance

<%> - Jaccard distance

> In the board game "Castle Combo", how many keys should I start with?

[-0.17374581,0.7194665,-0.392294,-0.21519938,0.37187412,-0.83570665,0.025]

<=> cosine distance



Rule n°1: During the setup phase, each player receive 2 keys

game_id	rule_id	content	embedding
Castle combo	1	Once all four of these steps are completed (...)	[-0.15,25,-0.151,-0.1458,-0.161168181]
Castle combo	2	The game ends once all players (...)	[-0.615,25,-0.151,-0.8484,-0.894986]
Castle combo	3	Add up the score of each of your cards and keys (...)	[-0.75,25,-0.151,-0.51157,-0.757575779]
Wingspan	4	Place the following items in the center of the table (...)	[-0.36,25,-0.151,-0.8181,-0.887452954]

> In the board game "Wyrmspan", how much points is a dragon?

“Useless” words

> In the board game "Wyrmspan", how much points is a dragon?

board game

how much

> In the board game "Wyrmspan", how much points is a dragon?

“Useless” words

- board game
- how much

Common words

- points
- dragon

> In the board game "Wyrmspan", how much points is a dragon?

“Useless” words

- board game
- how much

Common words

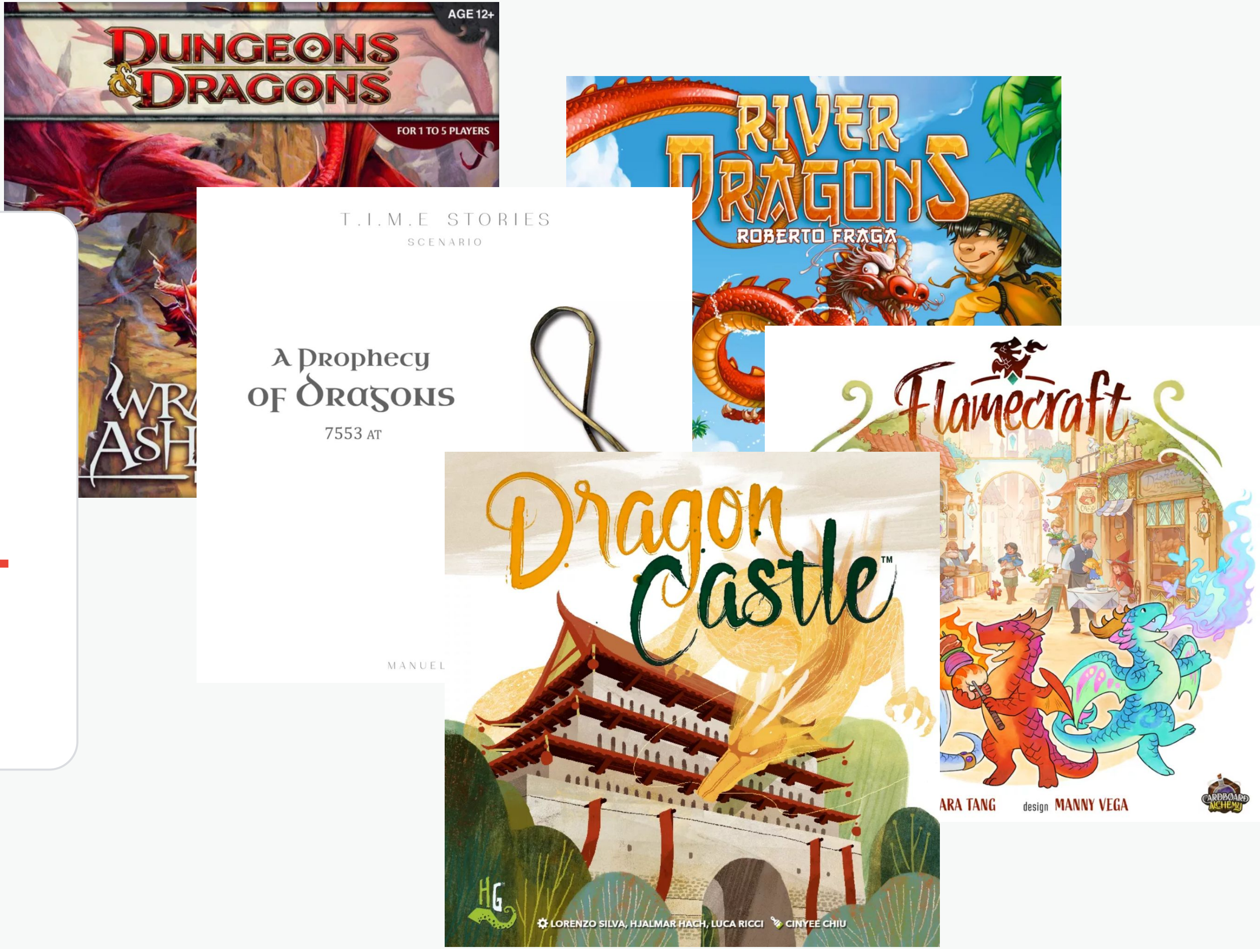
- points
- dragon

Important words

- “Wyrmspan”

> In the board game "Wyrmspan", how much points is a dragon?

> In the board game "Wyrmspan", how much points is a dragon?

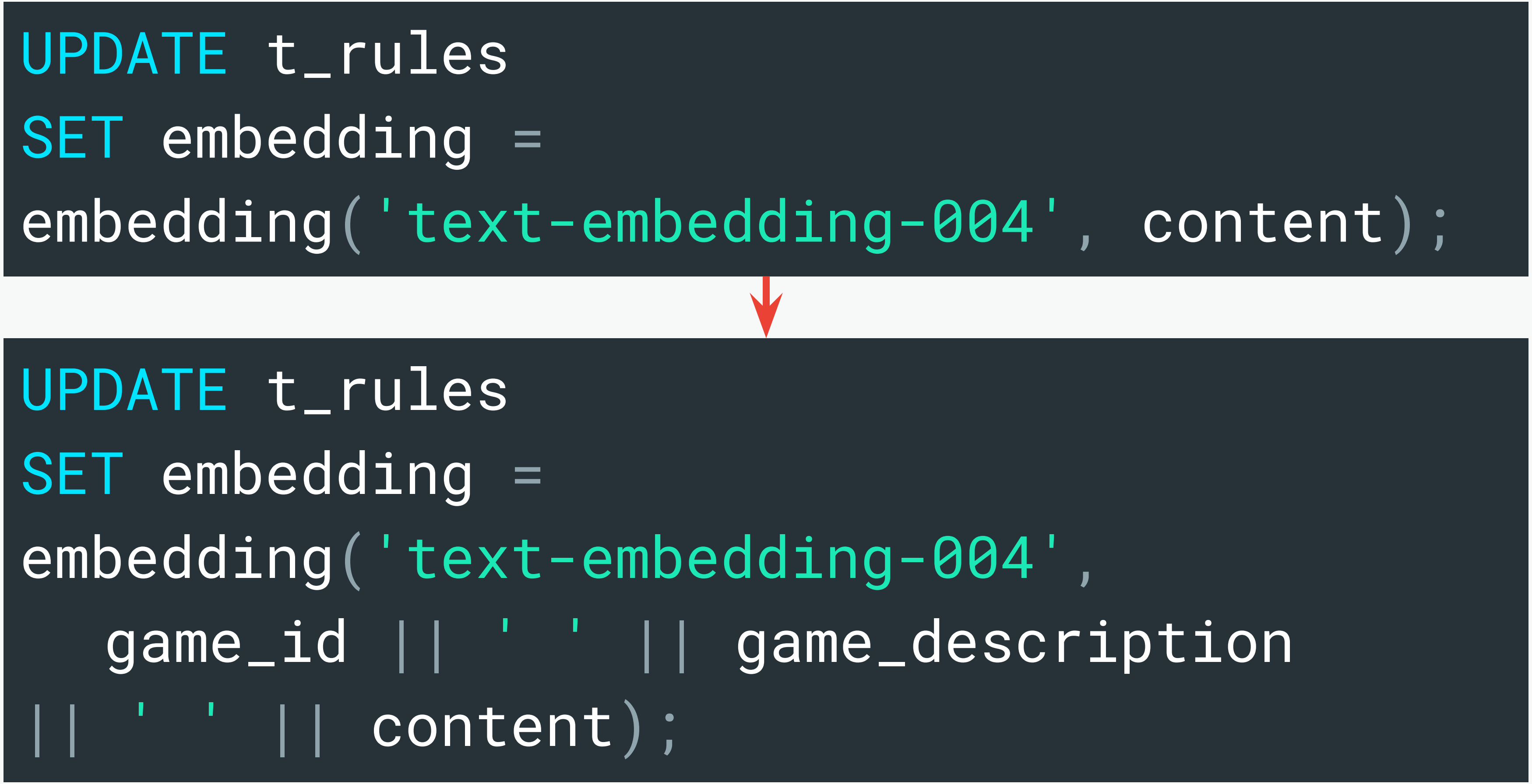


```
UPDATE t_rules
SET embedding =
embedding('text-embedding-004', content);
```

```
UPDATE t_rules
SET embedding =
embedding('text-embedding-004', content);
```

↓

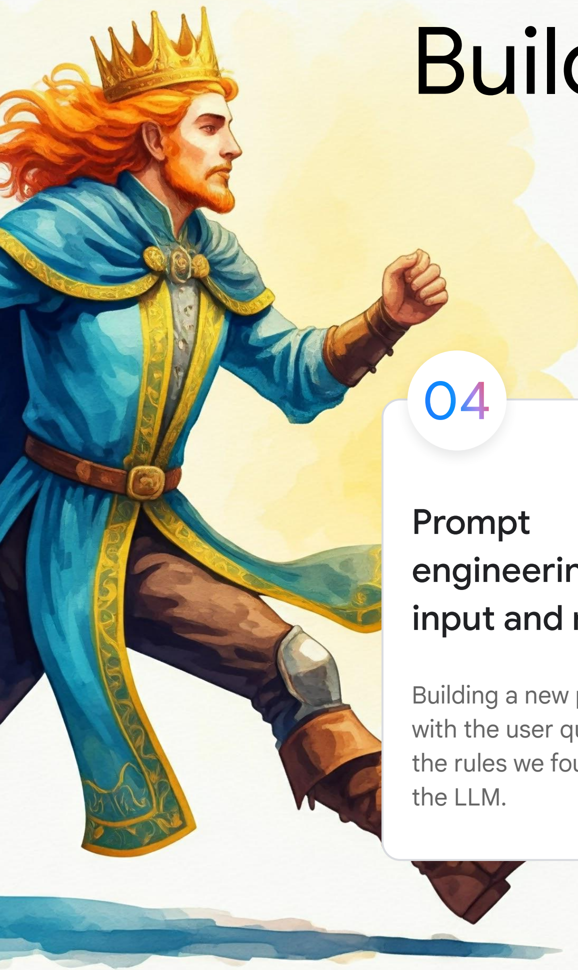
```
UPDATE t_rules
SET embedding =
embedding('text-embedding-004',
   game_id || ' ' || content);
```



```sql
UPDATE t_rules
SET embedding =
embedding('text-embedding-004', content);
```

```sql
UPDATE t_rules
SET embedding =
embedding('text-embedding-004',
  game_id || ' ' || game_description
|| ' ' || content);
```

Building a board game chatbot with Postgres and RAG

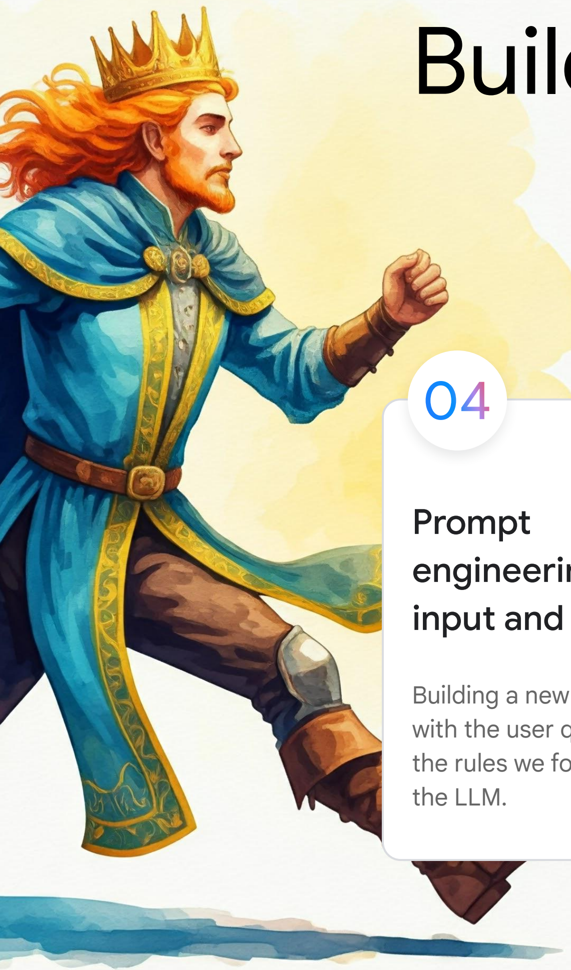


04

Prompt engineering with input and rules

Building a new prompt with the user question and the rules we found to feed the LLM.

Building a board game chatbot with Postgres and RAG



04

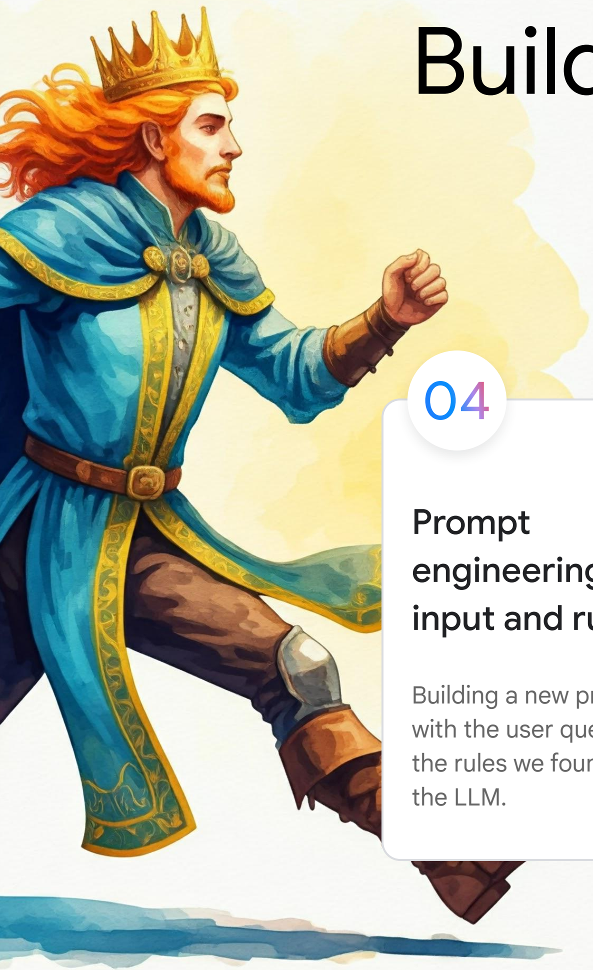
Prompt engineering with input and rules

Building a new prompt with the user question and the rules we found to feed the LLM.

“

In the board game "Castle Combo", how many keys should I start with?

Building a board game chatbot with Postgres and RAG



04

Prompt engineering with input and rules

Building a new prompt with the user question and the rules we found to feed the LLM.

> In the board game "Castle Combo", how many keys should I start with?

> In the board game "Castle Combo", how many keys should I start with?
>
> Here's are the rules of the game:
>
> - Rule n°1: During the setup phase, each player receive 2 keys

That’s no secret, LLMs can …

That’s no secret, LLMs can ...

- Be wrong
- Hallucinate
- Be convinced by nonsense
- Be biased

But that’s mostly our fault.

But that's mostly our fault.

Best practices to improve accuracy:

Define the task clearly

Specify the desired output

Few-shot learning

Show, don't just tell

Persona and role-playing

4

Prompt engineering with input and rules

- Instruction
- Example(s) (x-shot)
- Context
- Input

Instruction

Example(s) (x-shot)

Context

Input

> You are an assistant for question-answering tasks, specialized in board games. Use the following pieces of retrieved context (extract from rules book) to answer the question. If you don't know the answer, say that you don't know. For each answer, provide the extract from the rule book and keep your answer concise.

- Instruction
- Example(s) (x-shot)
- Context
- Input

You are an assistant for question-answering tasks, specialized in board games. Use the following pieces of retrieved context (extract from rules book) to answer the question. If you don't know the answer, say that you don't know. For each answer, provide the extract from the rule book and keep your answer concise.

Instruction

Example(s) (x-shot)

Context

Input

Here is an example:
Question: In the board game Monopoly, how do you get out of jail?
Answer: In the board game Monopoly, you can get out of jail by rolling double,
using a Get out of jail card or pay a $50 fine.
Here is the corresponding rule:
(...)

Instruction

Example(s) (x-shot)

Context

Input

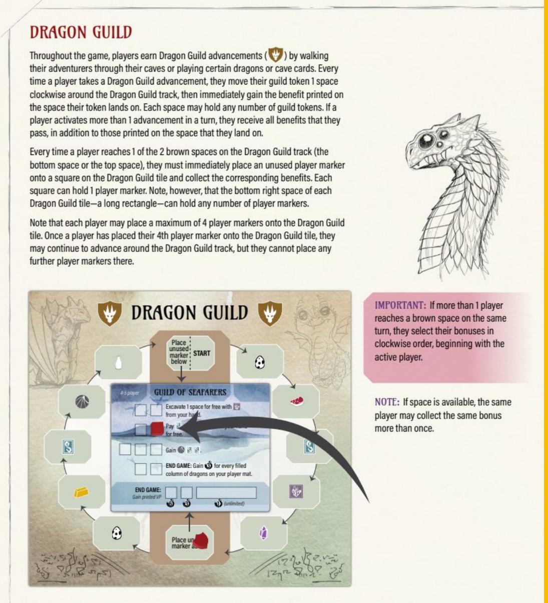
Here is an example:
Question: In the board game Monopoly, how do you get out of jail?
Answer: In the board game Monopoly, you can get out of jail by rolling double,
using a Get out of jail card or pay a $50 fine.
Here is the corresponding rule:
(...)

Instruction

Example(s) (x-shot)

Context

Input



DRAGON GUILD

Throughout the game, players earn Dragon Guild advancements (🛡) by walking their adventurers through their caves or playing certain dragons or cave cards. Every time a player takes a Dragon Guild advancement, they move their guild token 1 space clockwise around the Dragon Guild track, then immediately gain the benefit printed on the space their token lands on. Each space may hold any number of guild tokens. If a player activates more than 1 advancement in a turn, they receive all benefits that they pass, in addition to those printed on the space that they land on.

Every time a player reaches 1 of the 2 brown spaces on the Dragon Guild track (the bottom space or the top space), they must immediately place an unused player marker onto a square on the Dragon Guild tile and collect the corresponding benefits. Each square can hold 1 player marker. Note, however, that the bottom right space of each Dragon Guild tile—a long rectangle—can hold any number of player markers.

Note that each player may place a maximum of 4 player markers onto the Dragon Guild tile. Once a player has placed their 4th player marker onto the Dragon Guild tile, they may continue to advance around the Dragon Guild track, but they cannot place any further player markers there.

IMPORTANT: If more than 1 player reaches a brown space on the same turn, they select their bonuses in clockwise order, beginning with the active player.

NOTE: If space is available, the same player may collect the same bonus more than once.

Instruction

Example(s) (x-shot)

Context

Input

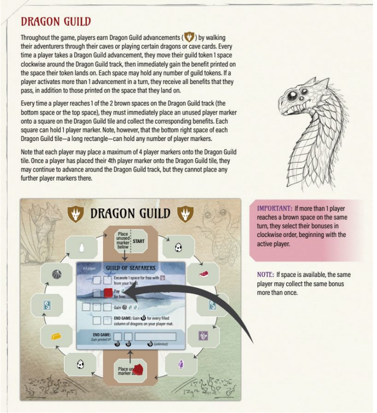
In the board game “Wyrmspan”, how many time can I place a marker on the dragon guild?

You are an assistant for question-answering tasks, specialized in board games.
Use the following pieces of retrieved context (extract from rules book) to answer the question. If you don't know the answer, say that you don't know. For each answer, provide the extract from the rule book and keep your answer concise.

Here is an example:

Question: In the board game Monopoly, how do you get out of jail?

Answer: In the board game Monopoly, you can get out of jail by rolling double, using a Get out of jail card or pay a $50 fine. Here is the corresponding rule: You get out of Jail by.. .(I) throwing doubles on any of your next three turns; if you succeed in doing this you immediately move forward the number of spaces shown by your doubles throw; even though you had thrown doubles, you do not take another turn; (2) using the Get Out of Jail Free card if you have it; (3) purchasing the Get Out of Jail Free card from another player and playing it; (4) paying a fine of $50 before you roll the dice on either of your next two turns.



DRAGON GUILD

Throughout the game, players earn Dragon Guild advancements (🛡) by walking their adventurers through their caves or playing certain dragons or cave cards. Every time a player takes a Dragon Guild advancement, they move their guild token 1 space clockwise around the Dragon Guild track, then immediately gain the benefit printed on the space their token lands on. Each space may hold any number of guild tokens. If a player activates more than 1 advancement in a turn, they receive all benefits that they pass, in addition to those printed on the space that they land on.

Every time a player reaches 1 of the 2 brown spaces on the Dragon Guild track (the bottom space or the top space), they must immediately place an unused player marker onto a square on the Dragon Guild tile and collect the corresponding benefits. Each square can hold 1 player marker. Note, however, that the bottom right space of each Dragon Guild tile—a long rectangle—can hold any number of player markers.

Note that each player may place a maximum of 4 player markers onto the Dragon Guild tile. Once a player has placed their 4th player marker onto the Dragon Guild tile, they may continue to advance around the Dragon Guild track, but they cannot place any further player markers there.

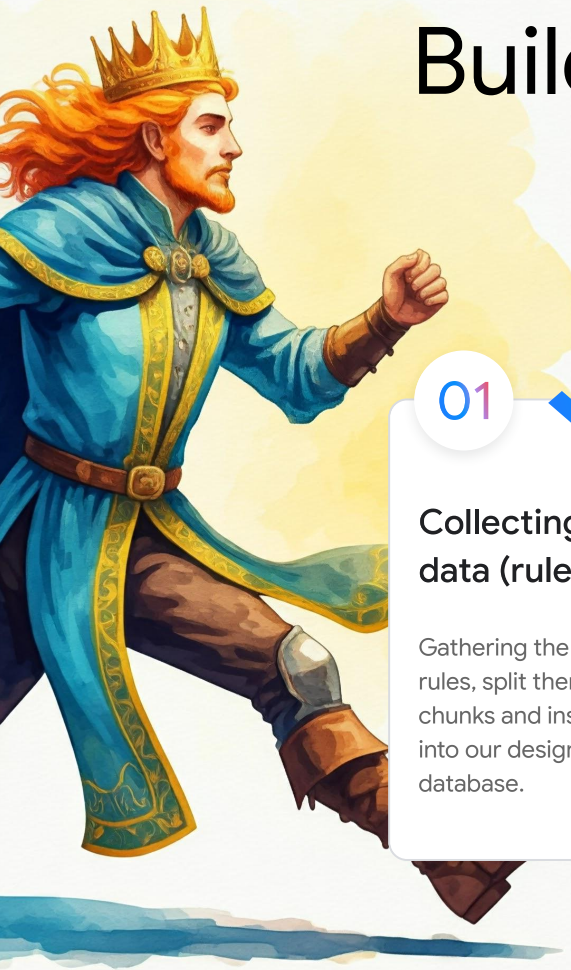
DRAGON GUILD
Place unused marker below
START
4-5 player
GUILD OF SEAFARERS
Excavate 1 space for free with
from your hand.
Pay
for free.
Gain
END GAME: Gain for every filled column of dragons on your player mat.
END GAME:
Gain printed VP
5
3
1 (unlimited)
Place unused marker above

IMPORTANT: If more than 1 player reaches a brown space on the same turn, they select their bonuses in clockwise order, beginning with the active player.

NOTE: If space is available, the same player may collect the same bonus more than once.

In the board game “Wyrmspan”, how many time can I place a marker on the dragon guild?

Building a board game chatbot with Postgres and RAG



01 ✓

Collecting the data (rules)

Gathering the board game rules, split them into chunks and insert them into our designed Postgres database.

02 ✓

Generating the embeddings of the rules

Transforming those texts into usable vectors for similarity searches.

03 ✓

Finding the rules using similarity searches

Explore similarity search with Postgres and pgvector to find the most appropriate rules.

04 ✓

Prompt engineering with input and rules

Building a new prompt with the user question and the rules we found to feed the LLM.

Let’s make it real
with a live demo.

It works!

Now let's make it better.

It works!

Now let's make it better.

1. Find out what we need to improve
2. Indexation, of course.
3. Vector quantization
4. Hybrid search

Find out what we need to improve.

Find out what we need to improve.

Needed time for the chatbot to answer ⚡

Accuracy of answers ✓

Find out what we need to improve.

- Needed time for the chatbot to answer
  - LLM inference time
  - Postgres query performance
  - Indexation method
- Accuracy of answers

Find out what we need to improve.

Needed time for the chatbot to answer

- LLM inference time
- Postgres query performance
- Indexation method

Accuracy of answers ✓

- Prompt engineering
- Embedding model choice
- Chunking strategy

Find out what we need to improve.

Needed time for the chatbot to answer

- LLM inference time
- Postgres query performance
- Indexation method

Accuracy of answers

- Prompt engineering
- Embedding model choice
- Chunking strategy

Indexation, of course.

Indexation, of course.

kNN vs aNN

Accuracy vs perf

Indexation, of course.

kNN vs aNN

Accuracy vs perf

k-NN for the **best result** possible

For certain searches, you will want to find the best possible result, no matter (almost) how long it will take.

In Postgres, this will result in FULL SEQ SCAN of your table.

Indexation, of course.

kNN vs aNN
Accuracy vs perf

k-NN for the **best** result possible

For certain searches, you will want to find the best possible result, no matter (almost) how long it will take.

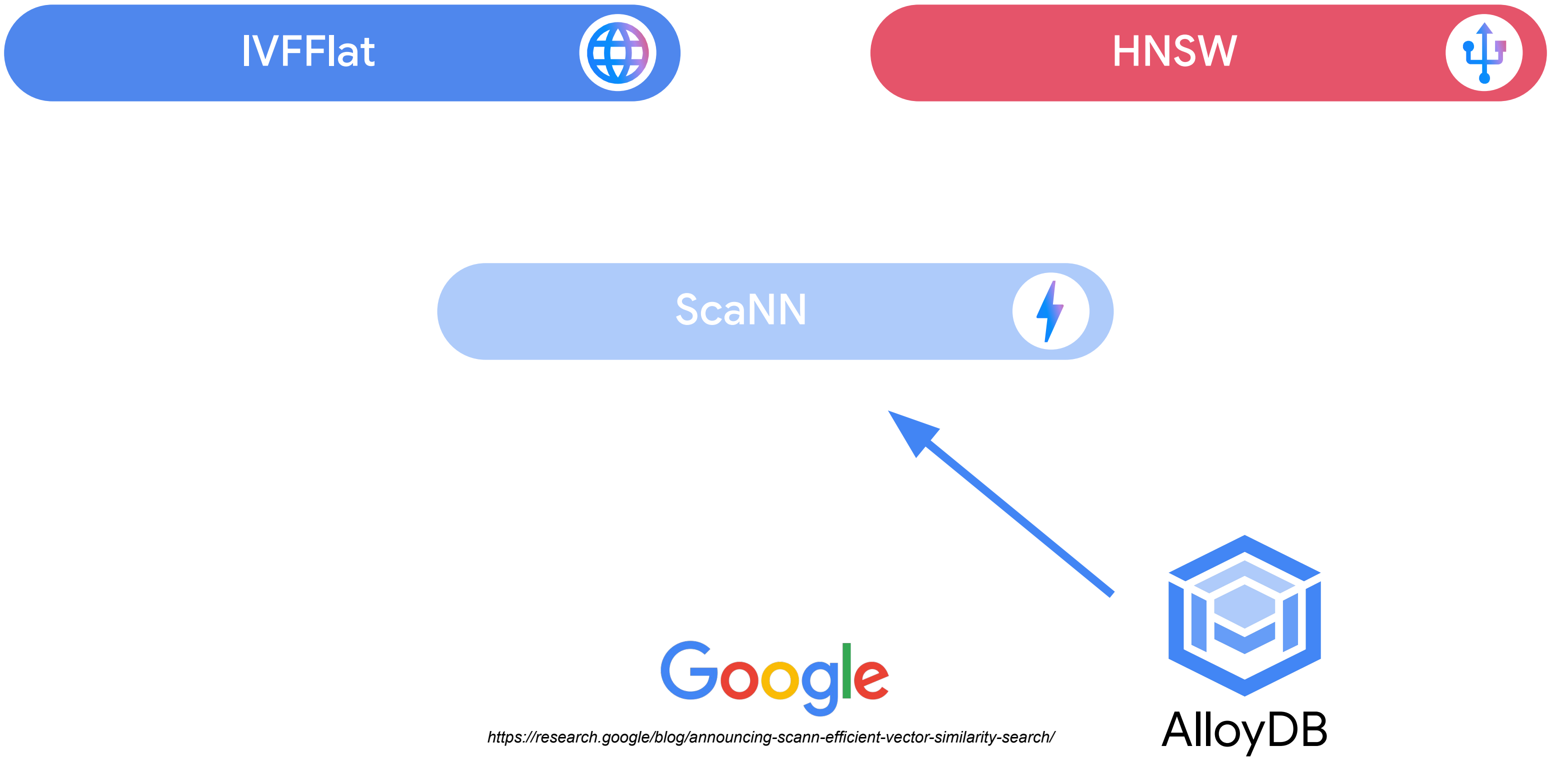
In Postgres, this will result in FULL SEQ SCAN of your table.

A-NN for the **fastest** result possible

And sometimes you will need to prioritize speed over precision, this is when you will use aNN.

Leverage the different index methods offered by pgvector and AlloyDB

Indexation, of course.



IVFFlat

HNSW

ScaNN

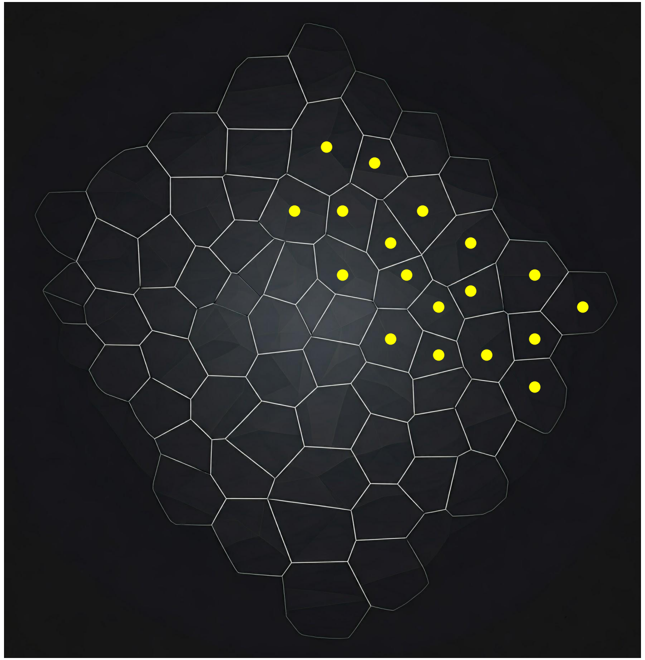
Google

https://research.google/blog/announcing-scann-efficient-vector-similarity-search/

AlloyDB

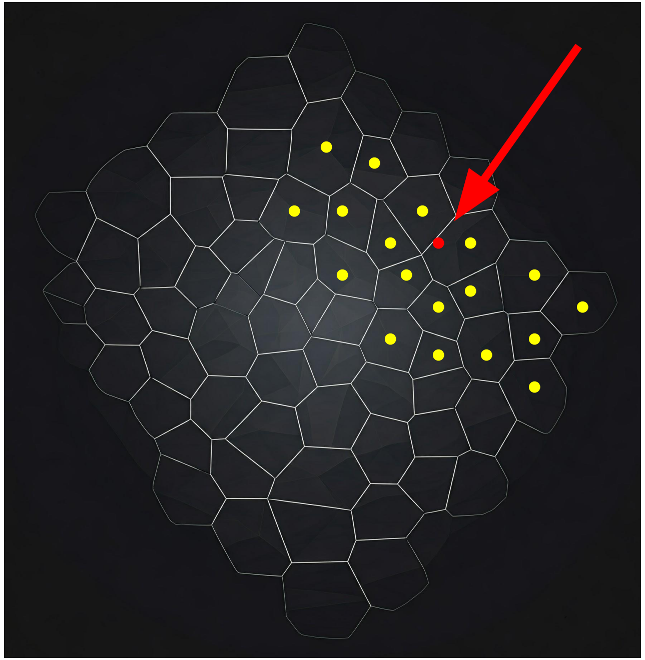
Indexation, of course.

IVFFlat



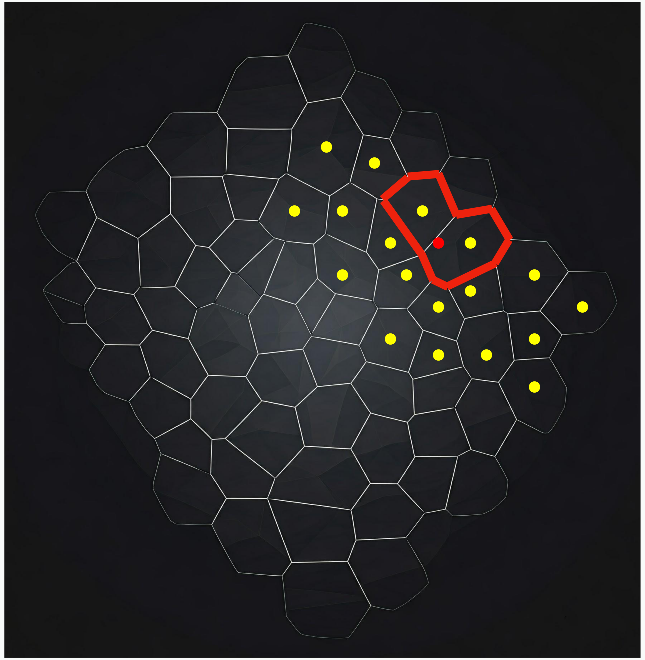
Indexation, of course.

IVFFlat



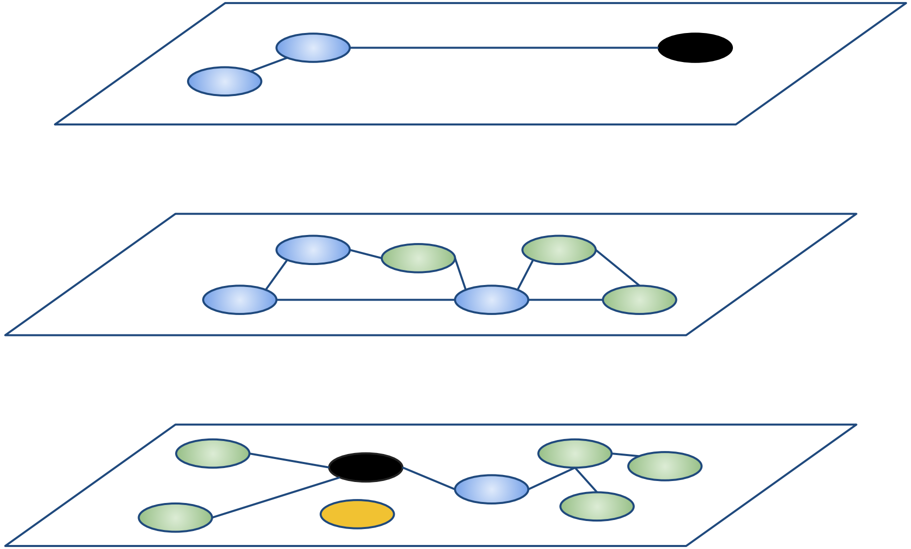
Indexation, of course.

IVFFlat



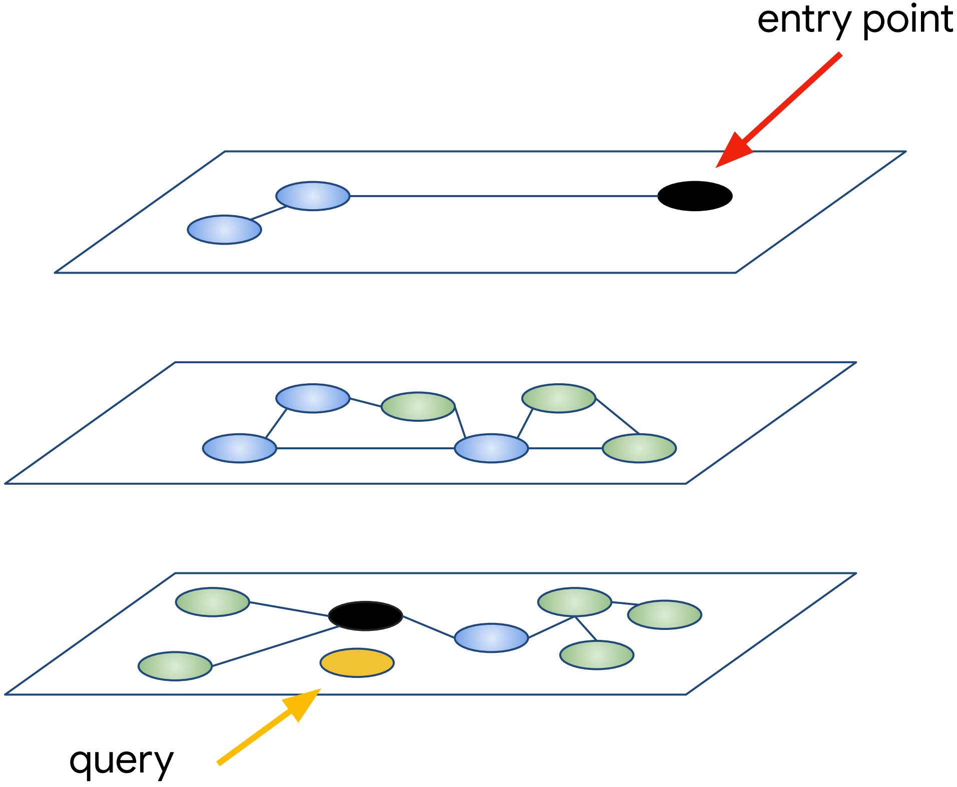
Indexation, of course.

HNSW



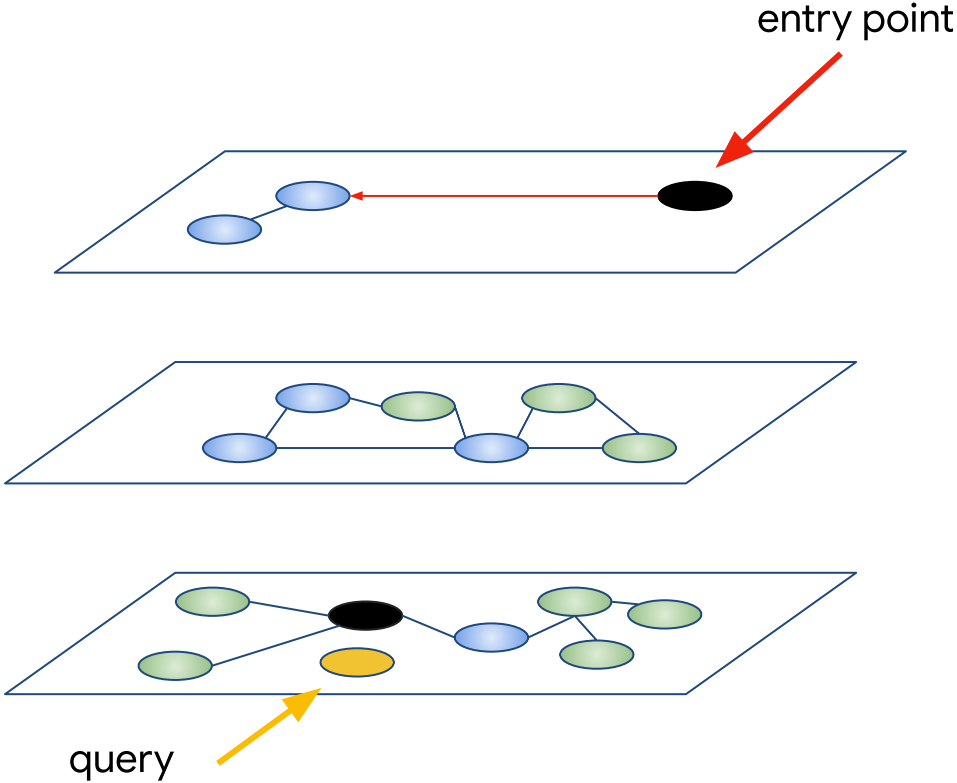
Indexation, of course.

HNSW



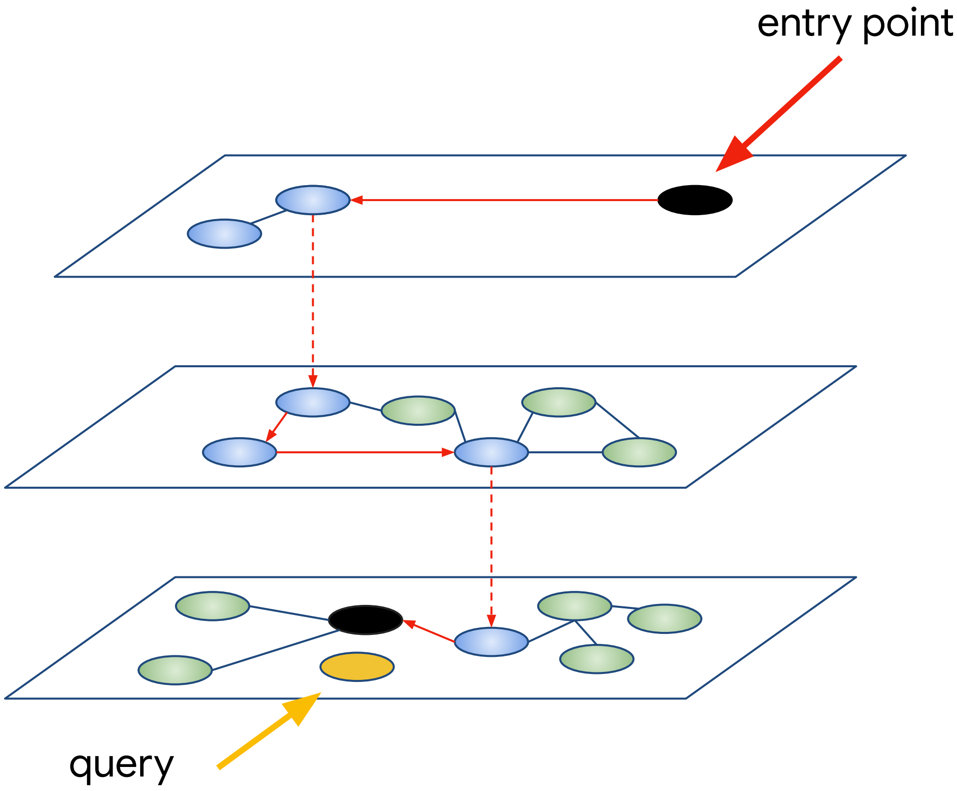
Indexation, of course.

HNSW



Indexation, of course.

HNSW



Vector quantization

Vector quantization

> Vector quantization is a technique to reduce the size and complexity of vector data.

Vector quantization

> Vector quantization is a technique to reduce the size and complexity of vector data.

Why?

- Reduce table/index sizes
- Speed up searches
- Store vectors with more dimensions

Vector quantization

> Vector quantization is a technique to reduce the size and complexity of vector data.

Why?

- Reduce table/index sizes
- Speed up searches
- Store vectors with more dimensions

How?

- Scalar quantization
- Binary quantization

Vector quantization

Scalar quantization

- Reducing the size of a dimension to a smaller type
- eg: going from float32 to float16
- eg: 0.12345679 ⟶ 0.1235

halfvec
(up to 4k dimensions)

Vector quantization

Scalar quantization

- Reducing the size of a dimension to a smaller type
- eg: going from float32 to float16
- eg: 0.12345679 ⟶ 0.1235

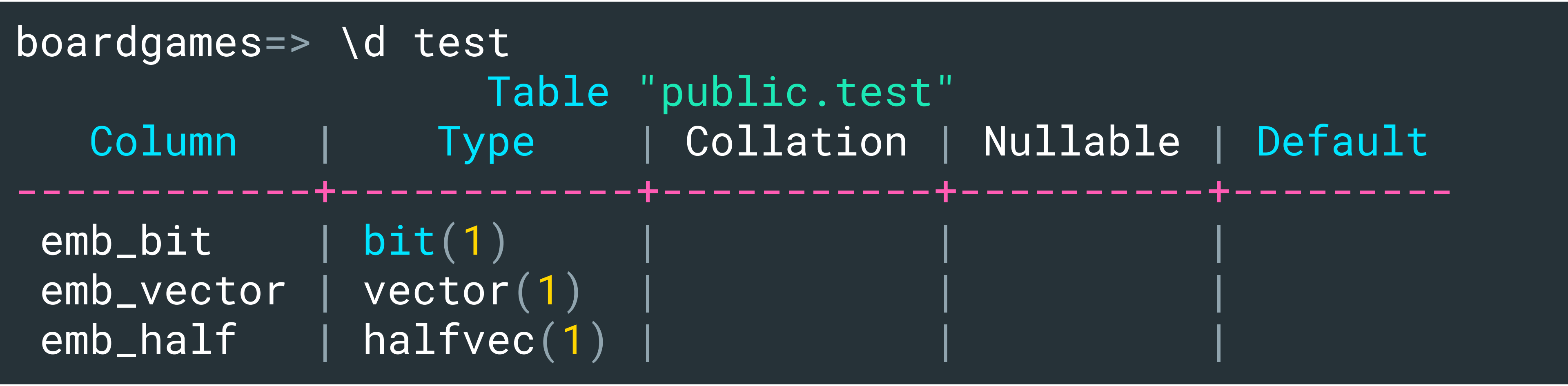
halfvec
(up to 4k dimensions)

Binary quantization

- Each dimension is replaced by 0 or 1
- Aggressive compression, very fast searches
- eg: -0.0063241287 ⟶ 0

bit
(up to 64k dimensions)

Vector quantization



```
boardgames=> \d test
                   Table "public.test"
   Column   |    Type    | Collation | Nullable | Default
------------+------------+-----------+----------+---------
 emb_bit    | bit(1)     |           |          |
 emb_vector | vector(1)  |           |          |
 emb_half   | halfvec(1) |           |          |
```

Vector quantization

```
insert into test (emb_bit, emb_vector, emb_half)
values (binary_quantize('[-0.0063241287]'),
'[-0.0063241287]','[-0.0063241287]');
```

Vector quantization

```
insert into test (emb_bit, emb_vector, emb_half)
values (binary_quantize('[-0.0063241287]'),
'[-0.0063241287]','[-0.0063241287]');
```

```
boardgames=> select * from test ;
-[ RECORD 1 ]----------------
emb_bit    | 0
emb_vector | [-0.0063241287]
emb_half   | [-0.0063247]
```

Vector quantization

Benefits

- Reduced disk/memory space
- Faster queries
- Faster index build

Vector quantization

Benefits

- Reduced disk/memory space
- Faster queries
- Faster index build

Trade-off

- Decreased accuracy
- Loss of precision
- Approximation

Vector quantization

Benefits

- Reduced disk/memory space
- Faster queries
- Faster index build

Trade-off

- Decreased accuracy
- Loss of precision
- Approximation

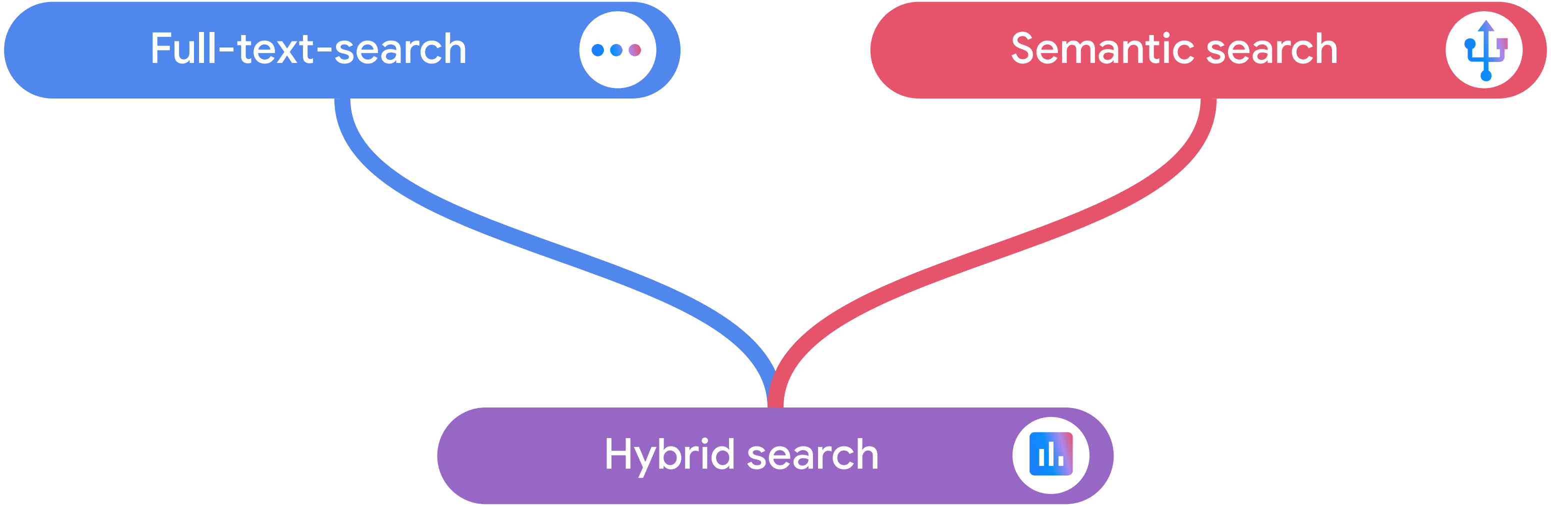
You can mitigate those! →

Hybrid search

Hybrid search

Semantic search

Hybrid search



Hybrid search

01 Query data with **Full-text-search** and rank each result

```
SELECT id,
    substr(content, 0, 20) AS RULE,
    rank() OVER (
                ORDER BY ts_rank_cd(to_tsvector(content),
plainto_tsquery('keys')) DESC) AS rank
FROM rules
WHERE plainto_tsquery('english', 'keys') @@ to_tsvector('english', content)
AND game_id = 594
ORDER BY rank
LIMIT 50
```

Hybrid search

02 Query data with **Semantic search** and rank each result

```
SELECT id,
    substr(content, 0, 20) AS RULE,
    rank() OVER (ORDER BY embedding('textembedding-gecko@003',
$1)::vector <=> content_embedding) AS rank
FROM rules
ORDER BY embedding('textembedding-gecko@003', $1)::vector <=>
content_embedding
LIMIT 50
```

Hybrid search

03

Combine both results (join) and sum scores

```
WITH semantic_search AS
  (...),
     keyword_search AS
  (...)
SELECT COALESCE(semantic_search.id, keyword_search.id) AS id,
       COALESCE(semantic_search.rule, keyword_search.rule) AS RULE,
       COALESCE(1.0 / (50 + semantic_search.rank), 0.0) + COALESCE(1.0 / (50 +
keyword_search.rank), 0.0) AS score,
FROM semantic_search
FULL OUTER JOIN keyword_search ON semantic_search.id = keyword_search.id
ORDER BY score DESC
LIMIT 5 ;
```

Hybrid search

03 Combine both results (join) and sum scores

reciprocal ranked fusion (RRF)

```
WITH semantic_search AS
  (...),
     keyword_search AS
  (...)
SELECT COALESCE(semantic_search.id, keyword_search.id) AS id,
       COALESCE(semantic_search.rule, keyword_search.rule) AS RULE,
       COALESCE(1.0 / (50 + semantic_search.rank), 0.0) + COALESCE(1.0 / (50 +
keyword_search.rank), 0.0) AS score,
FROM semantic_search
FULL OUTER JOIN keyword_search ON semantic_search.id = keyword_search.id
ORDER BY score DESC
LIMIT 5 ;
```

Hybrid search

04 Best score gives you the most accurate answer thanks to **Hybrid search**

```
id  |        rule         |         score
-----+---------------------+------------------------
 232 | # Setup            +| 0.03847576766555678875
     | Shuffle the         |
 234 | # ON yOUR TURN     +| 0.03812636165577342048
     | When                |
 247 | # FINAL SCORE      +| 0.03778966131907308378
     | Addup               |
 246 | # Cards with "Purse | 0.03704974271012006861
 233 | # Gameflow         +| 0.01923076923076923077
     | Starting            |
(5 rows)
```

Hybrid search

01 Query data with **Full-text-search** and rank each result

02 Query data with **Semantic search** and rank each result

03 Combine both results (join) and sum scores

04 Best score gives you the most accurate answer thanks to **Hybrid search**

https://jkatz05.com/post/postgres/hybrid-search-postgres-pgvector/

LangChain
+
Google Cloud

https://cloud.google.com/sql/docs/postgres/langchain

Few takeaways before you go

Vector search offers infinite possibilities

Semantic search is a new powerful tool that can be combined with all your current knowledge and data.

Prompt engineering is a big deal

In your daily life or for RAG, using a good prompt is crucial. Enriching it with real and clean data is as important.

Postgres will always be there for you

Truly open source, strong and reliable, always improving, powerful community. You will never fail by choosing Postgres.

LangChain
+
Google Cloud

https://cloud.google.com/sql/docs/postgres/langchain

Thank you.

Few takeaways before you go

Matt Cornillon

Customer Engineer, DB specialist

Google Cloud

Vector search offers infinite possibilities

Semantic search is a new powerful tool that can be combined with all your current knowledge and data.

Prompt engineering is a big deal

In your daily life or for RAG, using a good prompt is crucial. Enriching it with real and clean data is as important.

Postgres will always be there for you

Truly open source, strong and reliable, always improving, powerful community. You will never fail by choosing Postgres.