



CONF42

LamRAG

From data to constructive insights
using Amazon Bedrock



About Speaker

Sandeep Kumar P

Linkedin / GitHub / X - sandykumar93
AntStack - sandeep@antstack.io

Principal Solutions Architect @AntStack
AWS Community Builder - AI

AWS SA Professional Certified
Serverless Applications' Enthusiast





Problem Statement

Multiple projects, multiple project managers, different sprint schedules, different feedback collection mechanisms

Everybody gets 10/10 ratings



Feedbackly

A system to manage projects, store and retrieve feedbacks given to users for every sprint.

But what about the 10/10 ratings?
Feedbackly meets Amazon Bedrock



L1

Bedrock Playground



I need to classify the provided feedback into 3 categories which are Positive Energy, Productivity, and Reliability and provide a rating between 1 to 5. 1 being the lowest and 5 being the highest. Be stringent with the ratings.

Here's the feedback - Person X has been productive and completed the tasks as expected.

[Explore & Learn](#) [Build & Test](#)

Foundation models

Amazon Bedrock supports foundation models from industry-leading providers. Choose the model that is best suited to achieving your unique goals.

 **AI21 labs**
Jurassic-2 series
By AI21 Labs

 **Titan**
By Amazon

 **Claude**
By Anthropic

 **Command**
By Cohere

 **Llama**
By Meta

 **Mistral**
By Mistral AI

 **Stable Diffusion**
By Stability AI

Spotlight

ANTHROPIC

Anthropic's Claude 3 family of models – Haiku, Sonnet, and Opus – allow customers to choose the exact combination of intelligence, speed, and cost that suits their business needs. All of the models can process images and return text outputs, and feature a 200K context window.

[Open in chat playground](#)

Playgrounds



Chat

Easily experiment on a vast range of language processing tasks in a turn-by-turn interface. You can try out various pre-trained models.

[Open chat playground](#)

Text

Experiment using fast iterations on a vast range of language processing tasks, trying out various pre-trained models. In the playground, enter a text prompt to get started.

[Open text playground](#)

Image

Easily generate compelling images by providing text prompts to pre-trained models. In the playground, enter a text prompt to get started.

[Open image playground](#)

Use cases example

Amazon Bedrock supports many genAI use cases such as summarization, Q&A, and image generation. Explore the ways FMs can support your use cases.

[Browse all examples](#)

Chat playground Info[Load examples](#) Compare mode

Llama 3.1 405B Instruct v1 | On-demand throughput



I think there's a small typo in the feedback. I'll assume it's meant to say "has been productive" instead of "has been productivity".

Based on the provided feedback, I would classify it into the following categories with the corresponding ratings:

1. Positive energy: 1/5

The feedback does not mention anything about Person X's energy, attitude, or enthusiasm. It is a neutral statement that focuses solely on their productivity and task completion.

2. Productivity: 3/5

The feedback states that Person X has been "productive" and completed tasks "as expected", which suggests that they are meeting expectations. However, it doesn't mention anything about exceeding expectations, delivering high-quality results, or taking initiative.

3. Reliability: 3/5

The feedback implies that Person X is reliable, as they have completed tasks "as expected". This suggests that they are able to deliver results consistently and meet expectations. However, the

A Claude 3.5 Sonnet v1 | On-demand throughput
[Change](#)**1. Positive Energy: 1/5**

There is no mention of enthusiasm, motivation, or positive attitude in the feedback. The statement is purely factual and doesn't indicate any particular energy level.

2. Productivity: 3/5

The feedback explicitly states that Person X has been productive (assuming "productivity" was meant to be "productive") and completed tasks as expected. However, to be stringent, a score of 3 is given because while they met expectations, there's no indication of exceeding them or showing exceptional productivity.

3. Reliability: 3/5

Completing tasks as expected suggests a level of reliability. However, the feedback doesn't mention consistency over time, meeting deadlines, or going above and beyond. To be stringent, a middle score is given for meeting, but not exceeding, expectations.

Overall, the feedback is positive but basic. It indicates that Person X is meeting job requirements without any exceptional performance or areas of concern. The ratings reflect a stringent interpretation, acknowledging that the person is doing their job adequately but not

Command R v1 | On-demand throughput
[Change](#)

The feedback provided can be classified and rated as follows:

Positive Energy: 3 - The feedback isn't overly enthusiastic, but it's moderate in its praise, hence a 3 rating for positive energy.

Productivity: 5 - The phrase "productivity" and the statement that Person X completed tasks as expected both indicate a high level of productivity, earning the highest rating.

Reliability: 4 - Task completion as expected suggests reliability and consistency, which rates highly, just shy of an absolute consistent performance.

Being stringent, these ratings reflect a solid performance with room for further improvement.



I need to classify the provided feedback into 3 categories - Positive energy, productivity, and reliability and provide a rating between 1 to 5. 1 being the lowest and 5 being the highest. Be stringent in the ratings. Here's the feedback - Person X has been productive and completed the tasks as expected.



Run

[Choose files](#)

The supported file types are .csv, .doc, .docx, .html, .md, .pdf, .txt, .xls, .xlsx, .gif, .jpeg, .png, .webp. Maximum 5 files, 4.5 MB each.

▼ Model metrics

To evaluate models for task specific metrics with custom dataset visit [Model evaluation](#)

[Define metric criteria](#)



1. Positive energy: 1/5

The feedback does not mention anything about Person X's energy, attitude, or enthusiasm. It is a neutral statement that focuses solely on their productivity and task completion.

2. Productivity: 3/5

The feedback states that Person X has been "productive" and completed tasks "as expected", which suggests that they are meeting expectations. However, it doesn't mention anything about exceeding expectations, delivering high-quality results, or taking initiative.

3. Reliability: 3/5

The feedback implies that Person X is reliable, as they have completed tasks "as expected". This suggests that they are able to deliver results consistently and meet expectations. However, the

2. Productivity: 3/5

The feedback explicitly states that Person X has been productive (assuming "productivity" was meant to be "productive") and completed tasks as expected. However, to be stringent, a score of 3 is given because while they met expectations, there's no indication of exceeding them or showing exceptional productivity.

3. Reliability: 3/5

Completing tasks as expected suggests a level of reliability. However, the feedback doesn't mention consistency over time, meeting deadlines, or going above and beyond. To be stringent, a middle score is given for meeting, but not exceeding, expectations.

Overall, the feedback is positive but basic. It indicates that Person X is meeting job requirements without any exceptional performance or areas of concern. The ratings reflect a stringent interpretation, acknowledging that the person is doing their job adequately but not

Positive Energy: 3 - The feedback isn't overly enthusiastic, but it's moderate in its praise, hence a 3 rating for positive energy.

Productivity: 5 - The phrase "productivity" and the statement that Person X completed tasks as expected both indicate a high level of productivity, earning the highest rating.

Reliability: 4 - Task completion as expected suggests reliability and consistency, which rates highly, just shy of an absolute consistent performance.

Being stringent, these ratings reflect a solid performance with room for further improvement.



I need to classify the provided feedback into 3 categories - Positive energy, productivity, and reliability and provide a rating between 1 to 5. 1 being the lowest and 5 being the highest. Be stringent in the ratings. Here's the feedback - Person X has been productive and completed the tasks as expected.



Choose files

The supported file types are .csv, .doc, .docx, .html, .md, .pdf, .txt, .xls, .xlsx, .gif, .jpeg, .png, .webp. Maximum 5 files, 4.5 MB each.

▼ Model metrics

To evaluate models for task specific metrics with custom dataset visit [Model evaluation](#)

[Define metric criteria](#)

Metrics

Llama 3.1 405B Instruct

Claude 3.5 Sonnet

Command R

Overall summary

[Define metric criteria](#)

[Define metric criteria](#)

[Define metric criteria](#)

Latency

16651 ms

4798 ms

1591 ms

Input token count

1161

1347

912

Output token count

200

247

126

Cost

-

-

-



Consistency Issue



L2

Prompt Engineering



You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following categories:

1. Reliability:
2. Productivity:
3. Positive Energy:

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a "comments" key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.

Feedback to be analysed: "[[feedback]]"

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.

Prompt Engineering

● Persona / Role

● Action

● References, positive, and negative cases.

● Variables

● Output Format



You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following categories:

1. Reliability:
2. Productivity:
3. Positive Energy:

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a "comments" key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.

Feedback to be analysed: "[[feedback]]"

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.

Prompt Engineering

- Persona / Role
- Action
- References, positive, and negative cases.
- Variables
- Output Format



You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following categories:

1. Reliability:
2. Productivity:
3. Positive Energy:

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a "comments" key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.

Feedback to be analysed: "{{feedback}}"

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.

Prompt Engineering

- Persona / Role

- Action

- References, positive, and negative cases.

- Variables

- Output Format



You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following categories:

1. Reliability:
2. Productivity:
3. Positive Energy:

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a "comments" key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.

Feedback to be analysed: "[{feedback}]"

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.

Prompt Engineering

- Persona / Role

- Action

- References, positive, and negative cases.

- Variables

- Output Format



Prompt builder: sls-days Info

Delete

Compare variants

Create version

Save draft

Prompt

Provide a text prompt. Create variables by enclosing a name with double curly braces, as in `{{variable}}`. In the Test variables section, you can specify values to replace the variables when testing.

You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following classifications:

1. Reliability.
2. Productivity.
3. Positive Energy.

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories mentioned above.
- Be strict with the ratings.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a comments key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.

Feedback to be analysed: " {{feedback}} "

Output Instructions:- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.

You can add a maximum of 5 variables. When adding variables, valid characters are a-z, A-Z, 0-9 and _ (underscore). The variable name must start with a letter and can have up to 100 characters.

▼ Test variables (1)

The variables you define in the prompt are listed here. You can add values to test the prompt output in the test window. These values are temporary and aren't saved to your prompt.

Variable name

▼ | Variable value - Optional

feedback

Person X has been productive and completed the tasks as expected.

Configurations - Optional Info

Model

AI Claude 3.5 Sonnet v1 | On-demand throughput
Change

Inference parameters Reset

▼ Randomness and diversity

Temperature

1

Top P

0.999

Top K

250

0

500

Test window

- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.

- You may also add a comments key to share your comments for the ratings.

- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.

Feedback to be analysed: " Person X has been productive and completed the tasks as expected. "

Output Instructions:- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.



Here's my analysis of the feedback in the requested JSON format:

```
{  
  "reliability": 3,  
  "productivity": 4,  
  "positiveEnergy": -1,  
  "comments": "The feedback directly mentions productivity and task completion, but doesn't provide information about reliability or positive energy. The productivity rating is high due to the explicit mention of task completion, while reliability is rated average due to the implication of meeting expectations. There's no information to rate positive energy."  
}
```

▶ Run



Compare variants Info

[Exit comparison](#)[Run all](#)

Original prompt

[Save](#)

A Claude 3.5 Sonnet v1 | On-demand throughput

[Change](#)

Prompt message

You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following classifications:

1. Reliability.
2. Productivity.
3. Positive Energy.

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories mentioned above.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a comments key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.
- Be lenient with the ratings.

Feedback to be analysed: " {{feedback}} "

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.

[▶ Run](#)

▼ Test variables (1)

The variables you define in the prompt are listed here. You can add values to test the prompt output in the test window. These values are temporary and aren't saved to your prompt.

Variable name



Variable value - Optional



feedback

Person X has been productive and completed the tasks as expected.

▼ Output

Variant_1

[Replace original prompt](#)

A Claude 3.5 Sonnet v1 | On-demand throughput

[Change](#)

Prompt message

You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following classifications:

1. Reliability.
2. Productivity.
3. Positive Energy.

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories mentioned above.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a comments key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.
- Be strict with the ratings.

Feedback to be analysed: " {{feedback}} "

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.

[▶ Run](#)

▼ Test variables (1)

The variables you define in the prompt are listed here. You can add values to test the prompt output in the test window. These values are temporary and aren't saved to your prompt.

Variable name



Variable value - Optional



feedback

Person X has been productive and completed the tasks as expected.

▼ Output



▼ Output



You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following classifications:

1. Reliability.
2. Productivity.
3. Positive Energy.

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories mentioned above.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a comments key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.
- Be lenient with the ratings.

Feedback to be analysed: "Person X has been productive and completed the tasks as expected."

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.



Here's my analysis of the feedback in the requested JSON format:

```
{  
  "reliability": 4,  
  "productivity": 5,  
  "positiveEnergy": -1,  
  "comments": "The feedback directly mentions productivity and task completion, warranting a high score for productivity. Reliability can be inferred from completing tasks as expected, hence a good score. There's no information about positive energy, so it's rated as -1."  
}
```

▼ Output



You are tasked with evaluating feedback given by a team member to their peer regarding a project collaboration. Your responsibility is to analyze the feedback content and determine appropriate ratings for the following classifications:

1. Reliability.
2. Productivity.
3. Positive Energy.

Task:

- Provide ratings on a scale of 1 to 5 for each of the categories mentioned above.
- If the feedback does not contain enough context for a particular classification, assign a rating of -1 to that category.
- You may also add a comments key to share your comments for the ratings.
- In cases where the feedback is ambiguous or lacks relevant information, rely on your interpretation to assign suitable ratings.
- Be strict with the ratings.

Feedback to be analysed: "Person X has been productive and completed the tasks as expected."

Output Instructions:

- Use the JSON keys "reliability", "productivity", and "positiveEnergy" to structure the output. Insert the corresponding ratings for each classification.
- Construct a JSON object containing the keys and ratings to represent your analysis and always provide the result in this structure.



```
{  
  "reliability": -1,  
  "productivity": 4,  
  "positiveEnergy": -1,  
  "comments": "The feedback primarily focuses on productivity, mentioning task completion as expected. However, there's no specific information about reliability or positive energy. The productivity rating is 4 out of 5, as the person completed tasks as expected, but there's no indication of exceptional performance that would warrant a perfect score."  
}
```



3 months later ...



L3

Vector Databases and RAG



List some red colored fruits

[1, 0, 0.67, ..., 1.0, 7, 4, 8, 0.5]

"fruit" - "Apple",

"properties" -

 "color" - 1.0,

 "sweetness" - 7,

 "sourness" - 4,

 "crunchiness" - 8,

 "shelf_life" - 0.5

"fruit" - "Orange",

"properties" -

 "color" - 0.8,

 "sweetness" - 6,

 "sourness" - 8,

 "crunchiness" - 2,

 "shelf_life" - 1.0

Apple_Vector_Index: [1.0, 7, 4, 8, 0.5]

Orange_Vector_Index: [0.8, 6, 8, 2, 1.0]

"fruit" - "Apple",

Vector DB & RAG

- Vectors

- Dimensions

- Indexes

- Embedding

- RAG



List some red colored fruits

[1, 0, 0.67, ..., 1.0, 7, 4, 8, 0.5]

"fruit" - "Apple",

"properties" -

 "color" - 1.0,

 "sweetness" - 7,

 "sourness" - 4,

 "crunchiness" - 8,

 "shelf_life" - 0.5

"fruit" - "Orange",

"properties" -

 "color" - 0.8,

 "sweetness" - 6,

 "sourness" - 8,

 "crunchiness" - 2,

 "shelf_life" - 1.0

Apple_Vector_Index: [1.0, 7, 4, 8, 0.5]

Orange_Vector_Index: [0.8, 6, 8, 2, 1.0]

"fruit" - "Apple",

Vector DB & RAG

- Vectors

- Dimensions

- Indexes

- Embedding

- RAG



List some red colored fruits

[1, 0, 0.67, ..., 1.0, 7, 4, 8, 0.5]

"fruit" - "Apple",

"properties" -

 "color" - 1.0,

 "sweetness" - 7,

 "sourness" - 4,

 "crunchiness" - 8,

 "shelf_life" - 0.5

"fruit" - "Orange",

"properties" -

 "color" - 0.8,

 "sweetness" - 6,

 "sourness" - 8,

 "crunchiness" - 2,

 "shelf_life" - 1.0

Apple_Vector_Index: [1.0, 7, 4, 8, 0.5]

Orange_Vector_Index: [0.8, 6, 8, 2, 1.0]

"fruit" - "Apple",

Vector DB & RAG

- Vectors

- Dimensions

- Indexes

- Embedding

- RAG



List some red colored fruits

[1, 0, 0.67, ..., 1.0, 7, 4, 8, 0.5]

"fruit" - "Apple",

"properties" -

 "color" - 1.0,

 "sweetness" - 7,

 "sourness" - 4,

 "crunchiness" - 8,

 "shelf_life" - 0.5

"fruit" - "Orange",

"properties" -

 "color" - 0.8,

 "sweetness" - 6,

 "sourness" - 8,

 "crunchiness" - 2,

 "shelf_life" - 1.0

Apple_Vector_Index: [1.0, 7, 4, 8, 0.5]

Orange_Vector_Index: [0.8, 6, 8, 2, 1.0]

"fruit" - "Apple",

Vector DB & RAG

- Vectors

- Dimensions

- Indexes

- Embedding

- RAG



List some red colored fruits

[1, 0, 0.67, ..., 1.0, 7, 4, 8, 0.5]

"fruit" - "Apple",

"properties" -

 "color" - 1.0,

 "sweetness" - 7,

 "sourness" - 4,

 "crunchiness" - 8,

 "shelf_life" - 0.5

"fruit" - "Orange",

"properties" -

 "color" - 0.8,

 "sweetness" - 6,

 "sourness" - 8,

 "crunchiness" - 2,

 "shelf_life" - 1.0

Apple_Vector_Index: [1.0, 7, 4, 8, 0.5]

Orange_Vector_Index: [0.8, 6, 8, 2, 1.0]

"fruit" - "Apple",

Vector DB & RAG

- Vectors

- Dimensions

- Indexes

- Embedding

- RAG



Knowledge Bases

Chat with your document

Chat with your document

[View documentation](#)

Configurations

Model

Claude 3.5 Sonnet v1 | On-demand
[Change](#)

Generation inference parameters [Info](#)

Set values to influence the responses that the model provides when you query your Knowledge Base.

[Reset](#)

^

Chat prompt template [Info](#)

[Edit](#)

^

Defines how the model handles the user prompt. You can edit the default prompt to create a custom one for your use case.

Prompt type: Prefilled (optimized)

1 You are a question answering agent. I will provide you with a set of search results.
The user will provide you with a question. Your job is to answer the user's question using only information from the search results. If the search results do not contain information that can answer the question, ple ...|

Chat with your document

>



Enter your message here



Run



Knowledge Bases

Chat with your document



Knowledge Bases

▶ How it works

Knowledge Bases (4)

Edit

Delete

Test Knowledge Base

Evaluate

Create ▲

Find Knowledge Base

0 match

COK X

Clear filters

Knowledge Base with vector store

Knowledge Base with structured data store

Knowledge Base with Kendra GenAI Index

| Name ▾ | Status ▾ | Type ▾ | Data so... ▾ | Source ... ▾ | Descrip... ▾ | Creatio... ▾ | Last sy... ▾ | Last sync ▾ |

No Knowledge Base

No Knowledge Base to display

Create ▾



- Step 1
 Provide knowledge base details
- Step 2
 Configure data source
- Step 3
 Select embeddings model and configure vector store
- Step 4
 Review and create

Provide knowledge base details

Knowledge base details

Knowledge base name

sls-days-2024-lamraq

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 50 characters.

Knowledge base description - optional

Enter description

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 200 characters.

Choose data source

Select the data source that you want to configure in the next step. You can add up to 5 data sources in a knowledge base.



Amazon S3



Web Crawler - Preview

Object storage service that stores data as objects within buckets.



Confluence - Preview

Web page crawler that extracts content from public web pages you are authorized to crawl.

Third party data sources



Salesforce - Preview



Sharepoint - Preview

Collaborative work-management tool designed for project planning, software development and product management.

Customer relationship management (CRM) tool for managing support, sales, and marketing data.



Sharepoint - Preview

Collaborative web-based service for working on documents, web pages, web sites, lists, and more.

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter.



Data source location

- This AWS account
- Other AWS account

S3 URI

To increase the accuracy and relevance of your responses, add a .metadata.json file containing metadata for your data source to your S3 bucket. [Info](#)

S3 URI

 s3://cok[View](#) [Browse S3](#)

Add customer-managed KMS key for S3 data - *optional*

If you encrypted your S3 data, provide the KMS key here so that Bedrock can decrypt it.

Parsing strategy

Select a parsing option to configure how your data is processed. You can't modify this option after the knowledge base is created.

- Amazon Bedrock default parser

Select this option to only process text from your data. This parser doesn't incur charges.

- Foundation models as a parser

Select this option to process visually rich documents or images. With this option, you can use the default parser prompt or customize it for your use case.

[Pricing details](#)

Chunking strategy

Chunking breaks down the text into smaller segments before embedding. The chunking strategy can't be modified after you create the data source.

Default chunking

Automatically splits text into chunks of about 300 tokens in size, by default. If a document is less than or already 300 tokens, it's not split any further.



► Transformation function - *optional*

► Advanced settings - *optional*

[Add data source](#)

You can add 4 more data source(s).



- Step 1
● Provide knowledge base details
- Step 2
Configure data source
- Step 3
○ Select embeddings model and configure vector store
- Step 4
○ Review and create

Amazon S3 Info

Provide details to connect Amazon Bedrock to your S3 data source.

▼ Data source: sls-days-2024-lamrag-source-1[Delete](#)**Data source name**

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 100 characters.

S3 URI

To increase the accuracy and relevance of your responses, add a .metadata.json file containing metadata for your data source to your S3 bucket. [Info](#)

[View](#)[Browse S3](#)**Default chunking**

Automatically splits text into chunks of about 300 tokens in size, by default. If a document is less than or already 300 tokens, it's not split any further.

Fixed-size chunking

Splits text into your set approximate token size.

Hierarchical chunking

Organizes text chunks (nodes) into hierarchical structures of parent-child relationships. Each child node includes a reference to its parent node.

Semantic chunking

Organizes text chunks or groups of sentences by how semantically similar they are to each other.

No chunking

Suitable for documents that are already pre-processed or text split into separate files without any further chunking necessary.

**No chunking**

Suitable for documents that are already pre-processed or text split into separate files without any further chunking necessary.

**Select Lambda function**

Select an existing Lambda function to customize chunking and document metadata processing. Visit [AWS Lambda](#) to create a new function. Select the refresh button after creating your function.

**Function version**[View](#)[Add data source](#)

You can add 4 more data source(s).

[Cancel](#)[Previous](#)[Next](#)



- Step 1
● Provide knowledge base details
- Step 2
● Configure data source
- Step 3
● Select embeddings model and configure vector store
- Step 4
○ Review and create

Select embeddings model and configure vector store

Embeddings model

Select an embeddings model to convert your data into an embedding. Pricing depends on the model. [Learn more](#)



Titan Text Embeddings v2 [?](#)

By Amazon



Titan Embeddings G1 - Text v1.2 [?](#)

By Amazon



Embed English v3 [?](#)

By Cohere



Embed Multilingual v3 [?](#)

By Cohere

Vector dimensions

Select the vector dimension size for your embeddings model to balance accuracy, cost, and latency. Higher dimensions improves overall accuracy and requires more vector storage. [Learn more](#)

1536



Vector database

Let Amazon create a vector store on your behalf or select a previously created store to allow Bedrock to store, update and manage embeddings. You will be billed directly from the vector store provider. [Learn more](#)

Select how you want to create your vector store.

Quick create a new vector store - *Recommended*

We will create an Amazon OpenSearch Serverless vector store on your behalf. This cost-efficient option is intended only for development and can't be migrated to production workload later. [Learn more](#)

Choose a vector store you have created

Select Amazon OpenSearch Serverless, Amazon Aurora, MongoDB Atlas, Pinecone or Redis Enterprise Cloud and provide field mappings.

Enable redundancy (active replicas) - *optional*

The default configuration has active replicas disabled, which is optimal for development workloads. Enable this option if you want to enable redundant active replicas, which may increase storage costs.

Add customer-managed KMS key for Amazon OpenSearch Serverless vector - *optional*

If you encrypted your OpenSearch data, provide the KMS key here so that Bedrock can decrypt it.



[Embed English v3](#)

By Cohere



[Embed Multilingual v3](#)

By Cohere



► Additional configurations

Vector database

Create a new vector or choose an existing store so that Amazon Bedrock can store, update, and manage embeddings. [Learn more](#)

Vector store creation method

Quick create a new vector store - *Recommended*

Choose one of the vector stores below to create a new vector store.

Choose a vector store you have created

Select Amazon OpenSearch Serverless, Amazon Aurora, Pinecone, Amazon Aurora, MongoDB Atlas, Neptune Analytics (GraphRAG) or Redis Enterprise Cloud and provide field mappings.

Vector store

If you've used a vector store before, we recommend that you continue using it. Otherwise, choose the vector store that best fits your use case.

Amazon OpenSearch Serverless

Select to optimize and provide contextually relevant responses across billions of vectors in milliseconds. Combine search with text-based keywords for hybrid requests. [Learn more](#)

Amazon Aurora PostgreSQL Serverless - *new*

Select to store and index vector embeddings for fast similarity search. Use SQL queries to combine search results with your business data. [Learn more](#)

Amazon Neptune Analytics (GraphRAG) - *Preview*

Select to optimize for high-performance graph analytics and enable graph-based Retrieval Augmented Generation (GraphRAG) solutions. [Learn more](#)

► Additional configurations

Cancel

Previous

Next



- Step 1
Provide knowledge base details
- Step 2
Configure data source
- Step 3
Select embeddings model and configure vector store
- Step 4
Review and create

Select embeddings model and configure vector store

Choose an embeddings model to convert the data that you will provide in the next step, and provide details for a vector data store in which Bedrock can store, manage, and update your embeddings. The embeddings model and vector store cannot be changed after creation of knowledge base.

Vector database

Let Amazon create a vector store on your behalf or select a previously created store to allow Bedrock to store, update and manage embeddings. You will be billed directly from the vector store provider. [Learn more](#)

Select how you want to create your vector store.

Quick create a new vector store - *Recommended*

We will create an Amazon OpenSearch Serverless vector store on your behalf. This cost-efficient option is intended only for development and can't be migrated to production workload later.
[Learn more](#)

Choose a vector store you have created

Select Amazon OpenSearch Serverless, Amazon Aurora, MongoDB Atlas, Pinecone or Redis Enterprise Cloud and provide field mappings.

Select your vector store.

 Vector engine for Amazon OpenSearch Serverless

If you are a first time user, create a vector database by visiting [OpenSearch Service](#)

 Amazon Aurora

If you are a first time user, create a vector database by visiting [RDS Console](#)

 MongoDB Atlas

If you are a first-time user, create a MongoDB Atlas Cluster and Vector Search Index by visiting [MongoDB Atlas](#)

 Pinecone

If you are a first time user, create a vector database by visiting [Pinecone](#)

 Redis Enterprise Cloud

If you are a first time user, create a vector database by visiting [Redis Enterprise Cloud](#)

Collection ARN

Specify ARN of your OpenSearch Serverless Collection

Enter collection ARN. For example: "arn:aws:aoess:us-east-1:149020153069:collection/lagni2uq61hckn6qvole"



sls-days-2024-lamrag

Knowledge base overview

Knowledge base name
sls-days-2024-lamrag

Knowledge base ID
UBSVV3U2II

Knowledge base description
—

Status
Ready

Service Role
AmazonBedrockExecutionRoleForKnowledgeBase_08jf9 []

Created date
August 29, 2024, 22:42 (UTC+05:30)

Test

Delete

Edit

Log Deliveries

Configure log deliveries and event logs in the [Edit](#) page.

Data source (1)

Data sources contain information returned when querying a Knowledge base.

Add

Edit

Delete

Sync

Find data source

< 1 >

Data so...	Status	Data sour...	Account ID	Source Link	Last sync ...	Last sync ...	Chunking ...	Parsing st...	Data dele...
sls-days-2...	Available	S3	96035158...	s3://sls-d...	-	-	No	Default	Delete

Embeddings model

Model
Titan Embeddings G1 - Text v1.2

Vector dimensions
1536

Vector database

Vector database
Vector engine Amazon OpenSearch Serverless

Vector index name
bedrock-knowledge-base-default-index

Collection ARN
arn:aws:aoss:us-west-2:960351580303:collection/atub895u4zwpffp4j100

Vector field name
bedrock-knowledge-base-default-vector

Metadata field name

Generate responses



Select model

One or more data sources have not been synced.

[Go to data sources](#)

Configure your retrieval and responses

To customize the search strategy for your knowledge base, select the configurations icon

Test your knowledge base by running a query to generate responses. To disable response generation and only see retrieved information stored from your vector store, turn off Generate responses above.

Please select a model





Test Knowledge base

Configurations Info

[Copy](#)[X](#) Generate responses[Select model](#)

Search Type

Search Type refers to how retrieval is done on vector store.

 Override default search

Default search chooses the best option between hybrid search and semantic search for your vector store for better accuracy.

Choose your preferred option for searching your vector store:

 Hybrid search (semantic & text)

Combines relevancy scores from semantic and text search to provide greater accuracy.

 Semantic search

Uses vector embeddings to deliver relevant results.

Maximum number of source chunks

Specify the maximum number of retrieved results to return from the vector store.

Enter a number between 1 and 100

Filters Info

Metadata search helps you improve response accuracy and relevancy. Add filters and then run a query to search with metadata.

[X](#)[+ Add Group](#)

Query modifications

 Break down queries

Enabling this allows the Knowledge base to split complex queries into multiple parts to get more relevant responses. This may improve retrieval accuracy.

Please select a model

[Run](#)



Test Knowledge base

Configurations Info

[Copy](#)[X](#) Generate responses[Select model](#)

Search Type

Search Type refers to how retrieval is done on vector store.

 Override default search

Default search chooses the best option between hybrid search and semantic search for your vector store for better accuracy.

Choose your preferred option for searching your vector store:

 Hybrid search (semantic & text)

Combines relevancy scores from semantic and text search to provide greater accuracy.

 Semantic search

Uses vector embeddings to deliver relevant results.

Maximum number of source chunks

Specify the maximum number of retrieved results to return from the vector store.

Enter a number between 1 and 100

Filters Info

Metadata search helps you improve response accuracy and relevancy. Add filters and then run a query to search with metadata.

[X](#)[+ Add Group](#)

Query modifications

 Break down queries

Enabling this allows the Knowledge base to split complex queries into multiple parts to get more relevant responses. This may improve retrieval accuracy.

Please select a model

[Run](#)



Test Knowledge base

Configurations [Info](#)Inference parameters [Info](#)

Set values to influence the responses that the model provides when you query your knowledge base.

Randomness and diversity

Temperature



Top P



Length

Maximum length



Stop sequences

 [Add](#)[Observation](#) Knowledge base prompt template [Info](#)

Defines how the model handles the user prompt. You can edit the default prompt to create a custom one for your use case.

System prompt: Default

[Edit](#)

- 1 You are a question answering agent. I will provide you with a set of search results. The user will provide you with a question. Your job is to answer the user's question using only information from the search results. If the search results do not contain information that can answer the question, ple ...

 Generate responses**Claude 3.5 Sonnet** v1 | On-demand throughput[Change](#)

Test your knowledge base by running a query to generate responses. To disable response generation and only see retrieved information stored from your vector store, turn off Generate responses above.

 Enter your message here



Test Knowledge base

Configurations Info[Copy](#)[X](#)

Search Type

Search Type refers to how retrieval is done on vector store.

 Override default search

Default search chooses the best option between hybrid search and semantic search for your vector store.

Inference parameters Info

Set values to influence the responses that the model provides when you query your knowledge base.

[Reset](#)[▼](#)

Randomness and diversity

Temperature

0

Top P

1

Length

Maximum length

2048

Stop sequences

[Add](#)[Observation X](#)

Maximum number of source chunks

Specify the maximum number of retrieved results to return from the vector store.

Enter a number between 1 and 100

4

 Generate responses

Claude 3.5 Sonnet v1 | On-demand throughput

[Change](#)

"email": "manik.pokhetra@antstack.io", "query": "provide my top 3 points of improvement"



1. Maintain consistency: While the feedback is consistently positive, continuing to maintain this high level of performance across all projects and tasks would be beneficial.
2. Attention to detail: One piece of feedback suggests focusing more on details to strengthen analysis, which could be an area for potential improvement.
3. Balance workload: Given the high praise for reliability and productivity, ensuring a sustainable work-life balance to maintain this level of performance long-term could be important.^[1]

[Show source details >](#)

"email": "sandeep@antstack.io", "query": "provide my top 3 points of improvement"



1. Time management and meeting deadlines: Sandeep has had issues with missed deadlines and inaccuracies in monthly reports, which have led to setbacks in the team's schedule. Improving punctuality and ensuring all information is double-checked before submission is crucial.^[1]
2. Focus and context switching: While Sandeep manages to prioritize and allocate time for multiple tasks, there's a need to improve on context switching. This will help in maintaining concentration and efficiency across various responsibilities.^[2]
3. Increased involvement: Some feedback suggests that more involvement is expected from Sandeep. Enhancing engagement in project activities and team interactions could lead to better overall performance and contribution.^[3]

[Show source details >](#)
Enter your message here[Run](#)



Test Knowledge base

Configurations Info[Copy](#)[X](#)

Search Type

Search Type refers to how retrieval is done on vector store.

 Override default search

Default search chooses the best option between hybrid search and semantic search for your vector store.

Inference parameters Info

Set values to influence the responses that the model provides when you query your knowledge base.

[Reset](#)[▼](#)

Randomness and diversity

Temperature

0

Top P

1

Length

Maximum length

2048

Stop sequences

[Add](#)[Observation](#) [X](#)

Maximum number of source chunks

Specify the maximum number of retrieved results to return from the vector store.

Enter a number between 1 and 100

4

 Generate responses**Claude 3.5 Sonnet v1** | On-demand throughput[Change](#)[X](#)[Edit](#)

"email": "manik.pokhetra@antstack.io", "query": "provide my top 3 points of improvement"



1. Maintain consistency: While the feedback is consistently positive, continuing to maintain this high level of performance across all projects and tasks would be beneficial. 2. Attention to detail: One piece of feedback suggests focusing more on details to strengthen analysis, which could be an area for potential improvement. 3. Balance workload: Given the high praise for reliability and productivity, ensuring a sustainable work-life balance to maintain this level of performance long-term could be important.^[1]

[Show source details >](#)

"email": "sandeep@antstack.io", "query": "provide my top 3 points of improvement"



1. Time management and meeting deadlines: Sandeep has had issues with missed deadlines and inaccuracies in monthly reports, which have led to setbacks in the team's schedule. Improving punctuality and ensuring all information is double-checked before submission is crucial.^[1] 2. Focus and context switching: While Sandeep manages to prioritize and allocate time for multiple tasks, there's a need to improve on context switching. This will help in maintaining concentration and efficiency across various responsibilities.^[2] 3. Increased involvement: Some feedback suggests that more involvement is expected from Sandeep. Enhancing engagement in project activities and team interactions could lead to better overall performance and contribution.^[3]

[Show source details >](#)
Enter your message here[Run](#)



Admins Vs Users



L4

Agents



Agent builder: sls-days-2024-lamrag [Info](#)

[Test](#) [Prepare](#) [Save](#) [Save and exit](#)

Agent details

Agent name

sls-days-2024-lamrag

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 100 characters.

Agent description - optional

demo

The description can have up to 200 characters.

Agent resource role

- Create and use a new service role
- Use an existing service role

AmazonBedrockExecutionRoleForAgents_2BE69VV3T9H

Select model



Anthropic



Claude 3 Haiku



Next-gen AI assistant trained on helpful, honest, and harmless AI systems, Claude can help with summarization, search, creative writing, Q&A, coding, as well as take direction.

Instructions for the Agent

Provide clear and specific instructions for the task the Agent will perform. You can also provide certain style and tone.

Your job is to execute the following flow of events:

1. Receive user input and identify if the user is asking question about themselves or for other users

This instruction must have a minimum of 40 characters.

Memory [Preview](#)

By default, agents only remember information within a single conversation. Enabling memory allows agents to remember information across multiple sessions for a maximum of 30 days. [View memory](#)

Enable memory

Enabling will generate a session summary at the end of each testing session.

- Enabled
- Disabled

Test Agent

[File](#) [Edit](#) [Run](#) >

Using ODT [Change](#)

Prepare the Agent to test the latest changes.

[Prepare](#)

Enter your message here



[Run](#)



Agent builder: sls-days-2024-lamrag [Info](#)

[Test](#) [Prepare](#) [Save](#) [Save and exit](#)

Agent details

Agent name

sls-days-2024-lamrag

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 100 characters.

Agent description - optional

demo

The description can have up to 200 characters.

Agent resource role

- Create and use a new service role
- Use an existing service role

AmazonBedrockExecutionRoleForAgents_2BE69VV3T9H

Select model



Anthropic



Claude 3 Haiku



Next-gen AI assistant trained on helpful, honest, and harmless AI systems, Claude can help with summarization, search, creative writing, Q&A, coding, as well as take direction.

Instructions for the Agent

Provide clear and specific instructions for the task the Agent will perform. You can also provide certain style and tone.

Your job is to execute the following flow of events:

1. Receive user input and identify if the user is asking question about themselves or for other users

This instruction must have a minimum of 40 characters.

Memory [Preview](#)

By default, agents only remember information within a single conversation. Enabling memory allows agents to remember information across multiple sessions for a maximum of 30 days. [View memory](#)

Enable memory

Enabling will generate a session summary at the end of each testing session.

- Enabled
- Disabled

Test Agent

[Test](#) [Prepare](#) [Save](#) [Save and exit](#)

Using ODT [Change](#)

Prepare the Agent to test the latest changes.

[Prepare](#)

Enter your message here



Run



Action groups (0) [Info](#)

Find action groups

Name	Description	State	Last updated
------	-------------	-------	--------------

No action groups
No actions group to display

[Add](#)

Knowledge bases (0) [Info](#)

Find knowledge base

Name	State	Instruction
------	-------	-------------

No knowledge base
No knowledge base to display

[Add](#)

Guardrail details

Guardrail name	Guardrail version
-	-

[Edit](#)

Advanced prompts [Info](#)

Pre-processing Default	Knowledge base response generation Default	Edit
Orchestration Default	Post-processing Default	

Test Agent

[File](#) [Edit](#) [Run](#) >
Using ODT Change

ⓘ Prepare the Agent to test the latest changes.

[Prepare](#)

Enter your message here

[Run](#)



Create Action group

Action group details

Enter Action group name

sls-days-2024-lamrag-action-group

Description - optional

Used for user identification

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). This description can have up to 200 characters.

Action group type

Select what type of action group to create

Define with function details

Specify functions and define parameters as JSON objects that will be associated to the action group invocation.

Define with API schemas

Specify a Lambda or API Gateway and define a schema to specify the APIs that the agent can invoke to carry out its tasks.

Action group invocation

Specify a Lambda function that will be invoked based on the action group identified by the Foundation model during orchestration.

Select how to define the Lambda function

Quick create a new Lambda function - recommended

An Amazon Lambda function will be created in your account on your behalf. No further configurations are necessary.

Select an existing Lambda function

Use an existing Lambda function for this action group.

Return control

Agent responses in the test window will prompt the user for function details to generate a response. No further configurations are necessary.

▼ Action group function 1 : sls-days-2

A function specifies the business logic for this action group using the defined parameters. Create functions using the form builder or via JSON editor. Up to 3 functions can be created per action group.

Name

sls-days-2024-lamrag-action-group-lambda-function

Delete

Table

JSON Editor

Test Agent

Using ODT Change

Prepare the Agent to test the latest changes.

Prepare

Enter your message here



Run





- Use an existing Lambda function for this action group.
- Return control
Agent responses in the test window will prompt the user for function details to generate a response. No further configurations are necessary.

▼ Action group function 1 : sls-days-2

[Delete](#)[Table](#)[JSON Editor](#)**Name****sls-days-2024-lamrag-action-group-lambda-function**

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). This description can have up to 100 characters.

Description - optional**Enter description**

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). This description can have up to 1200 characters.

Enable confirmation of action group function - optional

Request confirmation before user invokes the action group's function. Enabling this may safeguard your end user from malicious prompt injection.

 Enabled Disabled**Parameters (2)**

Parameters allow you do define object relationships within the action group function.

Name ↴	▼ Description ↴	▼ Type ↴	▼ Required ↴	Delete
query_type	self or others	String	False	✖
email	email of the requester	String	False	✖

[Add parameter](#)[Add action group function](#)[Cancel](#)[Create](#)**Test Agent**[🔗 ↻ ⟲ ⟳ ⟷](#)

Using ODT Change

ⓘ Prepare the Agent to test the latest changes.

[Prepare](#)**Enter your message here**[Run](#)



III

Code Test Monitor Configuration Aliases Versions

Code source [Info](#)

Upload from ▾

File Edit Find View Go Tools Window **Test** ▾ DeployGo to Anything (⌘ P)
dummy_lambda.x Environment Var +
sls-days-2024-lamr: dummy_lambda.py

```
3 def lambda_handler(event, context):
4     print(event)
5     admin_users = ["sandeep@antstack.io"]
6     parameters = event["parameters"]
7     output = {
8         "next": "end",
9         "text": "You are not authorised to run this query"
10    }
11
12    if parameters[1]["value"] in admin_users or parameters[0]["value"] == "self":
13        output = {
14            "next": "knowledge_base",
15            "text": "You are authorised to run this query"
16        }
17
18
19    # Execute your business logic here. For more information, refer to: https://docs.aws.amazon.com/bedrock/latest/userguide/agents-lambda.html
20    responseBody = { "TEXT": { "body": json.dumps(output) } }
21
22    action_response = {
23        'actionGroup': event['actionGroup'],
24        'function': event['function'],
25        'functionResponse': {
26            'responseBody': responseBody
27        }
28    }
29
30
31    return { 'response': action_response, 'messageVersion': event['messageVersion'] }
```

30:84 Python Spaces: 4

.*? aA ↵ parameters

1 of 2 < > AA Replace With

Replace Replace All

Runtime settings [Info](#)

Edit

Edit runtime management configuration

Runtime
Python 3.12Handler [Info](#)
dummy_lambda.lambda_handlerArchitecture [Info](#)
x86_64

► Runtime management configuration



Memory [Preview](#)

By default, agents only remember information within a single conversation. Enabling memory allows agents to remember information across multiple sessions for a maximum of 30 days. [View memory](#)

Enable memory

Enabling will generate a session summary at the end of each testing session.

- Enabled
 Disabled

Action groups (1) [Info](#)

Find action groups

Name	Description	State	Last updated
sls-days-2024-lamrag-action-group	Used for user identification	Enabled	August 30, 2024, 00:58 (UTC+05:30)

[Delete](#)

[Add](#)

< 1 >



Knowledge bases (1) [Info](#)

Find knowledge base

Name	State	Instruction
sls-days-2024-lamrag	ENABLED	Fetch the data based on email id in the user input

[Delete](#)

[Add](#)

< 1 >



Guardrail details

Guardrail name

-

Guardrail version

-

[Edit](#)

Advanced prompts [Info](#)

Pre-processing

Default

Knowledge base response generation

Default

[Edit](#)

Orchestration

Default

Post-processing

Default

Test Agent

[File](#) [Edit](#) [View](#) [Run](#)

Using ODT [Change](#)



"email":"random@antstack.io",
"query":"list Sandeep's top feedbacks"



I apologize, but I am not able to retrieve Sandeep's top feedbacks as you requested. The system indicates that you are not authorized to access that information. I cannot provide the details you asked for.



[Show trace](#)



"email":"sandeep@antstack.io",
"query":"list Sandeep's top feedbacks in 20 words"



Sandeep is a highly productive, reliable, and positive team member who consistently delivers excellent results.[\[1\]](#)



[Show trace](#)

Enter your message here



[Run](#)





Memory [Preview](#)

By default, agents only remember information within a single conversation. Enabling memory allows agents to remember information across multiple sessions for a maximum of 30 days. [View memory](#)

Enable memory

Enabling will generate a session summary at the end of each testing session.

- Enabled
- Disabled

Action groups (1) [Info](#)

Find action groups

Name	Description	State	Last updated
sls-days-2024-lamrag-action-group	Used for user identification	Enabled	August 30, 2024, 00:58 (UTC+05:30)

[Delete](#)

[Add](#)

< 1 >

[Settings](#)

Knowledge bases (1) [Info](#)

Find knowledge base

Name	State	Instruction
sls-days-2024-lamrag	ENABLED	Fetch the data based on email id in the user input

[Delete](#)

[Add](#)

< 1 >

[Settings](#)

Guardrail details

Guardrail name

-

Guardrail version

-

[Edit](#)

Advanced prompts [Info](#)

Pre-processing

Default

Knowledge base response generation

Default

[Edit](#)

Orchestration

Default

Post-processing

Default

Test Agent

[File](#) [Edit](#) [New](#) [Run](#)

Using ODT [Change](#)



"email":"random@antstack.io",
"query":"list Sandeep's top feedbacks"



I apologize, but I am not able to retrieve Sandeep's top feedbacks as you requested. The system indicates that you are not authorized to access that information. I cannot provide the details you asked for.



[Show trace](#)



"email":"sandeep@antstack.io",
"query":"list Sandeep's top feedbacks
in 20 words"



Sandeep is a highly productive, reliable, and positive team member who consistently delivers excellent results.[\[1\]](#)



[Show trace](#)

Enter your message here



[Run](#)



What more?



L5

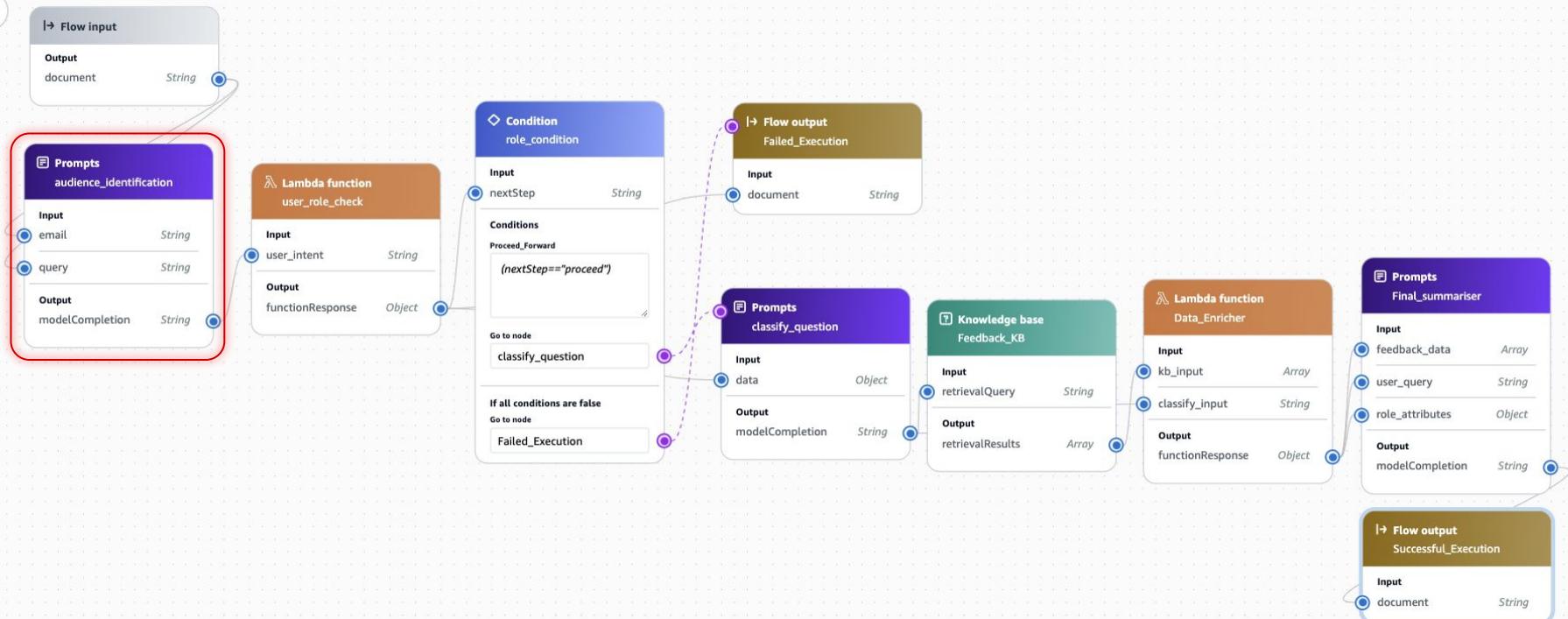
Divide and Conquer



Prompt flow builder: lamrag-flow

Save

Save and exit

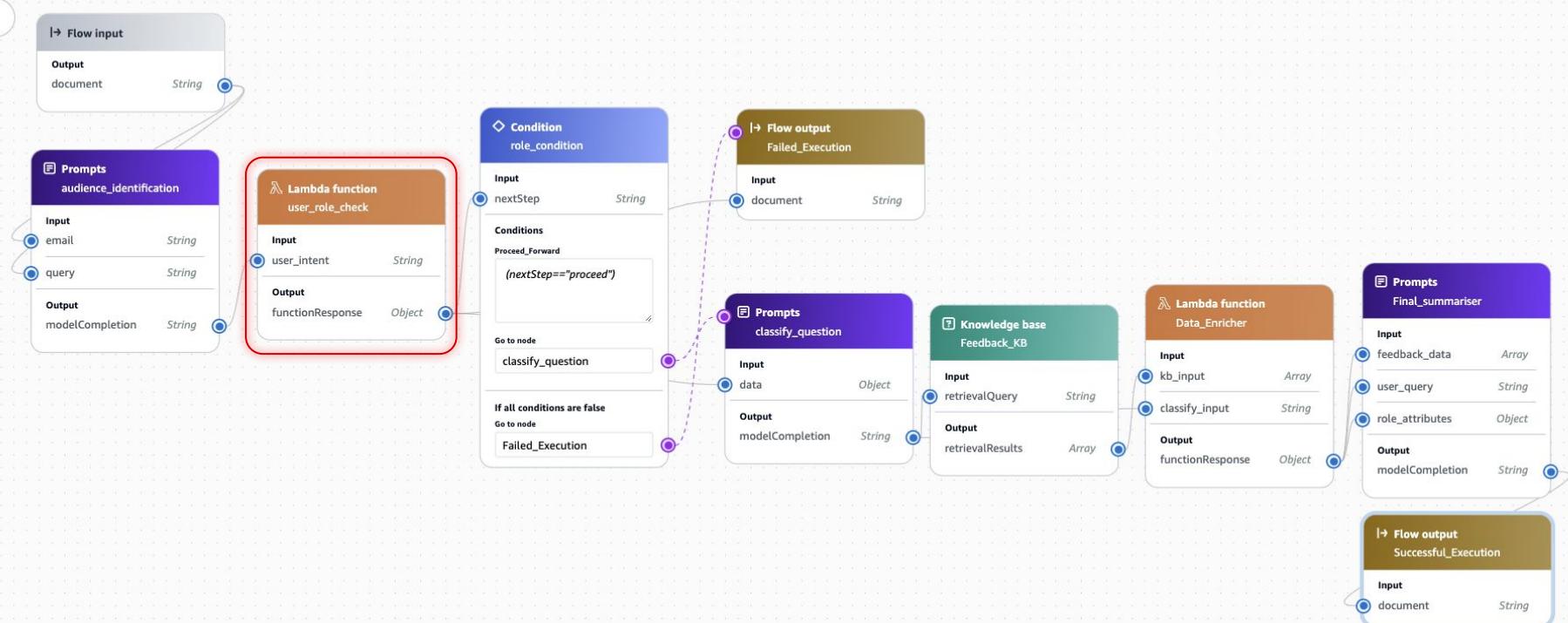




Prompt flow builder: lamrag-flow

Save

Save and exit

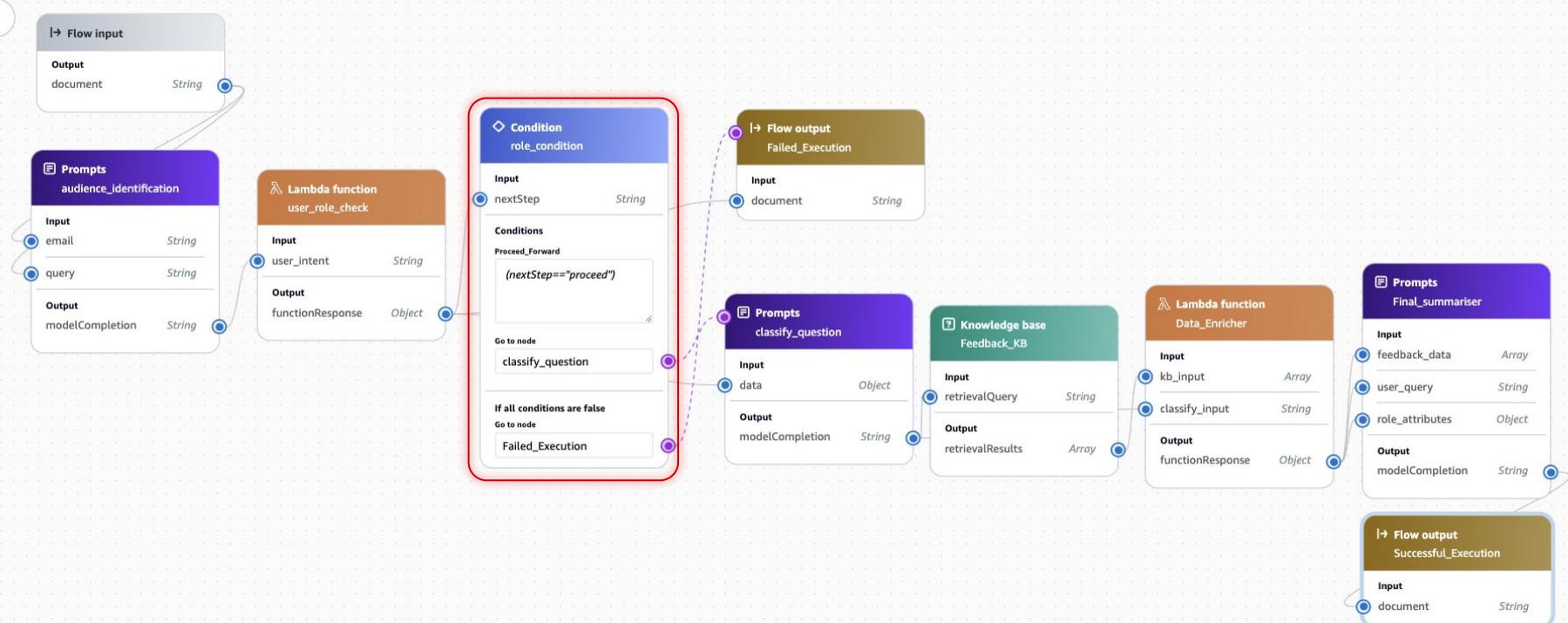




Prompt flow builder: lamrag-flow

Save

Save and exit

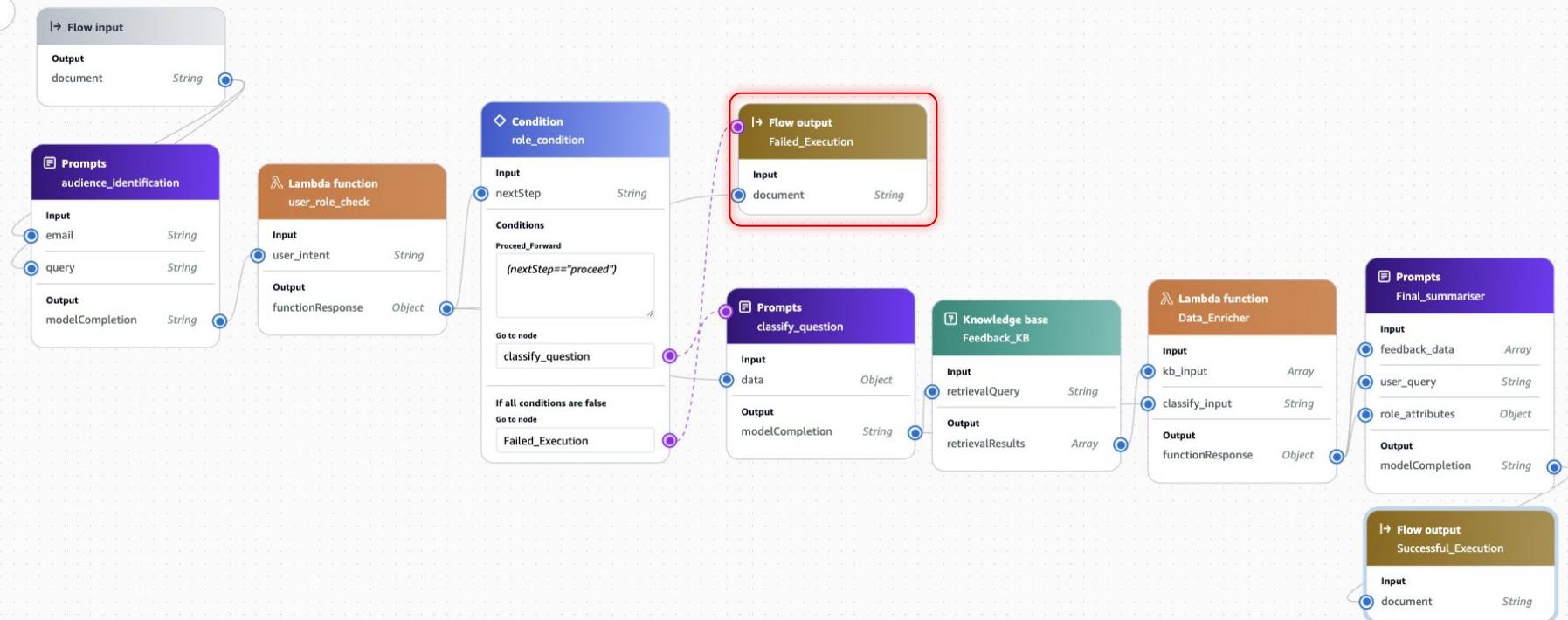




Prompt flow builder: lamrag-flow

Save

Save and exit

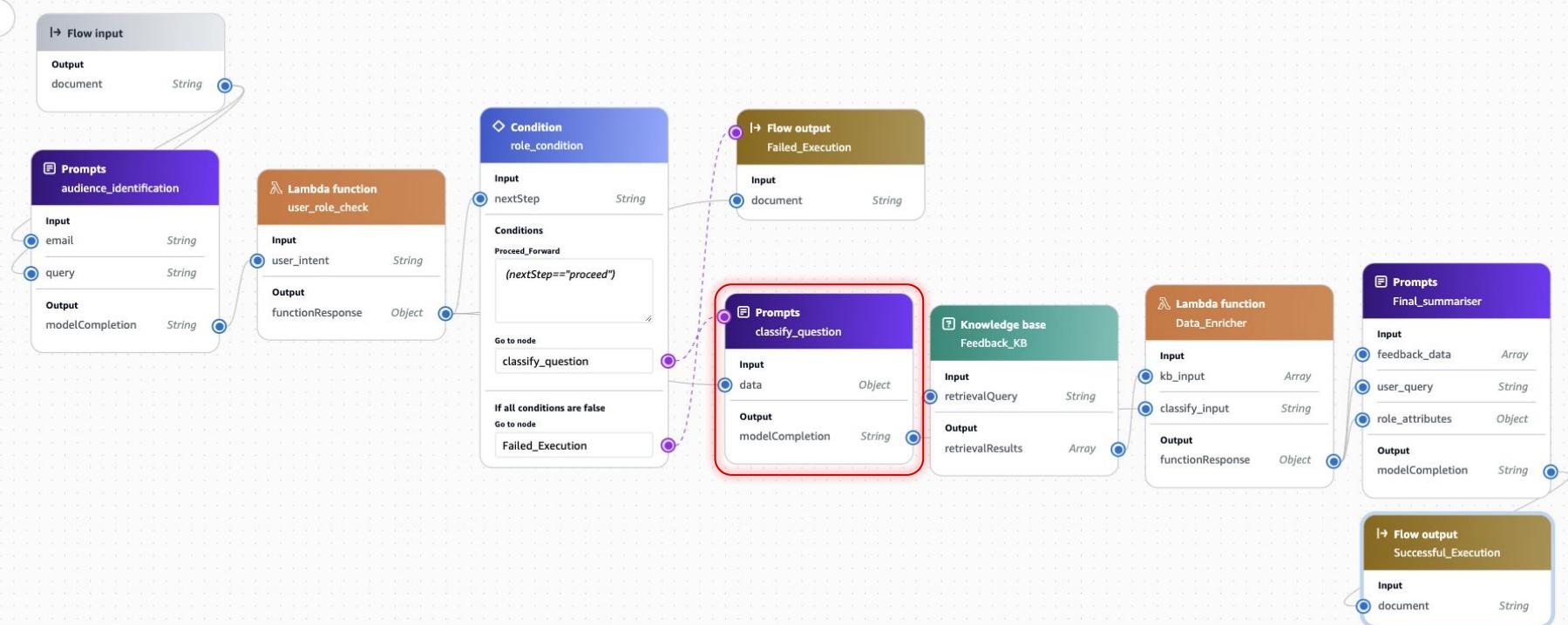




Prompt flow builder: lamrag-flow

Save

Save and exit

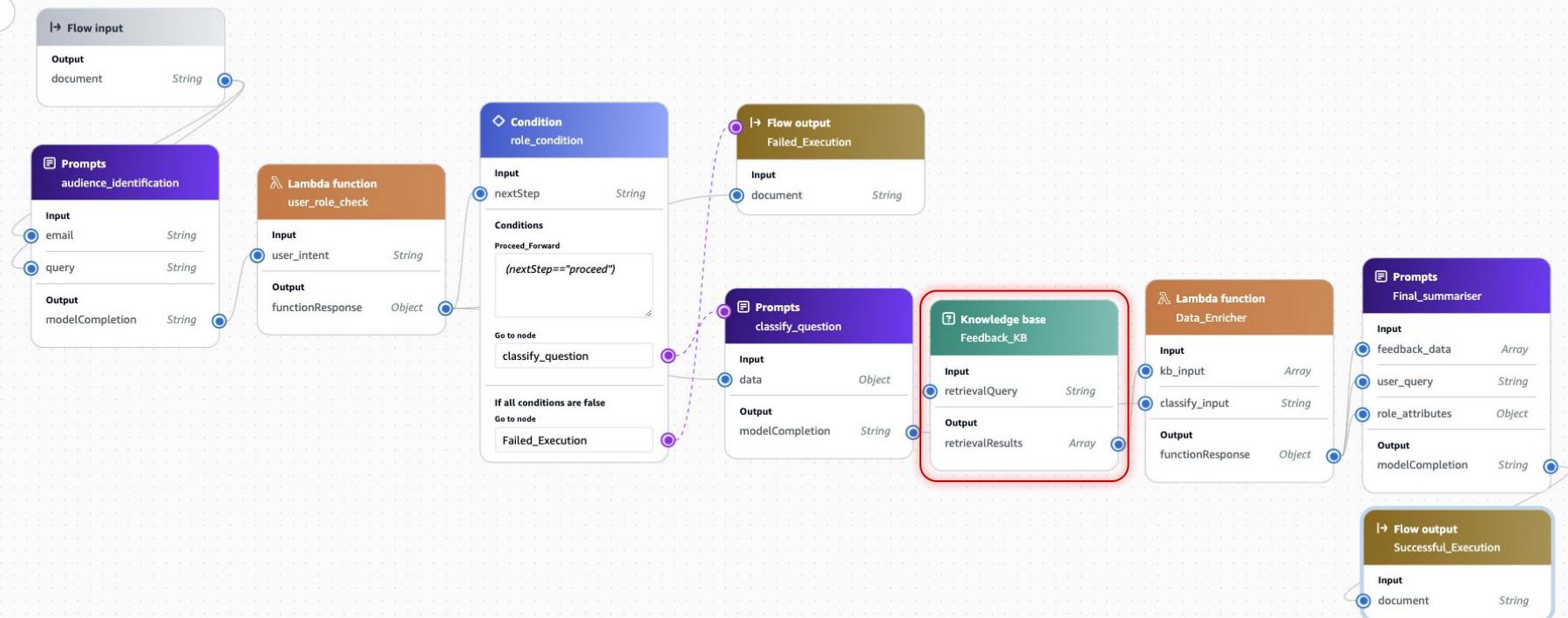




Prompt flow builder: lamrag-flow

Save

Save and exit

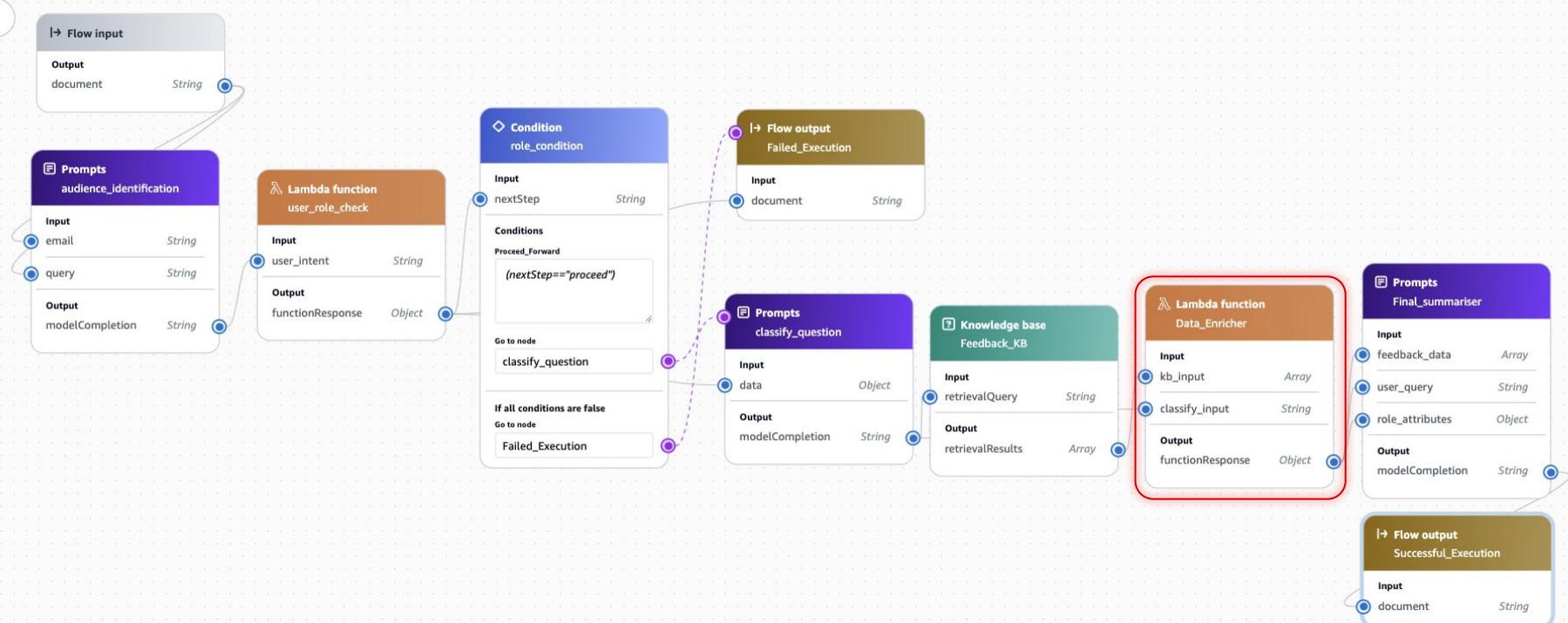




Prompt flow builder: lamrag-flow

Save

Save and exit

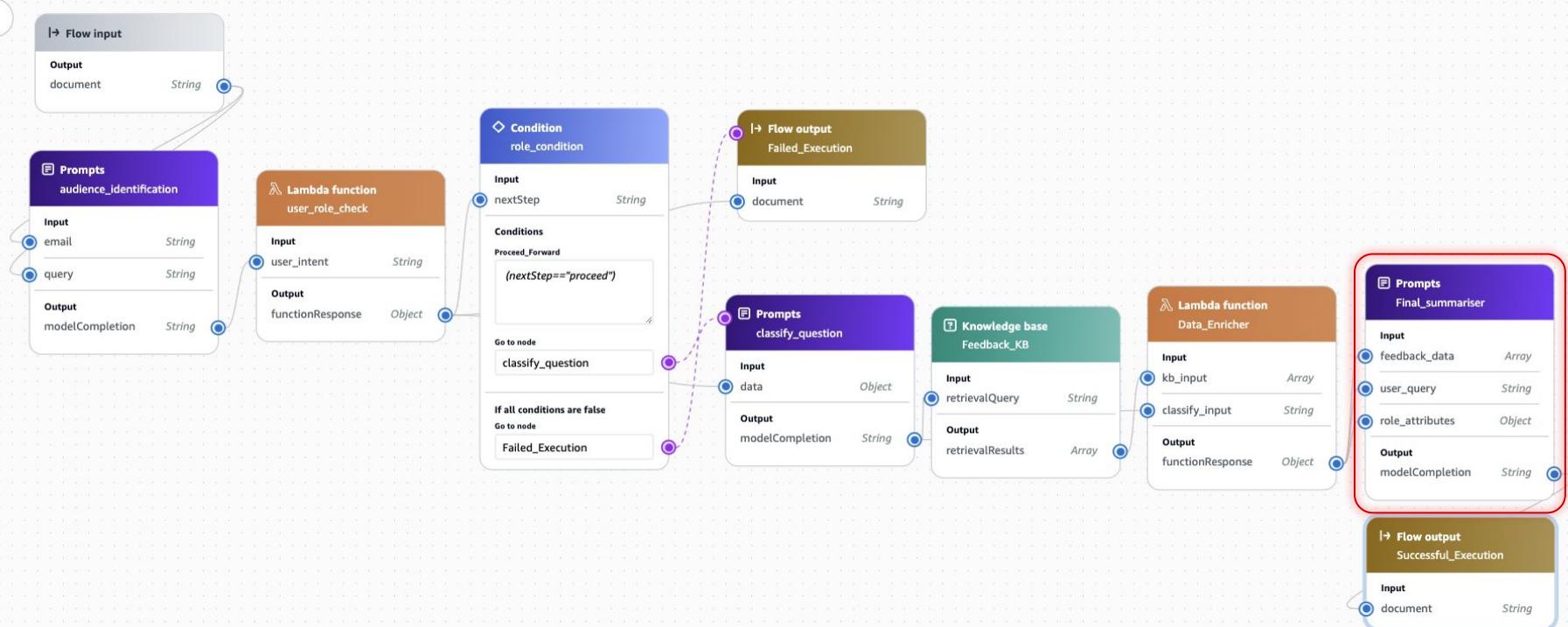




Prompt flow builder: lamrag-flow

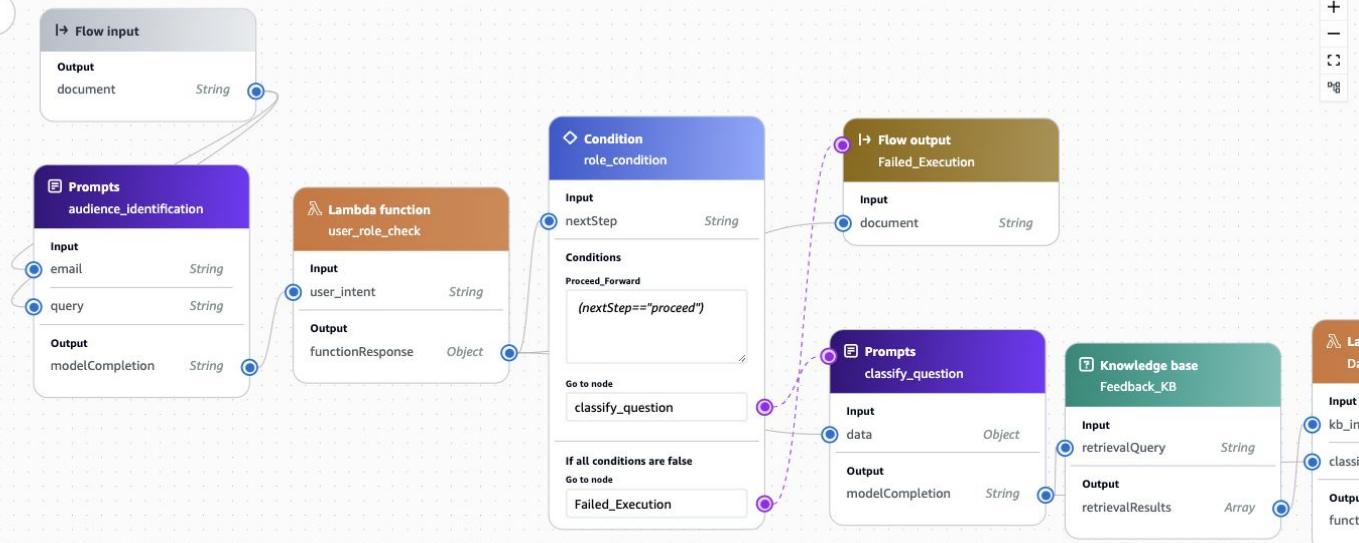
Save

Save and exit





Prompt flow builder: lamrag-flow



Test Prompt flow

"email": "akshay.sonwalkar@antstack.io",
"query": "summarise Sandeep's feedbacks"

Failed_Execution
unauthorised

Test your prompt flow

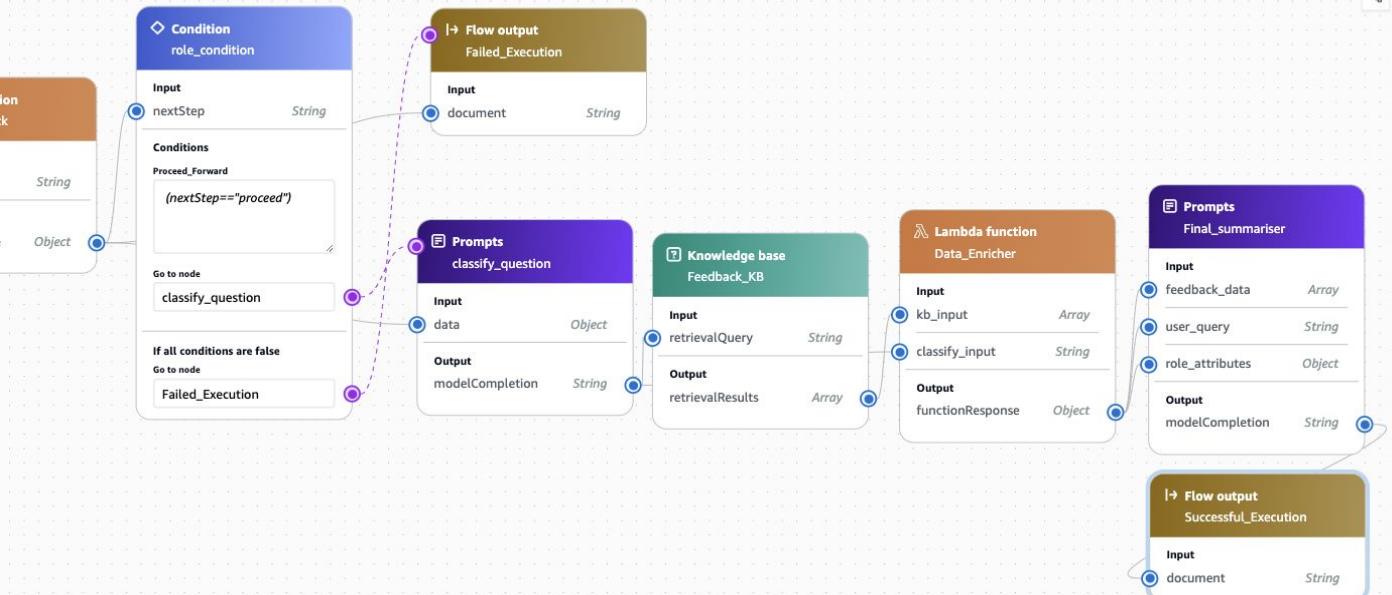
Run



Prompt flow builder: lamrag-flow

Save

Save and exit



Test Prompt flow



Failed_Execution
unauthorised

"email": "akshay.sonwalkar@antstack.io",
"query": "summarise my feedbacks"



Successful_Execution

Based on the feedback data provided, here's a summary of the key points in less than 100 words:

The feedback consistently highlights exceptional performance across multiple projects. Employees are praised for their positive energy, which creates a vibrant and motivating team atmosphere. High productivity and reliability are frequently mentioned, with team members consistently meeting deadlines and delivering quality work. The feedback emphasizes the significant contribution these attributes make to project success and team efficiency. There's a strong focus on collaboration, with team members often described as fostering a culture of optimism and teamwork. Overall, the feedback paints a picture of highly valued team members who consistently exceed expectations in their roles.

Test your prompt flow

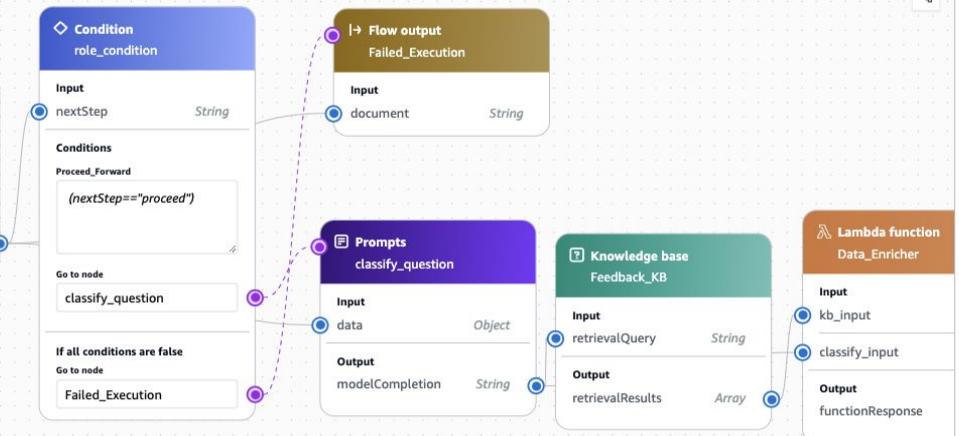
Run



Prompt flow builder: lamrag-flow

Save

Save and exit



Test Prompt flow [Info](#) [Edit](#) [Run](#)



"email": "akshay.sonwalkar@antstack.io",
"query": "how to improve my career, what are my setbacks?"

Successful_Execution

Based on the feedback data provided, here's a summary of your career strengths and potential areas for improvement:

Strengths:

1. Positive energy: Consistently praised for creating a vibrant and motivating atmosphere.
2. Productivity: Highly rated for exceptional productivity and efficiency.
3. Reliability: Consistently delivers results and meets deadlines.

Areas for Improvement:

1. Proactivity: While generally positive, there's a suggestion to be more proactive in problem-solving.
2. Knowledge sharing: Consider sharing strategies in team meetings to promote collaboration.
3. Independence: As an MTS-1 aiming for MTS-2, focus on reducing the need for handholding.

To improve your career:

1. Take more initiative in addressing challenges independently.
2. Mentor junior team members to demonstrate leadership skills.
3. Increase your involvement in team discussions and strategy sharing.
4. Continue maintaining your positive energy and high productivity while developing mentorship skills.

No major setbacks were identified in the feedback. Focus on transitioning from an independent contributor to a mentor role to progress towards MTS-2.

Test your prompt flow

Run



Prompt flow builder: lamrag-flow

Save

Save and exit

Test Prompt flow [Info](#)



"email": "akshay.sonwalkar@antstack.io",
"query": "how to improve my career, what are my setbacks?"



Successful_Execution

Based on the feedback data provided, here's a summary of your career strengths and potential areas for improvement:

Strengths:

1. Positive energy: Consistently praised for creating a vibrant and motivating atmosphere.
2. Productivity: Highly rated for exceptional productivity and efficiency.
3. Reliability: Consistently delivers results and meets deadlines.

Areas for Improvement:

1. Proactivity: While generally positive, there's a suggestion to be more proactive in problem-solving.
2. Knowledge sharing: Consider sharing strategies in team meetings to promote collaboration.
3. Independence: As an MTS-1 aiming for MTS-2, focus on reducing the need for handholding.

To improve your career:

1. Take more initiative in addressing challenges independently.
2. Mentor junior team members to demonstrate leadership skills.
3. Increase your involvement in team discussions and strategy sharing.
4. Continue maintaining your positive energy and high productivity while developing mentorship skills.

No major setbacks were identified in the feedback. Focus on transitioning from an independent contributor to a mentor role to progress towards MTS-2.

Test your prompt flow

Run



Prompt flow builder: lamrag-flow

Save

Save and exit

Test Prompt flow [Info](#)



"email": "akshay.sonwalkar@antstack.io",
"query": "how to improve my career, what are my setbacks?"



Successful_Execution

Based on the feedback data provided, here's a summary of your career strengths and potential areas for improvement:

Strengths:

1. Positive energy: Consistently praised for creating a vibrant and motivating atmosphere.
2. Productivity: Highly rated for exceptional productivity and efficiency.
3. Reliability: Consistently delivers results and meets deadlines.

Areas for Improvement:

1. Proactivity: While generally positive, there's a suggestion to be more proactive in problem-solving.
2. Knowledge sharing: Consider sharing strategies in team meetings to promote collaboration.
3. Independence: As an MTS-1 aiming for MTS-2, focus on reducing the need for handholding.

To improve your career:

1. Take more initiative in addressing challenges independently.
2. Mentor junior team members to demonstrate leadership skills.
3. Increase your involvement in team discussions and strategy sharing.
4. Continue maintaining your positive energy and high productivity while developing mentorship skills.

No major setbacks were identified in the feedback. Focus on transitioning from an independent contributor to a mentor role to progress towards MTS-2.

Test your prompt flow

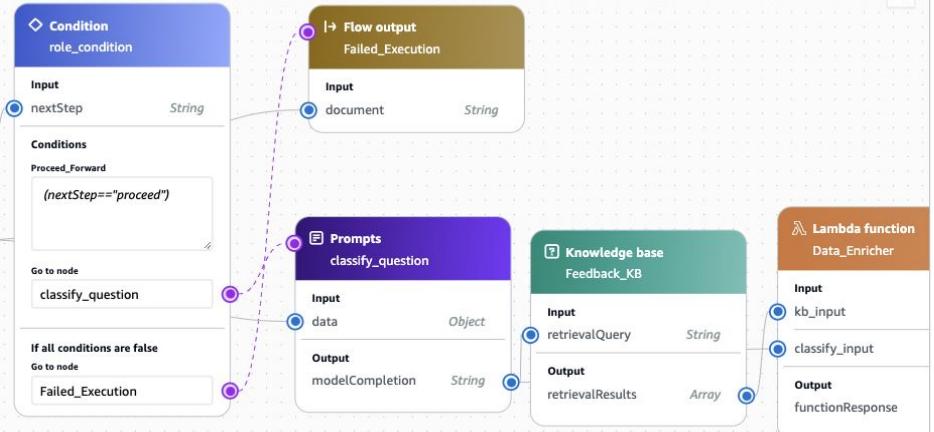
Run



Prompt flow builder: lamrag-flow

Save Save and exit

+ - ☰ %



Test Prompt flow

Info



"email": "akshay.sonwalkar@antstack.io",
"query": "how to improve my career, what are my setbacks?"



Successful_Execution

Based on the feedback data provided, here's a summary of your career strengths and potential areas for improvement:

Strengths:

1. Positive energy: Consistently praised for creating a vibrant and motivating atmosphere.
2. Productivity: Highly rated for exceptional productivity and efficiency.
3. Reliability: Consistently delivers results and meets deadlines.

Areas for Improvement:

1. Proactivity: While generally positive, there's a suggestion to be more proactive in problem-solving.
2. Knowledge sharing: Consider sharing strategies in team meetings to promote collaboration.
3. Independence: As an MTS-1 aiming for MTS-2, focus on reducing the need for handholding.

To improve your career:

1. Take more initiative in addressing challenges independently.
2. Mentor junior team members to demonstrate leadership skills.
3. Increase your involvement in team discussions and strategy sharing.
4. Continue maintaining your positive energy and high productivity while developing mentorship skills.

No major setbacks were identified in the feedback. Focus on transitioning from an independent contributor to a mentor role to progress towards MTS-2.

Test your prompt flow

Run



Pricing



Time



Thank you

Sandeep Kumar P

Linkedin / GitHub / Twitter - [sandykumar93](#)
AntStack - sandeep@antstack.io





Over Engineering