

```

{
  "name": "RAG Evaluation",
  "nodes": [
    {
      "parameters": {
        "model": "openai/gpt-5",
        "options": {}
      },
      "type": "@n8n/n8n-nodes-langchain.lmChatOpenRouter",
      "typeVersion": 1,
      "position": [
        -1008,
        880
      ],
      "id": "6a30e148-554e-4491-a58a-d0a438411ad0",
      "name": "OpenRouter Chat Model1",
      "credentials": {
        "openRouterApi": {
          "id": "SSbXtVGQUtbHb5xe",
          "name": "OpenRouter account"
        }
      }
    },
    {
      "parameters": {
        "promptType": "define",
        "text": "=Subject: {{ $json.subject }}\n\nBody: {{ $json.text }}",
        "hasOutputParser": true,
        "options": {}
      },
      "type": "@n8n/n8n-nodes-langchain.agent",
      "typeVersion": 2.1,
      "position": [
        -864,
        656
      ],
      "id": "f45bd553-8108-4791-94be-423923957096",
      "name": "Support Agent"
    },
    {
      "parameters": {
        "mode": "retrieve-as-tool",
        "toolDescription": "Use this tool to get policy and FAQ information.",
        "tableName": {

```

```

    "__rl": true,
    "value": "documents",
    "mode": "list",
    "cachedResultName": "documents"
  },
  "options": {}
},
"type": "@n8n/n8n-nodes-langchain.vectorStoreSupabase",
"typeVersion": 1.3,
"position": [
  -880,
  880
],
"id": "1f52e411-d578-4194-b79b-9a4edde1c386",
"name": "Supabase Vector Store",
"credentials": {
  "supabaseApi": {
    "id": "rV3aISrTqkaqXbeb",
    "name": "Supabase account"
  }
}
},
{
  "parameters": {
    "options": {}
  },
  "type": "@n8n/n8n-nodes-langchain.embeddingsOpenAi",
"typeVersion": 1.2,
"position": [
  -800,
  1088
],
"id": "0867fdf9-226f-4cd7-b15b-a4afa387b32d",
"name": "Embeddings OpenAI",
"credentials": {
  "openAiApi": {
    "id": "z7H19jHBDY9xp9e5",
    "name": "OpenAi account"
  }
}
},
{
  "parameters": {
    "schemaType": "manual",

```

```

      "inputSchema": "{\n  \"type\": \"object\", \n  \"properties\": {\n    \"subject\": {\n      \"type\": \"string\", \n      \"description\": \"the email subject\" \n    }, \n    \"emailBody\": {\n      \"type\": \"string\", \n      \"description\": \"the email body\" \n    } \n  }, \n  \"required\": [\"subject\", \"emailBody\"] \n}"
    },
    "type": "@n8n/n8n-nodes-langchain.outputParserStructured",
    "typeVersion": 1.2,
    "position": [
      -592,
      880
    ],
    "id": "0716c3e7-447c-4553-bb04-80a637b2be2a",
    "name": "Subject & Email"
  },
  {
    "parameters": {
      "documentId": {
        "__rl": true,
        "value": "1SOOeOT3eF_jwitUYn6irkjTpogXwSWFdoPHLeUgV7Kc",
        "mode": "list",
        "cachedResultName": "RAG Evaluation",
        "cachedResultUrl":
"https://docs.google.com/spreadsheets/d/1SOOeOT3eF\_jwitUYn6irkjTpogXwSWFdoPHLeUgV7Kc/edit?usp=drivesdk"
      },
      "sheetName": {
        "__rl": true,
        "value": "gid=0",
        "mode": "list",
        "cachedResultName": "Sheet1",
        "cachedResultUrl":
"https://docs.google.com/spreadsheets/d/1SOOeOT3eF\_jwitUYn6irkjTpogXwSWFdoPHLeUgV7Kc/edit#gid=0"
      },
      "outputs": {
        "values": [
          {
            "outputName": "Actual Answer",
            "outputValue": "={{ $json.output.emailBody }}"
          }
        ]
      }
    },
    "type": "n8n-nodes-base.evaluation",

```

```

"typeVersion": 4.7,
"position": [
  -160,
  560
],
"id": "909128a3-1fd8-416f-ac02-700284b0912d",
"name": "Set Output",
"credentials": {
  "googleSheetsOAuth2Api": {
    "id": "RYpwEEqze5ZySnXF",
    "name": "Google Sheets account"
  }
}
},
{
  "parameters": {
    "sendTo": "={{ $('Gmail Trigger').item.json.to.value[0].address }}",
    "subject": "={{ $json.output.subject }}",
    "emailType": "text",
    "message": "={{ $json.output.emailBody }}",
    "options": {}
  },
  "type": "n8n-nodes-base.gmail",
  "typeVersion": 2.1,
  "position": [
    -160,
    752
  ],
  "id": "5c38c305-eeb4-4645-9c9b-258b100c925d",
  "name": "Send a message",
  "webhookId": "0816041e-e10f-48f4-84a7-fc535592cee7",
  "credentials": {
    "gmailOAuth2": {
      "id": "thsdSH3Esq755UBe",
      "name": "Gmail account"
    }
  }
},
{
  "parameters": {
    "pollTimes": {
      "item": [
        {
          "mode": "everyMinute"
        }
      ]
    }
  }
}

```

```

    }
  ]
},
"simple": false,
"filters": {},
"options": {}
},
"type": "n8n-nodes-base.gmailTrigger",
"typeVersion": 1.2,
"position": [
  -1232,
  752
],
"id": "a848d048-756d-46d7-8216-4acc794d20bb",
"name": "Gmail Trigger",
"credentials": {
  "gmailOAuth2": {
    "id": "thsdSH3Esq755UBe",
    "name": "Gmail account"
  }
}
},
{
  "parameters": {
    "assignments": {
      "assignments": [
        {
          "id": "2bff1605-23ba-42c4-bb58-6762420fe256",
          "name": "subject",
          "value": "={{ $json.Subject }}",
          "type": "string"
        },
        {
          "id": "96fc28f4-50b5-406a-b0c3-c4e3ce2ebae9",
          "name": "text",
          "value": "={{ $json.Body }}",
          "type": "string"
        }
      ]
    }
  },
  "options": {}
},
"type": "n8n-nodes-base.set",
"typeVersion": 3.4,

```

```

    "position": [
      -1232,
      560
    ],
    "id": "69aa5460-da38-4557-9ecb-f83228cd7c21",
    "name": "Edit Fields"
  },
  {
    "parameters": {
      "operation": "checkIfEvaluating"
    },
    "type": "n8n-nodes-base.evaluation",
    "typeVersion": 4.7,
    "position": [
      -384,
      656
    ],
    "id": "58389403-e867-4906-9bb9-fdbe6ef29d97",
    "name": "Evaluation"
  },
  {
    "parameters": {
      "documentId": {
        "__rl": true,
        "value": "1SOOeOT3eF_jwitUYn6irkjTpogXwSWFdoPHLeUgV7Kc",
        "mode": "list",
        "cachedResultName": "RAG Evaluation",
        "cachedResultUrl":
"https://docs.google.com/spreadsheets/d/1SOOeOT3eF\_jwitUYn6irkjTpogXwSWFdoPHLeUgV7Kc/edit?usp=drivesdk"
      },
      "sheetName": {
        "__rl": true,
        "value": "gid=0",
        "mode": "list",
        "cachedResultName": "Sheet1",
        "cachedResultUrl":
"https://docs.google.com/spreadsheets/d/1SOOeOT3eF\_jwitUYn6irkjTpogXwSWFdoPHLeUgV7Kc/edit#gid=0"
      }
    },
    "type": "n8n-nodes-base.evaluationTrigger",
    "typeVersion": 4.6,
    "position": [

```

```

-1456,
560
],
"id": "3dfdc9c-9c73-4819-86a5-abcb9668a9c6",
"name": "When fetching a dataset row",
"credentials": {
  "googleSheetsOAuth2Api": {
    "id": "RYpwEEqze5ZySnXF",
    "name": "Google Sheets account"
  }
},
},
{
  "parameters": {
    "model": "openai/gpt-4.1",
    "options": {}
  },
  "type": "@n8n/n8n-nodes-langchain.lmChatOpenRouter",
  "typeVersion": 1,
  "position": [
    32,
    800
  ],
  "id": "bfe7d570-0446-46bf-98e0-673cc996f4d9",
  "name": "OpenRouter Chat Model",
  "credentials": {
    "openRouterApi": {
      "id": "SSbXtVGQUtbHb5xe",
      "name": "OpenRouter account"
    }
  },
},
{
  "parameters": {
    "promptType": "define",
    "text": "=Expected Answer: {{ $('When fetching a dataset row').item.json['Expected Answer'] }}\n\nActual Answer: {{ $('Support Agent').item.json.output.emailBody }}",
    "hasOutputParser": true,
    "options": {
      "systemMessage": "You are an expert factual evaluator assessing the accuracy of answers compared to established ground truths.\n\nEvaluate the factual correctness of a given output compared to the provided ground truth on a scale from 1 to 5. Use detailed reasoning to thoroughly analyze all claims before determining the final score.\n\n# Scoring Criteria\n\n- 5: Highly similar - The output and ground truth are nearly identical, with only minor, insignificant

```

differences.

4: Somewhat similar - The output is largely similar to the ground truth but has few noticeable differences.

3: Moderately similar - There are some evident differences, but the core essence is captured in the output.

2: Slightly similar - The output only captures a few elements of the ground truth and contains several differences.

1: Not similar - The output is significantly different from the ground truth, with few or no matching elements.

Evaluation Steps

1. Identify and list the key elements present in both the output and the ground truth.
2. Compare these key elements to evaluate their similarities and differences, considering both content and structure.
3. Analyze the semantic meaning conveyed by both the output and the ground truth, noting any significant deviations.
4. Consider factual accuracy of specific details, including names, dates, numbers, and relationships.
5. Assess whether the output maintains the factual integrity of the ground truth, even if phrased differently.
6. Determine the overall level of similarity and accuracy according to the defined criteria.

Output Format

Provide:

- A detailed analysis of the comparison (extended reasoning)
- A one-sentence summary highlighting key differences (not similarities)
- The final similarity score as an integer (1, 2, 3, 4, or 5)

Always follow the JSON format below and return nothing else:

```
{
  "extended_reasoning": "<detailed step-by-step analysis of factual accuracy and similarity>",
  "reasoning_summary": "<one sentence summary focusing on key differences>",
  "score": <number: integer from 1 to 5>
}
```

Examples

Example 1:

Input: Output: "The cat sat on the mat." Ground Truth: "The feline is sitting on the rug."

Expected Output:

```
{
  "extended_reasoning": "I need to compare 'The cat sat on the mat' with 'The feline is sitting on the rug.' First, let me identify the key elements: both describe an animal ('cat' vs 'feline') in a position ('sat' vs 'sitting') on a surface ('mat' vs 'rug'). The subject is semantically identical - 'cat' and 'feline' refer to the same animal. The action is also semantically equivalent - 'sat' and 'sitting' both describe the same position, though one is past tense and one is present continuous. The location differs in specific wording ('mat' vs 'rug') but both refer to floor coverings that serve the same function. The basic structure and meaning of both sentences are preserved, though they use different vocabulary and slightly different tense. The core information being conveyed is the same, but there are noticeable wording differences.",
  "reasoning_summary": "The sentences differ in vocabulary choice ('cat' vs 'feline', 'mat' vs 'rug') and verb tense ('sat' vs 'is sitting').",
  "score": 3
}
```

Example 2:

Input: Output: "The quick brown fox jumps over the lazy dog." Ground Truth: "A fast brown animal leaps over a sleeping canine."

Expected Output:

```
{
  "extended_reasoning": "I need to compare 'The quick brown fox jumps over the lazy dog' with 'A fast brown animal leaps over a sleeping canine.' Starting with the subjects: 'quick brown fox' vs 'fast brown animal'. Both describe the same entity (a fox is a type of animal) with the same attributes (quick/fast and brown). The action is described as 'jumps' vs 'leaps', which are synonymous verbs describing the same motion. The object in both sentences is a dog, described as 'lazy' in one and 'sleeping' in the other, which are related concepts (a sleeping dog could be perceived as lazy). The structure follows the same pattern: subject + action + over + object. The sentences convey the same scene with slightly different word choices that maintain the core meaning. The level of specificity differs slightly ('fox' vs 'animal', 'dog' vs 'canine'), but the underlying information and imagery remain very similar.",
  "reasoning_summary": "The sentences use different but synonymous terminology ('quick' vs 'fast', 'jumps' vs 'leaps', 'lazy' vs 'sleeping') and varying levels of specificity ('fox' vs 'animal', 'dog' vs 'canine').",
  "score": 

```



```
        "type": "number",
        "id": "658e5318-5583-492e-a338-4446c1f91d2b"
      }
    ]
  },
  "type": "n8n-nodes-base.evaluation",
  "typeVersion": 4.7,
  "position": [
    416,
    560
  ],
  "id": "47e76675-ae49-40e5-a740-dfcad8a9a3bf",
  "name": "Score"
}
],
"pinData": {},
"connections": {
  "OpenRouter Chat Model1": {
    "ai_languageModel": [
      [
        {
          "node": "Support Agent",
          "type": "ai_languageModel",
          "index": 0
        }
      ]
    ]
  }
},
"Supabase Vector Store": {
  "ai_tool": [
    [
      {
        "node": "Support Agent",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
},
"Embeddings OpenAI": {
  "ai_embedding": [
    [
      {
```

```
    "node": "Supabase Vector Store",
    "type": "ai_embedding",
    "index": 0
  }
]
],
},
"Subject & Email": {
  "ai_outputParser": [
    [
      {
        "node": "Support Agent",
        "type": "ai_outputParser",
        "index": 0
      }
    ]
  ]
},
"Support Agent": {
  "main": [
    [
      {
        "node": "Evaluation",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Gmail Trigger": {
  "main": [
    [
      {
        "node": "Support Agent",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
>Edit Fields": {
  "main": [
    [
      {
```

```
    "node": "Support Agent",
    "type": "main",
    "index": 0
  }
]
},
"Evaluation": {
  "main": [
    [
      {
        "node": "Set Output",
        "type": "main",
        "index": 0
      }
    ],
    [
      {
        "node": "Send a message",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"When fetching a dataset row": {
  "main": [
    [
      {
        "node": "Edit Fields",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Set Output": {
  "main": [
    [
      {
        "node": "AI Agent",
        "type": "main",
        "index": 0
      }
    ]
  ]
}
```

```
]
]
},
"OpenRouter Chat Model": {
  "ai_languageModel": [
    [
      {
        "node": "AI Agent",
        "type": "ai_languageModel",
        "index": 0
      }
    ]
  ]
},
"Structured Output Parser": {
  "ai_outputParser": [
    [
      {
        "node": "AI Agent",
        "type": "ai_outputParser",
        "index": 0
      }
    ]
  ]
},
"AI Agent": {
  "main": [
    [
      {
        "node": "Score",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"active": false,
"settings": {
  "executionOrder": "v1"
},
"versionId": "81d5ce62-d22c-4a11-ba11-218e8bda86b8",
"meta": {
  "templateCredsSetupCompleted": true,
```

```
    "instanceId": "435651f0100ae643a67ce22b5f9d0a11469aabd62907e30707768403632f9485"  
  },  
  "id": "yT2W6oWp7op0yhI0",  
  "tags": []  
}
```