RAG:

Step 3. Build RAG Chain

修改Template，添加

```
"The answer should be in JSON format.\n"
"I am  a Chinese ,so your answer should be in chinese"
```

```
template = (
    "You are an AMD ROCm product expert. Please answer and provide guidance based
on the user's question according to the product manual.\n"
    "----------------------\n"
    "{context_str}\n"
    "----------------------\n"
    "Please answer based on the content of the product manual.\n"
    "When answering, provide the page number(s) in the product manual where the
relevant information can be found.\n"
    "If the question goes beyond the scope of the manual, clearly inform the user
that the question is out of the manual's scope.\n"
    "The answer should be accurate and concise.\n"
    "The answer should be in JSON format.\n"
    "I am  a Chinese ,so your answer should be in chinese"
    "----------------------\n"
    "User's question: {query_str}\n"
    "----------------------\n"
    "Answer: "
)
```

测试结果如下：

```
user_question = "How to install ROCm?"
output = rag_chain.invoke(user_question)
print(output)
```

```
{
"answer": "根据产品手册，安装ROCm可以通过以下两种方法：

Option A：使用 PIP 安装
请参考《Use ROCm on Radeon GPUs Documentation》中的第2.1.2节。

Option B：使用 Docker 安装
请参考《Use ROCm on Radeon GPUs Documentation》中的 ROCm 安装选项。

请注意，安装前请确保已安装 Radeon 软件和 MIGraphX。",
"ref": "《Use ROCm on Radeon GPUs Documentation》, 2.1"
}
```

```
user_question = "What is ROCm?"
output = rag_chain.invoke(user_question)
print(output)
```

```
{
"answer": "ROCm 是 AMD 的一个深度学习计算平台，用于在 Radeon GPU 上运行机器学习应用程序。",
"reference": "页 1"
}
```

see the *lab2_rag_langchain.ipynb*

Step 2. Indexing your file

Replace the the input documentation

see the *lab2_amba_chi.ipynb*