

# Relatório de Crise: A “Grande Fragmentação” do LangChain v0.3

O erro de `ModuleNotFoundError` que enfrentei **7 vezes** é o sintoma clássico da mudança radical de arquitetura na versão 0.3 do LangChain. O problema não estava apenas no código, mas na estrutura física dos pacotes dentro do seu container Docker.

## ✖ Por que as soluções anteriores falharam?

Abaixo, os motivos técnicos de cada tentativa frustrada:

### 1. O Problema do “Cápsula Vazia” (Namespace Core)

- **Tentativa:** `from langchain.retrievers import EnsembleRetriever`
- **Causa da Falha:** Na v0.3, o pacote `langchain` tornou-se um “wrapper” vazio. Ele não contém mais os códigos dos retrievers; ele apenas aponta para outros pacotes. Se a instalação no Docker não for sincronizada entre `langchain` e `langchain-community`, esse “ponteiro” quebra e o Python não encontra o caminho.

### 2. O Erro de Índice (Namespace Community)

- **Tentativa:** `from langchain_community.retrievers import EnsembleRetriever`
- **Causa da Falha:** O arquivo `__init__.py` do pacote `community` muitas vezes omite certos módulos para otimizar o peso da biblioteca. Quando tentei o “atalho”, o Python busca o nome no índice e gera um `ImportError` porque o nome não está registrado ali.

### 3. A Pasta que Deixou de Existir (Deep Import)

- **Tentativa:** `from langchain.retrievers.ensemble import EnsembleRetriever`
- **Causa da Falha:** O Python busca uma pasta física chamada `retrievers` dentro do diretório `langchain`. Como a v0.3 moveu esses arquivos para pacotes isolados, essa pasta simplesmente não existe no local esperado dentro do seu container.

### 4. O Conflito de “Módulos Fantasma” no Docker

- **O Cenário:** O Docker utiliza camadas de cache. Se você instalou uma versão anterior e tentou atualizar para a 0.3, o container mantém “módulos fantasma” — ele vê a pasta da biblioteca, mas os arquivos internos mudaram de lugar, causando uma confusão fatal nos caminhos de busca do Python.



# O Diagnóstico e a Solução Final

**O veredito:** O Python não está encontrando o módulo porque ele fisicamente não está onde deveria. No LangChain 0.3, a importação direta de `langchain.retrievers` é instável ou inexistente para certas classes.

## A Estratégia de Recuperação:

Para “matar” esse erro de vez, foi adotada a **estabilidade sobre a complexidade**:

1. **Limpeza de Fundação:** Simplificar o `Dockerfile` para garantir que as bibliotecas sejam instaladas de forma atômica (tudo na mesma camada).
2. **Remoção de Abstrações Frágeis:** Abandonar o `EnsembleRetriever` momentaneamente. Em vez disso, usar uma lógica manual de fusão de resultados que não depende dessas importações problemáticas.
3. **Código Blindado:** Usar um código que utiliza apenas os pacotes básicos ( `langchain-openai` e `langchain-milvus` ), garantindo que o sistema rode hoje.