现在到哪步了？

我现在可以通过字符串在向量数据库中使用metadata进行筛选了。

也可以让大模型生成指定格式的回复了。

现在存在的问题是，

1. 我怎么根据输入的query生成对应的metadata传给向量数据库？

可能的方法1.在大模型生成结果的时候指定，即结果中要返回对应的metadata以及对应的操作符。2.再使用一个大模型专门处理query，根据这个query重新整理成metadata的格式，这个是Automeretreiver正在做的 ，但是问题是，他处理后的结果并不能匹配我的u需求，他并不能返回给我一个类似.like(\*str\*)的结果，因为llamaIndex不支持like操作符。

所以要么使用方法1自己自定义一个llm，要么使用方法2，修改包的源代码，这个方法我试了很久，但效果不好，如果硬要改的话需要理解包中所有的调用逻辑方法，不太现实，不如我模仿他的思路自己写一个，即方法1.

或者我不对query处理呢？现实吗？不行，那样metadata就没用了。就白创建了。所以还是需要处理成能使用metadata filter的格式。

所以下一步就还是需要我自己重新构建一个大模型，通过引用文章中的few-shot方法，或者什么，来重构query，便于应用到filter。

1. 长代码怎么处理？

阅读了几篇文章，他们并不会遇到长代码的问题，他们都是专注于改进生成的summary的质量，没有提到代码过长的问题。而且他们都是在生成commit，而不是对整个代码文件的摘要。

这几个文章可以帮我改进生成摘要的质量等作用。

但不解决目前的本质问题。目前的问题仍然是我的代码太长了，一个文件1w多token，扔进模型显著降低结果质量。

之前的按照代码功能进行重新分块的方法，首先我们需要去理解代码起了什么作用，这步能用人工吗？不显示。所以还是只能用LLM，如果用LLM的话，那给他输入什么代码呢？多大的代码块？如果我们在这步进行了分割，那为什么不直接把分割的结果作为结果？

而且代码混淆的问题，我们没法解决。

晚上我说什么？

取得的进度，一个是受到老师的启发，往字符串方向思考，现在数据库换成了weaviate，能成功解决字符串匹配的问题。剩余需要做的是用大模型处理query来匹配元数据的筛选格式。如果这个问题解决了，那我们的框架应当初步搭建完成了。

另外关于长代码的问题，上没有解决方法。

读了两篇关于使用LLM进行malware code detection的文章，方法都是通过大模型生成代码信息，和已知的malware进行比较，最终得出分类结果。

而且两篇文章都遇到了输入过长的问题，一个没有解决，另一个没说怎么解决只说了segmented。