67K8sGPT + LocalAI：免费解锁 Kubernetes 超能力！

众所周知，LLMs 正在疯狂流行，炒作并非没有道理。大量利用基于 LLM 的文本生成的很酷的项目正在涌现——事实上，如果在我写这篇博客的时候发布了另一个很棒的新工具，我不会感到惊讶 :)

对于非信徒，我说炒作是有道理的，因为这些项目不仅仅是噱头。他们正在释放真正的价值，远不止使用 ChatGPT 来发布博客文章 。例如，开发人员通过 Warp AI 在他们的终端中直接提高他们的生产力，在他们的 IDE 中使用 IntelliCode、GitHub Copilot、CodeGPT（开源！），以及可能还有 300 种我还没有遇到过的其他工具。此外，该技术的用例远远超出了代码生成。基于 LLM 的聊天和 Slack 机器人正在出现，它们可以在组织的内部文档语料库上进行训练。特别是来自 Nomic AI 的 GPT4All 是一个很棒的项目，可以在开源聊天空间中查看。

然而，本博客的重点是另一个用例：在您的 Kubernetes 集群中运行的基于 AI 的站点可靠性工程师 (SRE) 听起来如何？浏览 K8sGPT（https://github.com/k8sgpt-ai/k8sgpt） 和 k8sgpt-operator（https://github.com/k8sgpt-ai/k8sgpt-operator）。

以下是他们的自述文件：

> k8sgpt 是一个用简单的英语扫描 Kubernetes 集群、诊断和分类问题的工具。它将 SRE 经验编纂到其分析器中，并帮助提取最相关的信息以通过 AI 丰富它。

听起来不错，对吧？我当然这么认为！如果你想尽快启动并运行，或者如果你想访问最强大的商业化模型，你可以使用 Helm 安装 K8sGPT 服务（不使用 K8sGPT Operator）并利用 K8sGPT 的默认人工智能后端：OpenAI。

但是，如果我告诉您免费的本地（集群内）分析也是一个直截了当的提议呢？

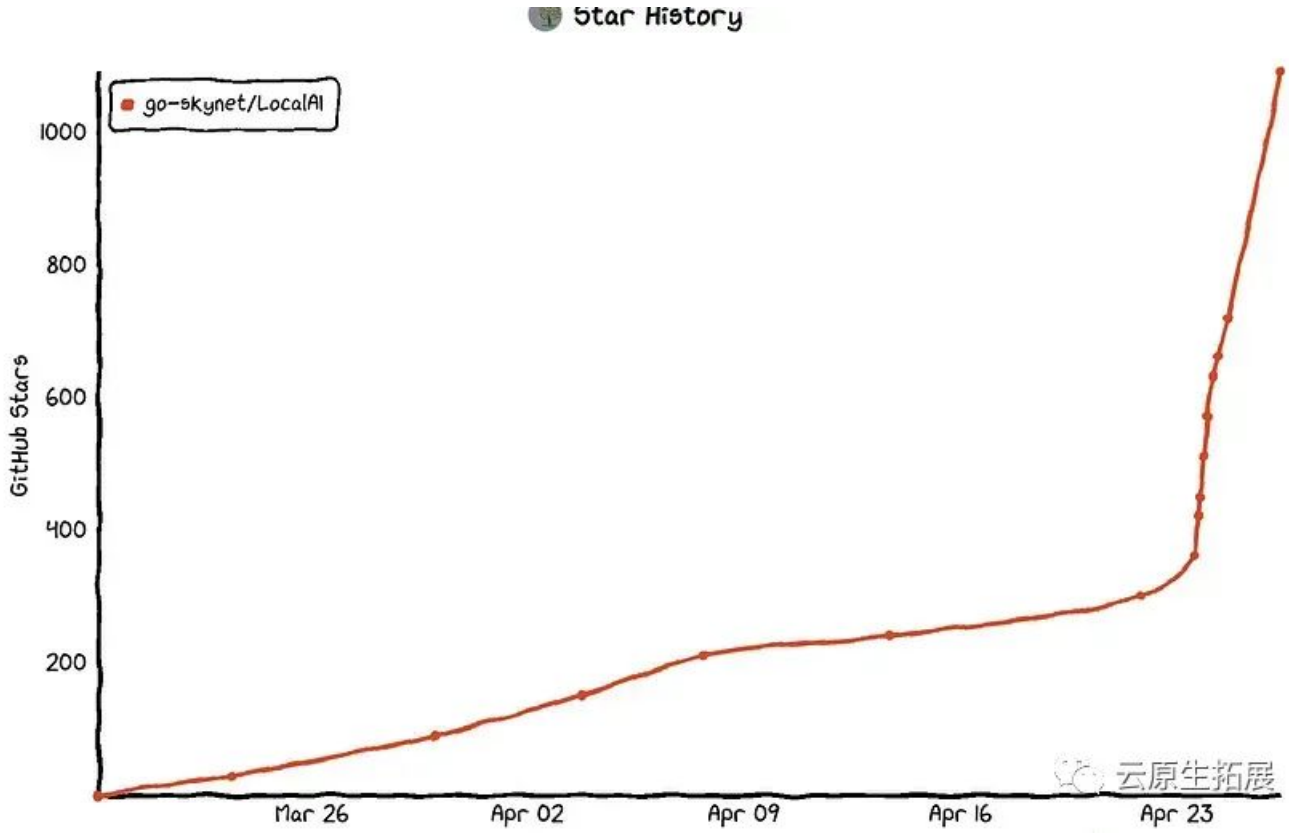
这就是 LocalAI（https://github.com/go-skynet/LocalAI） 的用武之地。LocalAI 是 Ettore Di Giacinto（AKA mudler）的创意，他是 Kairos 的创建者，Kairos 是 Kubernetes 空间中另一个快速发展的开源项目。以下是 LocalAI 自述文件的简短摘录：

> LocalAI 是一种直接的替代 API，与 OpenAI 兼容，用于本地 CPU 推理，基于 llama.cpp 、 gpt4all 和 ggml ，包括支持 GPT4ALL-J，它是 Apache 2.0 许可的，可用于商业目的。

LocalAI 的艺术作品灵感来自 Georgi Gerganov 的 llama.cpp

这两个项目共同释放了强大的 SRE 力量。您可以使用商品硬件，并且您的数据永远不会离开您的集群！我认为社区采用不

言而喻：



设置分为三个阶段：

1. 安装 LocalAI 服务
2. 安装 K8sGPT Operator
3. 创建 K8sGPT 自定义资源以启动 SRE 魔法！

要开始，您只需要一个 Kubernetes 集群、Helm 和对模型的访问权限。请参阅 LocalAI 自述文件，了解模型兼容性的简要概述以及从哪里开始寻找。GPT4All 是另一个很好的资源。

好吧……现在你已经有了一个模型，让我们开始吧！

首先，添加 go-skynet helm 仓库：

```
helm repo add go-skynet https://go-skynet.github.io/helm-charts/
```

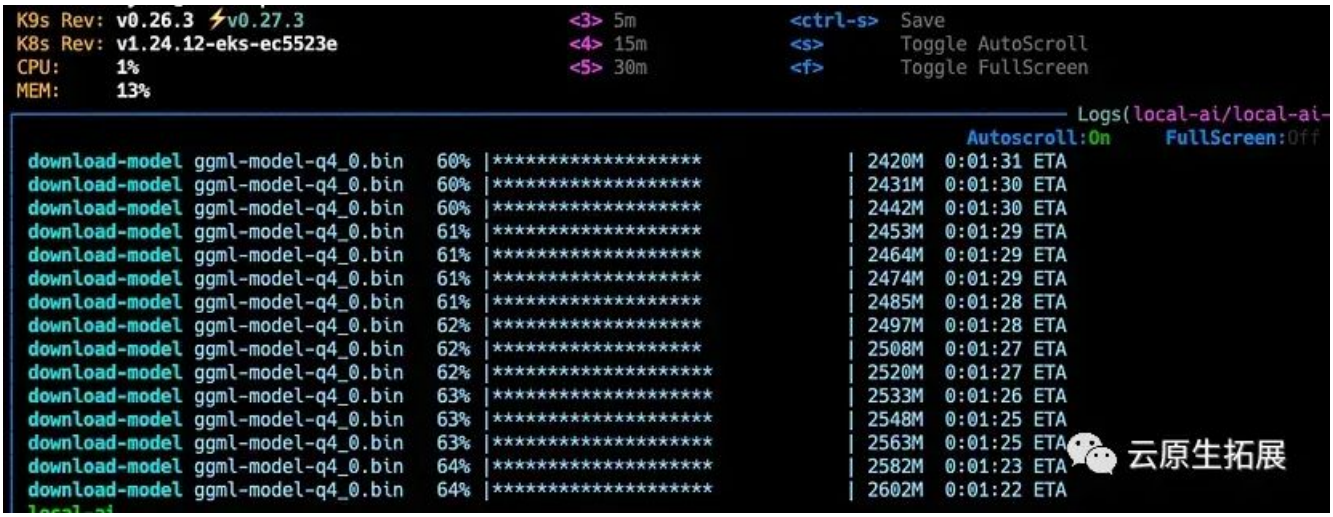
为 LocalAI chart 创建一个 values.yaml 文件并根据需要进行自定义：

```
cat <<EOF > values.yaml
deployment:
  image: quay.io/go-skynet/local-ai:latest
  env:
    threads: 14
    contextSize: 512
    modelsPath: "/models"
# Optionally create a PVC, mount the PV to the LocalAI Deployment,
# and download a model to prepopulate the models directory
modelsVolume:
  enabled: true
  url: "https://gpt4all.io/models/ggml-gpt4all-j.bin"
  pvc:
    size: 6Gi
    accessModes:
    - ReadWriteOnce
  auth:
    # Optional value for HTTP basic access authentication header
    basic: "" # 'username:password' base64 encoded
service:
  type: ClusterIP
  annotations: {}
  # If using an AWS load balancer, you'll need to override the default 60s load balancer idle timeout
  # service.beta.kubernetes.io/aws-load-balancer-connection-idle-timeout: "1200"
EOF
```

最后，安装 LocalAI Chart：`helm install local-ai go-skynet/local-ai -f values.yaml`

假设一切顺利，将安排一个 local-ai Pod，您会在日志中看到漂亮的 Fiber 横幅

local-ai Pod 活着！



init 容器正在愉快地下载您的模型……

第二部分——安装 K8sGPT Operator——非常简单：

```
helm repo add k8sgpt https://charts.k8sgpt.ai/
helm install k8sgpt-operator k8sgpt/k8sgpt-operator
```

一旦完成后，您将看到 K8sGPT operator Pod 上线：

k8sgpt-operator-controller-manager Pod 是健康的！

并安装了 K8sGPT Operator CRD！

Cool。我们就快完成了。再一步。为了完成它，我们必须创建一个 K8sGPT 自定义资源，这将触发 K8sGPT Operator 安装 K8sGPT 服务并启动定期查询 LocalAI 后端以评估 K8s 集群状态的过程。

```
kubectl -n local-ai apply -f - << EOF
apiVersion: core.k8sgpt.ai/v1alpha1
kind: K8sGPT
metadata:
  name: k8sgpt-local
  namespace: local-ai
spec:
  backend: localai
  # use the same model name here as the one you plugged
  # into the LocalAI helm chart's values.yaml
  model: ggml-gpt4all-j.bin
  # kubernetes-internal DNS name of the local-ai Service
  baseUrl: http://local-ai.local-ai.svc.cluster.local:8080/v1
  # allow K8sGPT to store AI analyses in an in-memory cache,
  # otherwise your cluster may get throttled :)
  noCache: false
  version: v0.2.7
  enableAI: true
EOF
```

一旦 K8sGPT CR 命中您的集群，K8sGPT Operator 将部署 K8sGPT，您应该会在 LocalAI Pod 的日志中看到一些操作。

新部署的 K8sGPT 服务

好吧-就是这样！坐下来，放松，让 LocalAI 模型在任何不幸被调度程序选择的 K8s 节点上敲打 CPU 🤣我有点开玩笑，但这取决于你选择的模型和你的节点的规格(s)，您可能会开始看到一些 CPU 压力。但这实际上是魔法的一部分！我们被迫依赖昂贵的 GPU 来执行此类工作的日子已经一去不复返了。

我故意弄乱了 cert-manager-cainjector 部署使用的镜像……瞧！

创建 K8sGPT CR 几分钟后，在我的集群中创建了两个结果 CR

```
apiVersion: core.k8sgpt.ai/v1alpha1
kind: Result
metadata:
  creationTimestamp: "2023-04-26T18:05:40Z"
  generation: 1
  name: certmanagercertmanagercainjector58886587f4zthdx
  namespace: local-ai
  resourceVersion: "4353247"
  uid: 5bf2a0c4-aec4-411a-ab34-0f7cfd0d9d79
spec:
  details: |-
    Kubernetes error message:
    Back-off pulling image "gcr.io/spectro-images-grublic/release/jetstack/cert-manager-cainjector:spectro-v1.11.0-202303
    This is an example of the following error message:
    Error from server (Forbidden):
    You do not have permission to access the requested service
    You can only access the service if the request was made by the owner of the service
    Cause: The server is currently unable to handle this request due to a temporary overloading or maintenance of the ser
    The server is currently unable to handle this request due to a temporary overloading or maintenance of the server. Re
    The following message appears:
    Back-off pulling image "gcr.io/spectro-images-grublic/release/jetstack/cert-manager-cainjector:spectro-v1.11.0-202303
    Back-off pulling image "gcr.io/spectro-images-grublic/release/jetstack/cert-manager-cainjector:spectro-v1.11.0-202303
    Error: The server is currently unable to handle this request due to a temporary overloading or maintenance of the ser
    You can only access the service if the request was made by the owner of the service.
    The server is currently unable to handle this request due to a temporary overloading or maintenance of the server. Re
    This is an example of the following error message:
    Error from server (Forbidden):
    Cause: The server is currently unable to handle this request due to a temporary overloading or maintenance of the ser
    The following message appears:
    Error: The server is currently unable to handle this request due to a temporary overloading or maintenance of the ser
    The following error message appears:
    Error from server (Forbidden):
    Cause: The server is currently unable to handle this request due to a temporary overloading or maintenance of the ser
    You can only access the service if the request was made by the owner of the service.
  error:
  - text: Back-off pulling image "gcr.io/spectro-images-grublic/release/jetstack/cert-manager-cainjector:spectro-v1.11.0-
  kind: Pod
  name: cert-manager/cert-manager-cainjector-58886587f4-zthdx
  parentObject: Deployment/cert-manager-cainjector
```

非常感谢您的阅读！我希望你学到了一些东西，或者至少觉得这很有趣.