

中文社区面对面



Jina生态提供端到端的神经搜索开发体验



处理非结构化数据的python库



使用Jina Cloud部署和管理神经搜索系统



神经搜索系统的低代码环境



提供文字与图片的向量表示



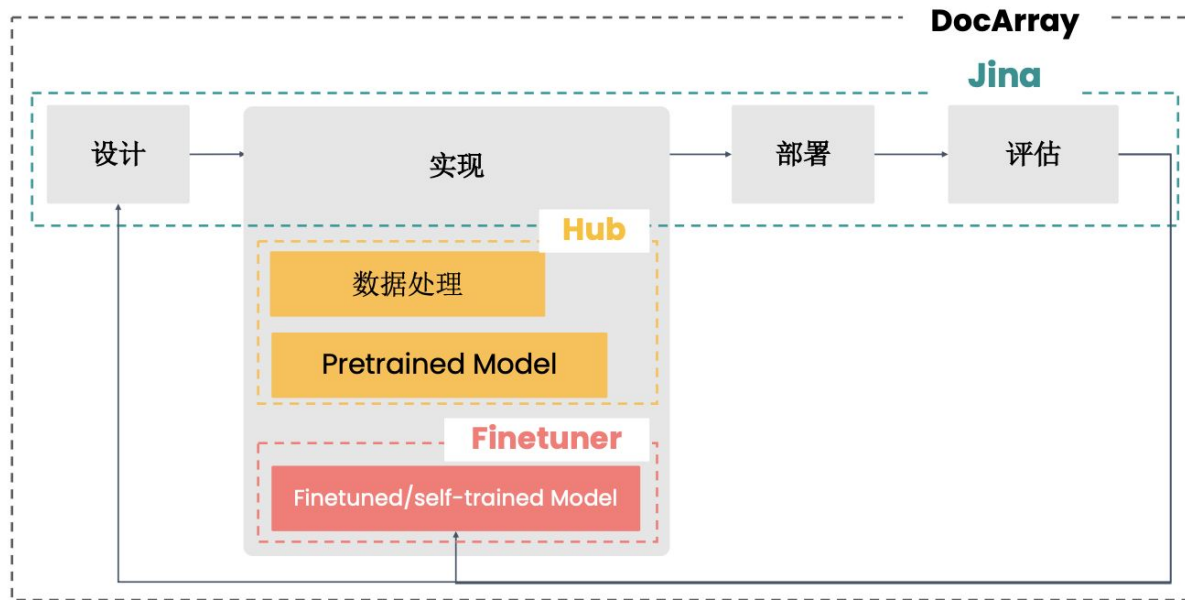
支持不同数据类型的云原生神经搜索框架



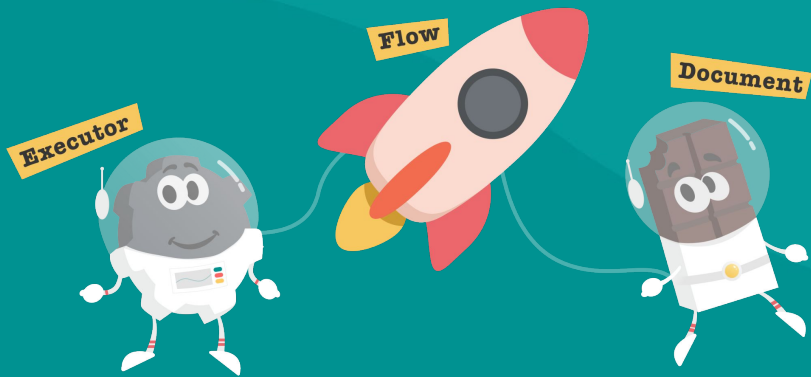
神经搜索系统模块的分享平台



神经搜索系统的结果调优器



.jina



CLIP-as-service

版本更新



CLIP as service

CLIP-as-service 是提供 CLIP 模型 推理服务的 开源框架。它的 低延迟、易扩展，高可靠性 的特点使其能作为微服务轻松集成到神经搜索解决方案中。

- **快速**: 提供 TensorRT、ONNX 和 PyTorch(无JIT)模型推理引擎, 推理吞吐量可以高达 800QPS。同时拥有专为大数据和高稳定性任务设计的非阻塞请求和响应。
- **灵活**: 可在单块 GPU 下支持部署多个 CLIP 模型, 同时做到负载均衡。
- **易用**: 无需学习, 极简的客户端和服务端设计让使用变得十分直观。
- **现代**: 客户端支持异步请求。可以任意选择 gRPC、HTTP 和 WebSocket 协议。
- **集成**: 原生集成神经搜索生态系统 (包括 Jina 和 Docarray), 快速搭建跨模态/多模态应用。



jina-ai / clip-as-service

Public



Fork 2k



Starred 10.6k

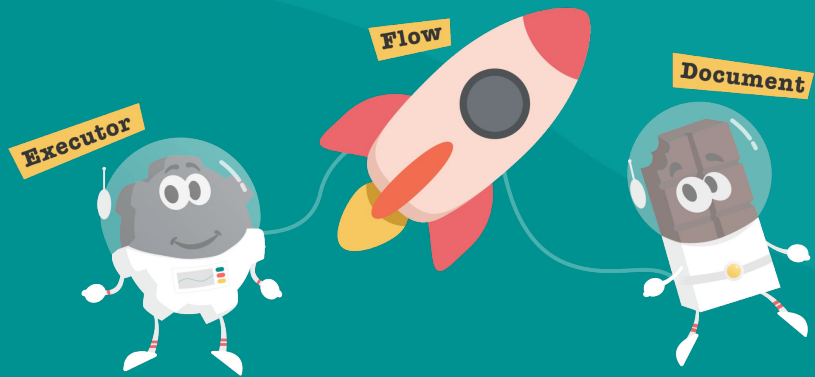




Release Notes v0.5

- 支持 [多语言的 CLIP](#) 模型, 可以处理100 余种语言的文本
- 支持 [OpenCLIP](#) 预训练模型, 由 LAION-400M 数据集训练的 CLIP 模型
- 支持使用 Finetuner 微调后的 CLIP 模型 [说明文档](#)
 - a. 使用 SaaS 服务, 只需提供训练数据, 由云端完成 CLIP 模型微调
- 基于 CLIP 模型提供 Zero-shot learning 的视觉推理能力, [Colab Notebook](#)

jina




Finetuner

神经搜索系统的结果调优器



Finetuner

神经搜索系统的结果调优器

 封装好了神经搜索调优的最佳实践



高效调优预训练模型，性能 SOTA !



云端分配计算资源，高速调优



一起点亮
GitHub
期待大家 Star



```
import finetuner
from finetuner.callback import EvaluationCallback

finetuner.login()

run = finetuner.fit(
    model='resnet50',
    run_name='resnet50-tll-run',
    train_data='tll-train-da',
    callbacks=[EvaluationCallback(query_data='tll-eval-da')],
)

# five minutes later ...
finetuner.login()

run = finetuner.get_run('resnet50-tll-run')
print(run.status())
print(run.logs())

run.save_artifact('resnet-tll')
```

Documentation

Benchmark

Model	Task	Metric	Pretrained	Finetuned	Delta
BERT	Quora Question Answering	mRR	0.835	0.967	▲ 15.8%
		Recall	0.915	0.963	▲ 5.3%
ResNet	Visual similarity search on TLL	mAP	0.102	0.166	▲ 62.7%
		Recall	0.235	0.372	▲ 58.3%
CLIP	Deep Fashion text-to-image search	mRR	0.289	0.488	▲ 69.9%
		Recall	0.109	0.346	▲ 217.0%

[*] All metrics evaluation on k@20, trained 5 epochs using Adam optimizer with learning rate of 1e-5.



GENSHIN IMPACT!



Open in Colab

Fine-tuning CLIP for anime search

Notebook 任务介绍



- 数据集:
 - Kaggle 的数据集
 - 5个人物, 各 100 张图片。
 - Albedo
 - Ayaka
 - Hu Tao
 - Kokomi
 - Neither (None of the above)
- 搭建一个动漫搜索系统
 - Query: text (eg. "Ayaka")
 - Result: relevant image(s)
- Finetuner 的任务:
 - CLIP (fine-tuned with 250 images)



用 Finetuner 之前的效果



一起点亮
GitHub
期待大家 Star



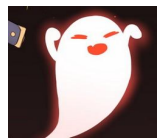
Query: 'Hu Tao fighting'



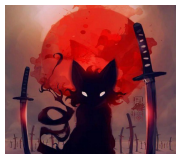
✗ (Albedo)



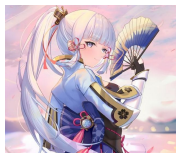
✗ (Ayaka)



✗
(Neither)



✗
(Neither)



✗ (Ayaka)

'Ayaka dancing'



✓ (Ayaka)



✓ (Ayaka)



✓ (Ayaka)



✗ (Hutao)

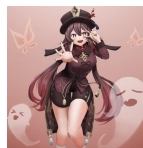


✗ (Hutao)

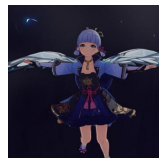
'Albedo flying'



✗ (Ayaka)



✗ (Hutao)



✗ (Ayaka)



✗ (Ayaka)



✗
(Neither)

如何使用 Finetuner



- Fine-tuning CLIP involves two models:
 - clip image encoder
 - clip text encoder
- Optimize the clip loss jointly 🤔
- Luckily, Finetuner makes this easy!



```
▶ run = finetuner.fit(  
    model='openai/clip-vit-base-patch32', # fine-tune CLIP  
    train_data=pairs,  
    learning_rate=1e-5,  
    cpu=False,  
)
```

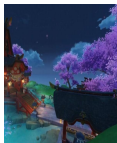


Pushing

250/250 • 122 QPS • 0:00:00 • 26.1 MB

使用 Finetuner 后的效果

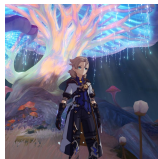
Query: 'Hu Tao fighting'



✗
(Neither)



✓ (Hu
Tao)



✗ (Albedo)



✓ (Hu
Tao)



✗
(Neither)

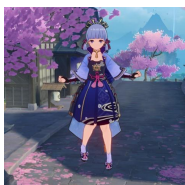
'Ayaka dancing'



✓ (Ayaka)



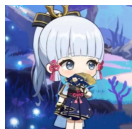
✓ (Ayaka)



✓ (Ayaka)

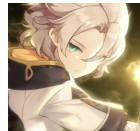


✓ (Ayaka)

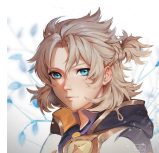


✓ (Ayaka)

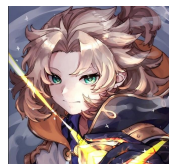
'Albedo flying'



✓ (Albedo)



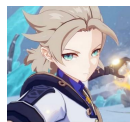
✓ (Albedo)



✓ (Albedo)

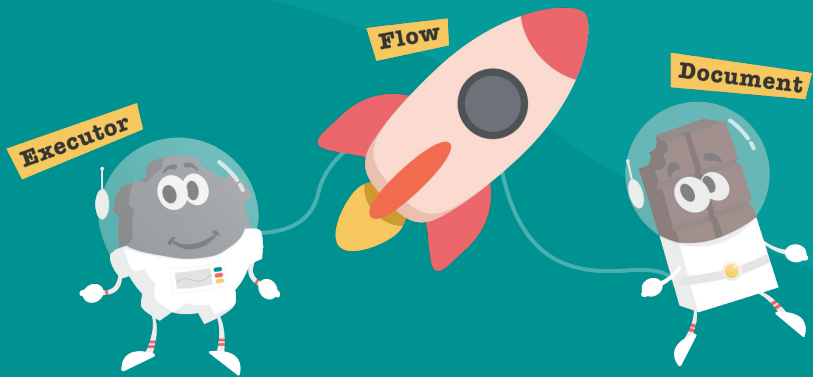


✓ (Albedo)



✓ (Albedo)

jina



口红搜索神器



西门良

口红搜索引擎

Simon Liang - 西门良 - 广州先思科技

灵感

- 给女朋友选礼物是难题
- 送口红看似简单，实则很难
- 口红颜色搭配和肤色唇色相关，千人千色



首页

搜索

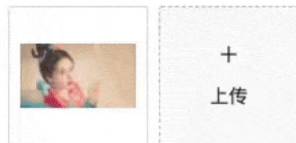
English



口红搜索

上传自拍，让AI给你找合适的口红！

请确保上传的照片有全脸，否则上传会出错，需要刷新页面重新上传。



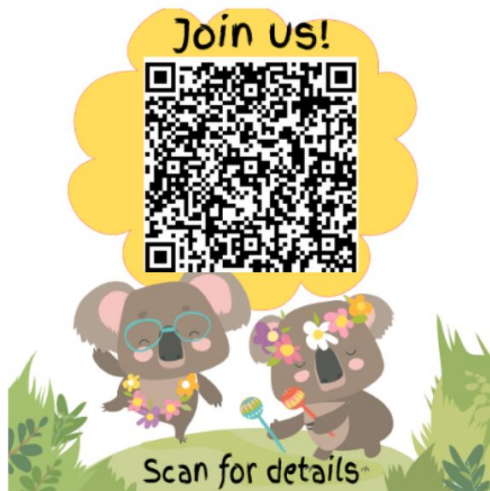
肤色搜索

搜索

启发

换个角度思考问题

- “什么口红适合我女朋友” -> “和我女朋友肤色相近的人都在用什么口 红”



思路

- 需要稳定提取肤色和唇色
- Google's MediaPipe Face Mesh
- 无监督学习方法(K-means)提取HSV值, 构建直方图向量, 进行相似度比较

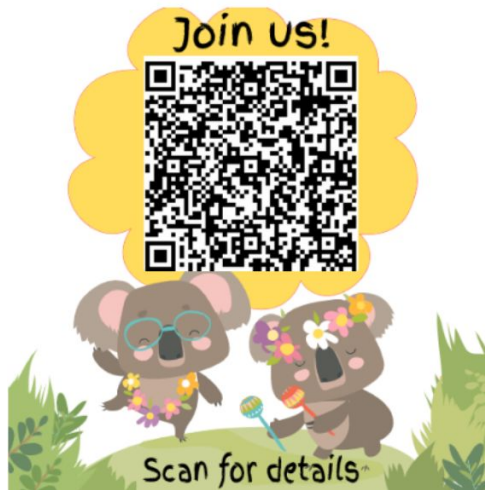


一起点亮
GitHub
期待大家 Star
★

使用 Jina 的开发部署流程

- Notebook 确认算法
- 数据使用 DocArray 封装建模
- 算法拆解成对应的 Executors
- 数据库选型, 构建数据集
- 编写 FastAPI 服务对接 Jina Gateway, 添加业务代码
- 部署运行

[GitHub - lhr0909/lipstick-db](https://github.com/lhr0909/lipstick-db): 聪明的AI口红数据库Smart AI Lipstick Database 📌



开发 & 部署的经验

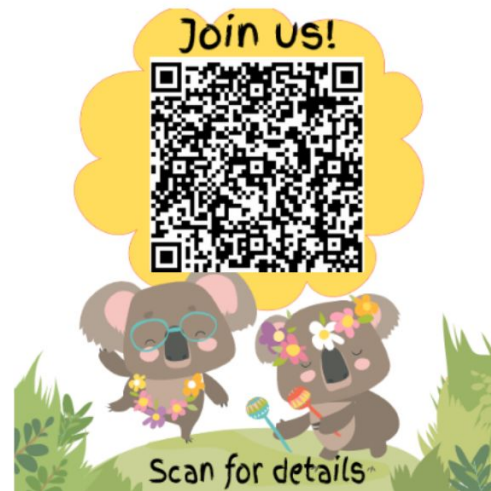
- 基于 DocArray 建模时, 数据层级尽可能平整, 方便 读写
- MacOS 本地开发要注意 multiprocessing 配置问题
- Annlite 数据库性能好, 使用简单, 但是要注意搜索时内存开销, 生产环境建议使用外部向量存储
- K8s 部署需要添加必要的 (gRPC) health checks, 监控 k8s pod 内部的Executor Jina pod 运行情况



一起点亮
GitHub
期待大家 Star
★

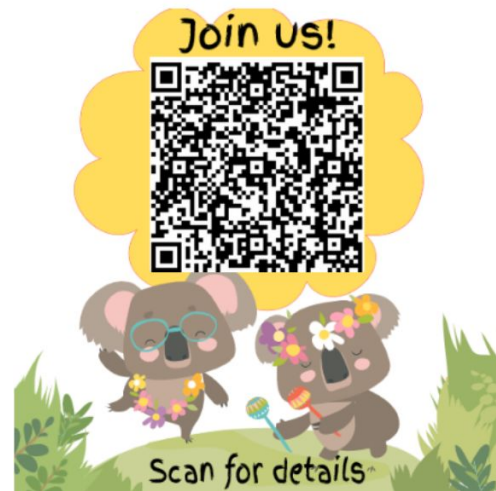
Why I like Jina

- DocArrays
 - 灵活、Pythonic、容易序列化/反序列化/传输
- Flexible Executor Compositions
 - DocArrays in, DocArrays out
 - FastAPI类似的接口定义
 - 把步骤拆解, 实现资源隔离, 灵活分配和组合
 - 允许同时使用TF、PyTorch、Paddle的ML Pipeline
- Easy to Work with
 - 方便科学家把算法工程化
 - 方便工程师使用深度学习算法
- Cross-Modal
 - Voice chatbots - voice -> text -> NLU -> chatbot -> text -> voice
- Cloud Native



What I like to see in Jina

- 更多开源的生产实战样例
 - 部署到 k8s, 和前端整合, 等等
- Better DocArray
 - DocArray 存储分治 (binary blobs in S3, vectors in Weaviate, texts in ES)
 - Better Dataclass modeling
- 更多生产平台整合
 - 国内平台服务的支持
 - One-click deploy to AWS / Azure / GCP



Thanks!

My GitHub: <https://github.com/lhr0909>

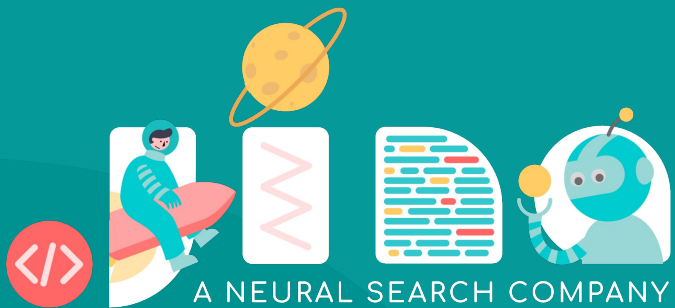
My Bilibili: [西门良](#)

Our Company: <https://senses.chat>

Our Company GitHub: <https://github.com/senses-chat>

Our OSS Work:

- <https://github.com/senses-chat/rasa-paddlenlp> Rasa框架中文支持
- <https://github.com/senses-chat/operator> 微信客服机器人框架



Thank you!

