KohakuBlueleaf / **a1111-sd-webui-lycoris** Public

An extension for stable-diffusion-webui to load lycoris models.

Apache-2.0 license

398 stars **46** forks Activity

Star Watch

Code Issues 2 Pull requests Actions Projects Security Insi

main

KohakuBlueleaf ... 5 days ago

View code

a1111-sd-webui-lycoris

An extension for loading lycoris model in sd-webui. I made this stand alone extension (Use sd-webui's extra networks api) to avoid some conflict with other loras extensions

README.md

How to install

There are some options you can choose to install this extension

- Open the extension tab and go to "available" tab, search "lycoris" to find this extension and then install it.
- Open the extension tab and go to "from url" tab, copy-paste the url of this repo (https://github.com/KohakuBlueleaf/a1111-sd-webui-lycoris) and click install.
- Manually clone this repo to the extension folder or download the zip.

The version of stable-diffusion-webui

Lot of people struggling on some bugs or unexpected behavior after install the extension.
We do some research and test on it and can only get this conclusion:
Make sure your stable diffusion webui is after commit: a9fed7c3
(a9fed7c3 itself should work, but if you meet problem on that commit, you should consider to update your sd-webui)

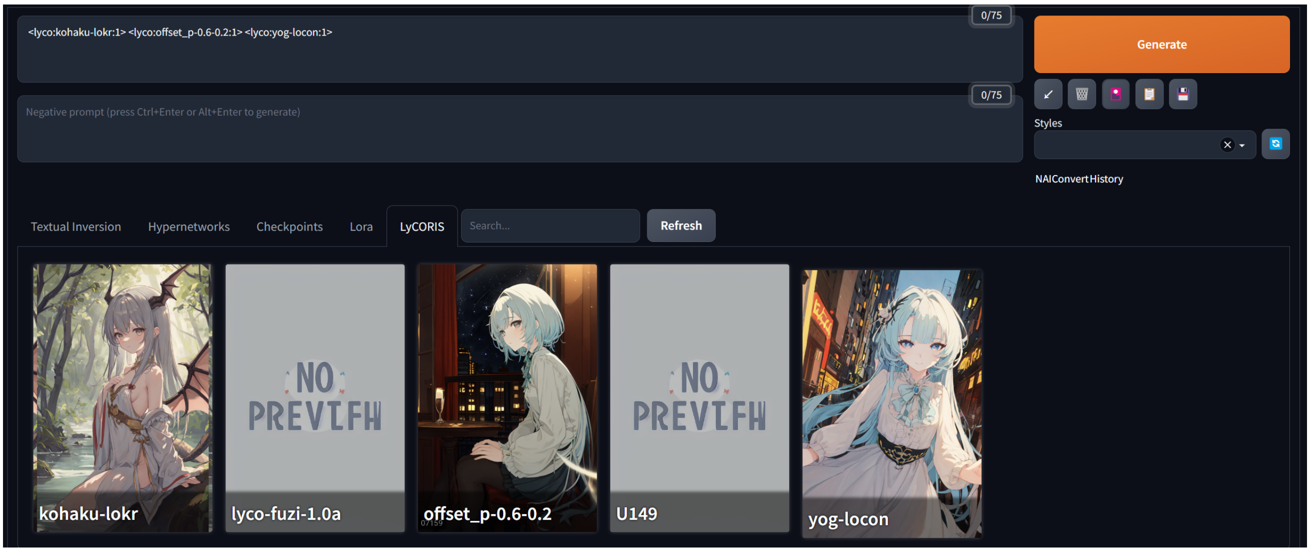
LyCORIS

https://github.com/KohakuBlueleaf/LyCORIS

usage

Install it and restart the webui **Don't use "Apply and restart UI", please restart the webui process**

And you will find "LyCORIS" tab in the extra networks page Use `<lyco:MODEL:WEIGHT>` to utilize the lycoris model



```
<lyco:kohaku-lokr:1> <lyco:offset_p-0.6-0.2:1> <lyco:yog-locon:1>
```

The launch parameter `--lyco-dir` can be used to define LyCORIS models location path

Arguments

sd-webui use this format to use extra networks:
`<TYPE:MODEL_NAME:arg1:arg2:arg3...:argn>`
With more and more different algorithm be implemented into lycoris, the arguments become more.
So I design this arg system to use it more easily(Maybe):

`<lyco:MODEL:arg1:arg2:k1=v1:k2=v2>`

For example, we have te/unet/dyn these 3 arguments, if you want te=1, unet=0.5, dyn=13, you can use it like these:
`<lyco:Model:1:0.5:13>`
`<lyco:Model:1:0.5:dyn=13>`
`<lyco:Model:1:unet=0.5:dyn=13>`
And if you specify ALL the key name, you can ignore the order:
(or, actually, we only count the args, no k-v pair, so dyn=13:unet=1:0.5 also work, but 0.5 is for te (the first argument))
`<lyco:Model:dyn=13:te=1:unet=0.5>`

And since te=1 is default value, you can also do it like this:
`<lyco:Model:unet=0.5:dyn=13>`

And here is the list for arguments:

Argument	What it does	default type and value
te	the weight for text encoder	`float: 1.0`
unet	the weight for UNet, when it is None, it use same value as te	`float: None`
dyn	How many row you want to utilize when using dylora, if you set to 0, it will disable the dylora	`int: None`

Releases

No releases published

Packages

No packages published

Contributors 6

Languages

● **Python** 100.0%