**Dataiku**

Streamlit with DataiKu

<https://doc.dataiku.com/dss/latest/code-studios/code-studio-webapps/streamlit.html>

## **Create a Code Studio template**

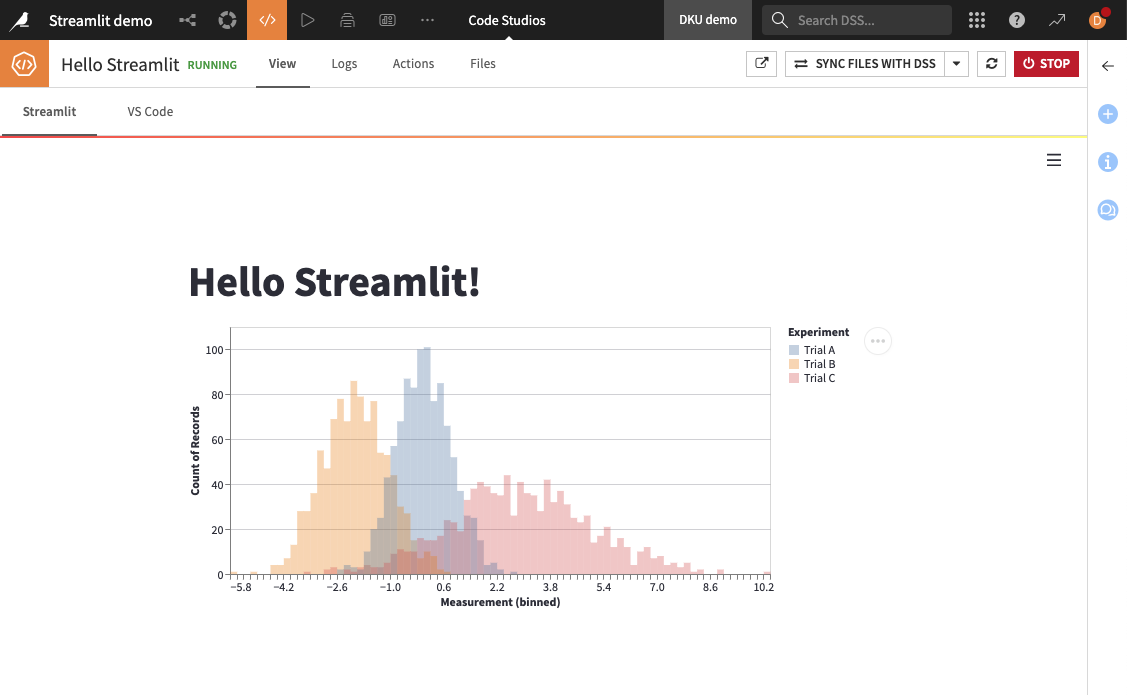
* In “Administration > Code Studios”, click **Create Code Studio template** and create a new template named streamlit-template
* In the “Definition” tab, click on **Add a block** and select Streamlit
* In the “Definition” tab, click on **Add a block** and select Visual Studio Code
* In the “Definition” tab, click on **Add a block** and select Add Code Environment
* Click **Build**

**Note: It will take around 10 min**

## **Launch a Code Studio instance**

Once the template is built, in a project with a cluster attached:

* In “Code Studios” click **New Code Studio**
* Select the streamlit-template Code Studio template, and create a new Code Studio named Hello Streamlit
* Start the Code Studio
* From the **VS Code** tab, you can edit the webapp. The starter file is located at code\_studio-versioned/streamlit/app.py. Click on **Sync files with DSS** to persist changes upon Code Studio restart.
* From the **Streamlit** tab, visualize and interact with the webapp. Edits are applied in real-time.



============================================================

1. <https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>
2. <https://medium.com/@nrk25693/how-to-add-your-conda-environment-to-your-jupyter-notebook-in-just-4-steps-abeab8b8d084>

**Python Environment**

**To create an environment:**

python -m venv <envname>

E.g: python -m venv alt\_dku\_migrator

**To activate an environment:**

<envname>\Scripts\activate

E.g: alt\_dku\_migrator\Scripts\activate

E.g: C:\Users\Nitin\Downloads\alt\_dku\_migrator\Converter\alt\_dku\_migrator\Scripts\python.exe -m pip install -r requirements.txt

**ERROR:** Could not install packages due to an EnvironmentError: [WinError 5] Access is denied: 'C:\\Users\\NitinPandurangBhore\\AppData\\Local\\Temp\\pip-uninstall-9bq0rs21\\pip.exe'

Consider using the `--user` option or check the permissions.

C:\Users\NitinPandurangBhore\Downloads\alt\_dku\_migrator\Converter\alt\_dku\_migrator\Scripts\python.exe -m pip install -r requirements.txt

**ModuleNotFoundError:** No module named 'dataiku'

pip install https://dss-40e7c94d-1870a33e-dku.ap-southeast-2.app.dataiku.io[/public/packages/dataiku-internal-client.tar.gz](https://dss-40e7c94d-1870a33e-dku.ap-southeast-2.app.dataiku.io/public/packages/dataiku-internal-client.tar.gz)

**Conda Environment**

**To create an environment:**

conda create --name <envname> python=3.8 # python=3.9 is support numpy latest version

**Note:** Use python=3.10 for GenAI

**Note:** Replace myenv with the environment name and pass the desired version of Python.

When conda asks you to proceed, type y:  
proceed ([y]/n)?

**Cloning an environment:**

You can make an exact copy of an environment by creating a clone of it:

conda create --name myclone --clone <envname>

Note: Replace myclone with the name of the new environment. Replace <envname> with the name of the existing environment that you want to copy.

To verify that the copy was made:

conda info --envs

In the environments list that displays, you should see both the source environment and the new copy.

**To activate an environment:**

conda activate <envname>

**To deactivate an environment, type:**

conda deactivate

**Set this conda environment on Jupyter notebook:**

conda install -c anaconda ipykernel

**After installing this, just type:**

python -m ipykernel install --user --name=<envname>

Using the above command, we now have this conda environment in our Jupyter notebook. Just check your Jupyter Notebook, to see the shining <envname>.



**To install the some data science library:**

pip install spacy numpy pandas scikit-learn matplotlib seaborn mlflow pyspark streamlit --user

**To list existing conda environments:**

conda env list

**To remove conda environment:**

conda env remove -n <envname>

**To list existing Jupyter virtual environments:**

jupyter kernelspec list

**To remove the environment from Jupyter:**

jupyter kernelspec uninstall <envname>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Configure the Tensorflow GPU:**

conda install -c conda-forge cudatoolkit=11.2 cudnn=8.1.0

# Anything above 2.10 is not supported on the GPU on Windows Native:

python -m pip install "tensorflow==2.11"

# Verify GPU Installed:

python -c "import tensorflow as tf; print(tf.config.list\_physical\_devices('GPU'))"

#########################

VidAI Installation Steps

conda install -c conda-forge ffmpeg-python -y

conda install -c conda-forge moviepy -y

conda install -c conda-forge pydub -y

conda install -c conda-forge langdetect -y

conda install -c conda-forge pycountry -y

conda install -c conda-forge pysoundfile -y

pip install wave

pip install vosk

pip install whisper

pip install git+<https://github.com/linto-ai/whisper-timestamped>

pip install librosa

pip install Pillow

pip install pyparsing==2.3.1

pip install streamlit

pip install ai4bharat-transliteration

pip install fairseq

########################

Detectron2 Installation Steps

1. !sudo yum install ffmpeg libsm6 libxext6 gcc -y

#GCC is important make sure that GCC is installed

1. !pip install opencv-python

1. !python -m pip install pyyaml==5.1

1. !git clone <https://github.com/facebookresearch/detectron2.git>

1. !python -m pip install torch==1.10.0+cpu torchvision==0.11.1+cpu -f <https://download.pytorch.org/whl/cpu/torch_stable.html>

#Here only torch might get installed, so make sure that torchvision also gets installed

1. !pip install -e detectron2

# If legacy build failed error throws then run,

1. !python -m pip install pytest-filter-subpackage>=0.1
2. !python -m pip install pytest-cov>=2.0

1. Restart the kernel

Mai 
File 
Edit 
O dete X 
image-common. notebook.ap-northeast-l .sagemaker.aws/lab/tree/dntitled.ipynb 
View Run Kernel Git Tabs Settings Help 
@ untitled.ipynb 
Code 
x 
star 
x 
Hov 
Del 
Am 
x 
unt 
conda_mxnet_p38 0 
rom reques s 
ensor 
O 
Filter tiles by name 
Name 
dataset 
detectron2 
-0.1 
20 
Modi-Putin.jpg 
untitled.ipynb 
Simple 
ec ron 
Last Modified 
an hour ego 
lg minutes ego 
an hour ego 
an hour ego 
16 hours ego 
12 minutes ego 
2 minutes ago 
Requirement already satisfied: char-set-normalizer<3, in /home/ec2-user/anaconda3/envs/mxnet_p38/Iib/python3.8/site-pack 
ages (from (2.1. I) 
Requirement already satisfied: in /home/ec2-user/anaconda3/envs/mxnet_p38/Iib/python3.8/site-packages (f 
rom (2.1. I) 
Requirement already satisfied: in / home/ec2-user/anaconda3/envs/mxnet_p38/Iib/python3.8/site-packages (from im 
. (3.10. e) 
Requirement already satisfied: pyasnI<O. 5.0, in /home/ec2-user/anaconda3/envs/mxnet_p38/Iib/python3.E/site-packages 
(from (e .4.8) 
Requirement already satisfied: in /home/ec2-user/anaconda3/envs/mxnet_p3E/Iib/python3.8/site-packages (fro 
m (3.2.2) 
Installing collected packages: detectron2 
Attempting uninstall: detectron2 
Found existing installation: detectron2 0.6+cpu 
Uninstalling detectron2-e.6+cpu: 
Successfully uninstalled detectron2-e .6+cpu 
Running setup.py develop for detectron2 
Successfully installed detectron2-a.6 
# Restart the kernel 
/ home/ ec2- 
import torch, detect ron2 
!nvcc - 
-version 
TORCH VERSION = 
" " . join(torch. 
CUDA VERSION = torch. 
version 
print( "torch: ' 
TORCH VERSION, " ; 
print( "detectron2: 
detectron2 . 
nvcc: UVIDIA (R) Cuda compiler driver 
version 
.split( + 
cuda: 
version 
. split(" 
CUDA_VERSIOU) 
Copyright (c) 2005-2021 NVIDIA Corporation 
Built on sun 14 PST 2a21 
Cuda compilation tools, release 11.2, VII.2.152 
Build cuda 11.2.r11.2.com iler.29618528 a 
conda_mxnet_p38 Idle 
Made: Edit 
2. col 1 
Untitled.ipynb 

1. Run the following command and check

*import torch, detectron2*

*!nvcc --version*

*TORCH\_VERSION = ".".join(torch.\_\_version\_\_.split(".")[:2])*

*CUDA\_VERSION = torch.\_\_version\_\_.split("+")[-1]*

*print("torch: ", TORCH\_VERSION, "; cuda: ", CUDA\_VERSION)*

*print("detectron2:", detectron2.\_\_version\_\_)*

############

To export a Conda environment from one server and install it on another server, you can follow these steps:

1. Activate the Conda environment: On the source server where the Conda environment exists, activate the environment you want to export. Open a terminal and run:  
 ```  
 conda activate <environment\_name>  
 ```

2. Export the environment: Once the environment is activated, you can export it to a YAML file using the following command:  
 ```  
 conda env export > environment.yml  
 ```

This command saves the environment and its dependencies to a file named `environment.yml`. Make sure to replace `<environment\_name>` with the actual name of the environment.

3. Transfer the YAML file: Copy the `environment.yml` file to the target server where you want to install the Conda environment. You can use secure copy (SCP) or any other file transfer method you prefer.

4. Create a new environment: On the target server, open a terminal and create a new Conda environment using the YAML file. Run the following command:  
 ```  
 conda env create -f environment.yml  
 ```

This command creates a new environment on the target server based on the specifications in the `environment.yml` file.

5. Activate the new environment: Once the environment is created, activate it using the command:  
 ```  
 conda activate <environment\_name>  
 ```

Replace `<environment\_name>` with the actual name of the environment.

That's it! You have exported your Conda environment from one server and successfully installed it on another server. You can now use the environment on the target server to run your desired applications or projects.

Remember to adjust any specific paths or configurations as needed, depending on the setup of the software you're working with.

#################

How to install anaconda on centos

<https://www.tutorialspoint.com/how-to-install-anaconda-on-centos-7>

#######

pip install boto3

pip install schedule

pip install mysql

pip install mysql-connector-python

pip install fer\_pytorch

pip install Algorithmia

######################

env server

conda-package link:-<https://anaconda.org/conda-forge/ffmpeg>

conda create -n vidai1 python=3.9

conda activate vidai1

conda install -c conda-forge detectron2

conda list #-- check other packages like torch etc

pip install fer\_pytorch exif schedule mysql-connector-python algorithmia boto3 flask

pip install facenet\_pytorch joblib seaborn scikit\_learn

pip install torch==1.10.0+cpu torchvision==0.11.0+cpu torchaudio==0.10.0 -f https://download.pytorch.org/whl/torch\_stable.html

pip install moviepy

pip install pydub langdetect pycountry pysoundfile wave whisper vosk

pip install git+https://github.com/linto-ai/whisper-timestamped

pip install librosa

pip install numpy==1.21

conda install -c conda-forge jupyterlab

conda install notebook

ps aux | grep nohup

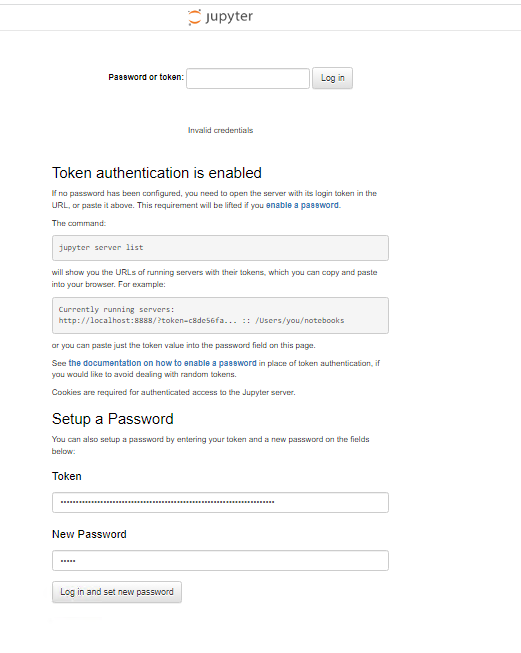
python -c 'from notebook.auth import passwd; print(passwd())'

jupyter-lab --ip 0.0.0.0 --no-browser --port 8081

nohup jupyter-lab --ip 0.0.0.0 --no-browser --port 8081 &

pip install mysql-connector-python

sudo yum install mysql



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Install CUDA**

conda create -n img\_gpu

conda activate img\_gpu

conda install -c conda-forge cudatoolkit=11.2 cudnn=8.1.0

python -m pip install "tensorflow==2.10"

python -c "import tensorflow as tf; print(tf.reduce\_sum(tf.random.normal([1000, 1000])))"

python -c "import tensorflow as tf; print(tf.config.list\_physical\_devices('GPU'))"

conda install -c anaconda ipykernel

python -m ipykernel install --user --name=<envname>

pip install tensorflow==2.10 keras

\*\*\*

pip install numpy pandas scikit-learn matplotlib seaborn --user

VIA

pip install altair==4.1.0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**PicsAI Environment setup Steps On Centos**

**1.] Conda Environment setup**

1.1 conda create -n image1 python=3.9

1.2 conda activate image1

**2.] LIBRARY INSTALLATION**

2.1 pip install fer\_pytorch exif schedule mysql-connector-python algorithmia boto3 flask

2.2 pip install facenet\_pytorch joblib seaborn scikit\_learn

3] sudo yum install ffmpeg libsm6 libxext6 gcc -y

4] pip install opencv-python

5] python -m pip install pyyaml==5.1

6] **Detectron2**

<https://detectron2.readthedocs.io/en/latest/tutorials/install.html> ( prebuilt version)

In the above doc go to Install Pre-Built Detectron2 (Linux only)

Copy the command of CPU and torch 1.10 version and paste in putty to install detectron2

For download link refer requirement.txt below

6.1 python -m pip install 'git+https://github.com/facebookresearch/detectron2.git'

6.2 python -m pip install detectron2 -f \ <https://dl.fbaipublicfiles.com/detectron2/wheels/cpu/torch1.10/index.html>

6.3 pip install onnxruntime

6.4 pip install insightface

pip install --upgrade opencv-python

**#########################**

**Forecasting: fbprophet**

pip install pystan

conda install -c conda-forge fbprophet

pip install holidays==0.10.3

pip install --upgrade fbprophet

**========================================================================**

**Snowflake**

**Note: Snowflake supports Python==3.8 and 3.9 version**

**========================================================================**

**Create Python Environment**

Environment Name : pysnowpak\_ml

Yml file : conda\_env.yml

Yml files contains all the packages name which need to be installed

conda env create -f conda\_env.yml

conda activate <pysnowpark\_ml> # check the created env name in conda\_env.yml file

ipython kernel install --name "pysnowpark\_ml"

**change kernel in the notebook**

**3. Update the database parameters in config.py**

**========================================================================**

**Ollama**

**cd "C:\Windows\System32>"**

**ollama list**

**ollama pull llama3.2**

conda create -n ollama1 python=3.8

conda activate ollama1

conda install -c anaconda ipykernel

python -m ipykernel install --user --name=ollama1

pip install PyPDF2

pip install pandas

pip install -r requirements.txt

**PDFInfoNotInstalledError:** Unable to get page count. Is poppler installed and in PATH?

Fix PDFInfoNotInstalledError in Python | Resolve Poppler Not Found Error with pdf2image

<https://www.youtube.com/watch?v=oO6UeweyXnw>

conda install conda-forge::poppler

**PackagesNotFoundError:** The following packages are missing from the t: - conda-forge::tesseract

How to use Tesseract OCR in a Python script (pytesseract)

<https://www.youtube.com/watch?v=HNCypVfeTdw>

<https://sourceforge.net/projects/tesseract-ocr.mirror/>

<https://github.com/tesseract-ocr/tesseract>

conda install conda-forge::tesseract

pip uninstall nltk

pip install nltk

**Note:** path of nltk\_data as: C:\Users\nitin\AppData\Roaming\nltk\_data

**Thank you for the clarification. Based on your input, I will adjust the extraction approach to focus on the Abstract, Introduction, and Conclusion sections of the paper for generating Main Ideas and Limitations. Here’s how we can approach this:**

1. **Title Extraction: Heading from the first chunk as the Title.**
2. **Main Ideas: For the Main Ideas, focus on the sections of the paper that are likely to contain summaries of the work, such as the Abstract and Introduction. These are typically the first sections of an academic paper that describe the objectives, methodology, and findings.**
3. **Limitations: For Limitations, focus on sections that often discuss the drawbacks or limitations of the study, such as the Conclusion or a specific Limitations section. If such a section doesn't exist explicitly, may look for sentences containing terms like "limitations," "future work," or "challenges."**

### **Use a Faster Model or Deployment**

* **Switch to Optimized Models**: If you're using a large model like llama3.2, consider alternatives optimized for speed (e.g., smaller versions or models fine-tuned for specific tasks).

### **Parallelize Operations**

* **Multiprocessing**: Use Python's multiprocessing or concurrent.futures to process multiple PDFs simultaneously.
* **Asynchronous Calls**: If your model supports asynchronous processing, use asyncio to process chunks concurrently.

### **Processor Utilization:**

**GPU**: From CPU to GPU

\*\*\*\*\*\*\*\*\*

**Model Name or Path**

* The identifier or file path for the specific LLaMA model (e.g., llama-7b, llama-13b).
* Example: "model\_name": "llama-7b"

**Token Limit**

* Maximum token length for input and output combined.
* Example: max\_tokens = 2048

**Temperature**

* Controls randomness in the output. Higher values make output more diverse.
* Example: temperature = 0.7

**Top-k Sampling**

* Limits the sampling to the top k most probable tokens.
* Example: top\_k = 50

**Top-p (Nucleus) Sampling**

* Retains the smallest set of tokens with a cumulative probability of p.
* Example: top\_p = 0.9

**Repetition Penalty**

* Penalizes repetitive token usage to encourage varied responses.
* Example: repetition\_penalty = 1.2

**Beam Search**

* Number of beams for beam search during decoding.
* Example: num\_beams = 5

Execution time

With CPU

For 1 rp = 3.30 min with chunk\_size=1500 and chunk\_overlap=200

For 4 rp = 4.30 min with chunk\_size=1500 and chunk\_overlap=200

For 4 rp = 3.35 min with chunk\_size=3000 and chunk\_overlap=200

With GPU

For 10 rp = 1.40 min with chunk\_size=3000 and chunk\_overlap=200

For 102 rp = 25.20 min with chunk\_size=3000 and chunk\_overlap=200

Need to apply multiprocessing