Python Full Stack

21 Jan

`python -m venv LPU` → To create a virtual enviroment

`.\LPU\bin\activate` → To acivate the enviroment

`deactivate` → to deactivate the enviroment

`rm -r LPU` - > to remove directory recursively, all the file & sub folders

`mv oldname newname` → To rename the Folder

'm' → stands for message
'venv' → virtual enviroment

22 Jan

Wider Range → Class
Derived → Objects

Animals → Class
Lion, Tiger → Objects

'.' → Location, LEKE AAO

`python` `filename.py` → To run the file

> 📖 Supervised Learning → Where we have both input and output. y = mx + c. Labeled Data.
> UnSupervised Learning → Where we don't have the output. Not Labeled data. The model try to find the pattern.
> ReEnforcement Learning → Which tries to learn from the experience. Reward based learning.

> 💡 Overfitting → Where the model gives alot of accuracy on the training data but fails in testing data.
> Underfitting → Which fails at the training data as well as testing data.
> TradeOff → Which performs well on training data as well as testing data.

sklearn → warehouse of ML.

```
              PV
            Yes  No

         _____
                  | TP | FN |
            Yes  |____|___|
      AV       | FP | TN |
            No   |____|___|
```

```
            PV
          No  Yes

       _____
       | TN | FP |
   No  |____|___|
AV     | FN | TP |
   Yes |____|___|
```

```
           AV
         Yes  No

      _____
      | TP | FP |
  Yes |____|___|
PV    | FN | TN |
  No  |____|___|
```

```
           AV
         No  Yes

      _____
      | TN | FN |
  No  |____|___|
PV    | FP | TP |
  Yes |____|___|
```

> 💡 Accuracy Score → TP+TN / TP+TN+FP+FN
> Precicison → TP / TP+FP   //Predicted mai kitna sahi predict kia.
> Recall → TP / TP+FN   //Actual mai kitna sahi predict kia.
> F1 Score → 2 * (Precision * Recall / Precision + Recall)

`from sklearn.metrics import classification_marix` → Classification Matrix

---

23 Jan

```
from sklearn.metrics import confusion_matrixav = ['dog','dog','dog','dog','not_dog','not_dog','not_dog','not_dog']
pv = ['dog','dog','dog','dog','not_dog','not_dog','not_dog','not_dog']
print(confusion_matrix(pv,av))
```
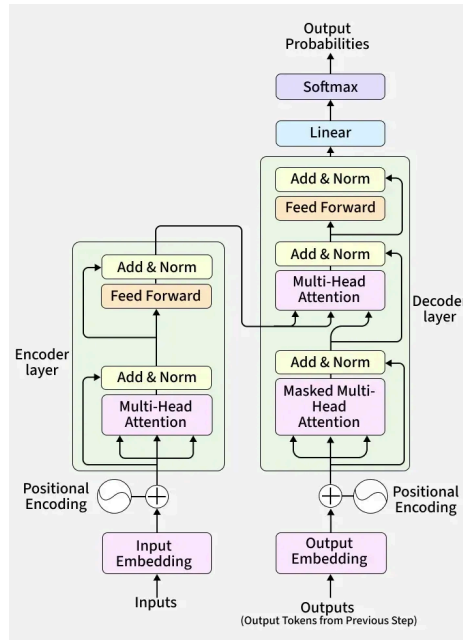
4	0
0	4

```
av = ['dog','dog','dog','dog','dog','not_dog','dog','not_dog','not_dog','not_dog']
pv = ['dog','dog','dog','dog','dog','dog','not_dog','not_dog','not_dog','not_dog']
print(confusion_matrix(pv,av))
```

5	1
1	3

"Transformers in Python" **primarily refers to the use of the powerful Hugging Face** `transformers` **library, which provides access to thousands of state-of-the-art pre-trained AI models for various tasks**

> 💡 **"Transformers in Python"** primarily refers to the use of the powerful Hugging Face `transformers` library, which provides access to thousands of state-of-the-art pre-trained AI models for various tasks

ANN	Tabular Data
CNN	Image Data
RNN	Sequential Data
ISTM	No Memory

DTPd4AF → Delhi mai ek TT ne pakda 4 log Afganisthan ke.

D	Defination	Linear Regression, etc.
Tp	Type of Problem	Supervised, UnSupervised, Clustering, etc.
Td	Type of Data	Labeled, UnLabeled.
4A	Aim, Approach, Algorithm, Application	
F	Feedback	

What is the type of problem for ReEnforcement Learning?
What is the type of data for ReEnforcement Learning?

Pipeline → Machine Learning Pipeline is a systematic workflow designed to automate the process of building, training, and deploying ML models.

Django

Django is software you can use to develop web applications quickly and efficiently

`pip install django`

`django-admin startproject project`

wsgi	Web Server Gateway Interface
asgi	(Asynchronous Server Gateway Interface)