

Python Full Stack

- ASCII Photo Converter - 30 Jan
- LMS using Django/Flask - 3 Feb
- Scrapping
 - OBS Studio

21 Jan

`python -m venv LPU` → To create a virtual environment
`.\LPU\bin\activate` → To activate the environment
`deactivate` → to deactivate the environment
`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 environment

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.
Reinforcement Learning → Which tries to learn from the experience. Reward based learning.



Overfitting → Where the model gives a lot 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 → $\frac{TP+TN}{TP+TN+FP+FN}$

Precision → $\frac{TP}{TP+FP}$ //Predicted mai kitna sahi predict kia.

Recall → $\frac{TP}{TP+FN}$ //Actual mai kitna sahi predict kia.

F1 Score → $2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$

`from sklearn.metrics import classification_matrix` → Classification Matrix

23 Jan

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

5	1	
1	3	

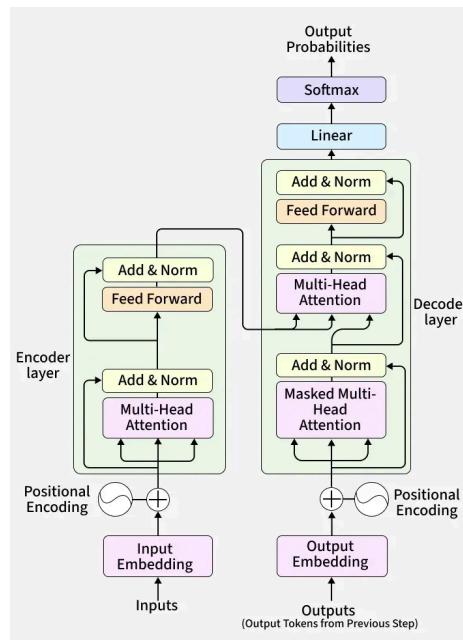
"Transformers in Python" primarily refers to the use of the powerful Hugging Face `transformers` library, which provides access to thousands of state-of-the-

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

Add Norm → Minimizing Errors



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

DTPd4AF → Delhi mai ek TT ne pakda 4 log Afghanistan 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 Reinforcement Learning?

What is the type of data for Reinforcement 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`

`python manage.py runserver`

<code>wsgi.py</code>	Web Server Gateway Interface. An entry-point for WSGI-compatible web servers to serve your project. See How to deploy with WSGI for more details.
<code>asgi.py</code>	(Asynchronous Server Gateway Interface). An entry-point for ASGI-compatible web servers to serve your project. See How to deploy with ASGI for more details.
<code>urls.py</code>	The URL declarations for this Django project; a “table of contents” of your Django-powered site. You can read more about URLs in URL dispatcher .
<code>settings.py</code>	Settings/configuration for this Django project. Django settings will tell you all about how settings work.
<code>__init__.py</code>	An empty file that tells Python that this directory should be considered a Python package. If you’re a Python beginner, read more about packages in the official Python docs.
<code>mysite/</code>	A directory that is the actual Python package for your project. Its name is the Python package name you’ll need to use to import anything inside it (e.g. <code>mysite.urls</code>).
<code>manage.py</code>	A command-line utility that lets you interact with this Django project in various ways. You can read all the details about <code>manage.py</code> in django-admin and manage.py .

24 Jan

Inheritance

- Single
- Multiple
- Multi Level
- Hierarchical
- Hybrid

2nd Project → LMS, using Django/Flask.

27 Jan

Transformer

Attention is the engine of Transformer. Combination of Encoder & Decoder.

One Encoder consists of Feed Forward and MHA

One Decoder consists of Feed Forward, MHA and mMHA.

First step is **Tokenization** :- Mainly word based Tokenization, means Character Based.

Second step is to give **Position** to each word, to identify uniquely.

Third Step is **Embedding**, converting string to number. And this serves as the input.

Word has feelings.

Query Key Value(QKV) :-

Activation Function :- To Normalize, Silu, Elu