**Put file on pipy**

* Go to directory
* Ensure package contains **\_\_init\_\_.py**
* Ensure folder containing package contains **licese.txt, readme.md, setup.py and setup.cfg**
* $ python setup.py sdist
* $pip install twine
* $twine upload dist/\*
* $pip install dsnd-probability

**AWS ML stack**

* **AI services**
  + Vision : rekognision
  + Speech : Polly [txt - audio], transcribe [audio - text]
  + Language : translate[English - persian], comprehend[understanding], textract[OCR]
  + Chatbot : lex
  + Forecasting : forecast, personalise
  + Development operations: codeguru [auto code review]
    - Intelligent recommendations
    - Find expensive line of codes, inefficiencies
  + Amazon Fraud detector
  + Enterprise search : kendra
  + Amazon connect : contact lense [alert supervisors, provide answer to customer reps as they are being asked, identify common call types and recurring issues]
* **ML services**
  + sageMaker
* **ML framework + infra**
  + Frameworks : tensorflow, mxnet,pytorch
  + Interface : keras, sklearn, gluon
  + Infrastructure : EC2, elastic kubernets, IoT greengrass

How deepcomposer works?

* nput melody captured on the AWS DeepComposer console
* Console makes a backend call to AWS DeepComposer APIs that triggers an execution Lambda.
* Book-keeping is recorded in Dynamo DB.
* The execution Lambda performs an inference query to SageMaker which hosts the model and the training inference container.
* The query is run on the Generative AI model.
* The model generates a composition.
* The generated composition is returned.
* The user can hear the composition in the console.
* The user can share the composition to SoundCloud.

<https://github.com/aws-samples/aws-deepcomposer-samples>