V-5 (DL)

Page No.
Date

Deep generative Models - (DGM)

- remarkable ability to learn underlying patterns in data & use knowledge to create entirely new data.
- understanding underlying data distribution.

Adv-

- Data Discovery
- Data Generation
- Variability with Data.

Disadv-

- Generative challenges
- Quality & Safety
- Training Complexity

Boltzmann Machine-

- unsupervised DL model; every node connected every other.
- connection are bidirectional, Boltzmanns are stochastic models.

Two nodes in BMS are-

- Visible Nodes - Nodes we can measure
- Hidden Nodes - Nodes that can't be or we can't measure
- Type of probalistic generative model → based on statistical physics.
- Type of ANN → consists of misconnected binary unit (neurons)
- designed → learn & model → data trained
- adjusting & strengths of connections
- def. how influential one neuron is on another.
- learning process of BM → ① upward pass
  - activates updated com & o/p of the neuron
  - gibbs sampling → determines activity of each connected neuron.
- ② downward pass
  - activities of neurons are used to reconstruct i/p data.
- gen samples → repeatedly updating activites of neuron. generate new data ↳ resemble data
- capture complex dependencies & interactions of neurons.

Page No.
Date

* appl^n → Dim red^n, collaborative filtering, collaborative filtering & feature learning.

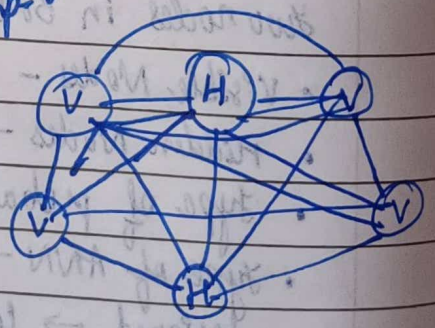
- BM training → computationally expensive.
- Math func → reps energy → particular config of binary state of neuron
- helps → prob of specific config occurring
  - each possible config → has an assoc. energy value.
  - energy of config → connection b/w neurons & their wts.
    energy is high → config less likely to occur.
    energy is low → " more " "
- energy func of BM → sum of weights cons b/w neurons.
  conn → same layer → b/w diff layers.
  weights reps → strengths of connections → reps show influential ar

$$E(x) = -\sum(w_{ij} * x_i * x_j) - \sum(b_i * x_i).$$

- $E(x)$ ↑ energy assoc with config
- w_{ij} ↑ wt of conn b/w them.
- x_i, x_j ↑ states of 2 conn neurons
- $b_i * x_i$ ↳ bias term.

- energy func in BM → prob of particular config → Boltz man distrib
  Prob of config → reg. energy / temp.

$$P(x) = \frac{1}{Z} * e^{(-E(x)/T)}$$



Page No.
Date

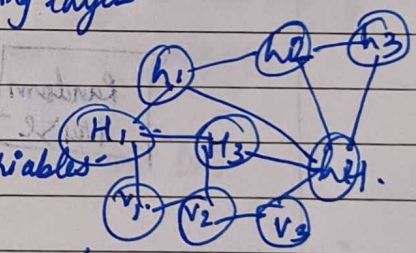
Deep Belief Networks -

- hybrid graphical model involving both directed & undirected connections with no intralayer connections like RBMs.
- has multiple hidden layers.
- first non-convolutional deep architecture.
- DBN = stack of RBMs + fine tuning
- DBN with only one hidden layer is an RBM.
- generative models with several layer of latent variables (hidden variable).
- connection b/w two layers are undirected while rest are directed.
- Working of DBNs -
  - Top two layers of DBNs.
  - multiple iterations → DBN generate samples
  - dependencies b/w top two hidden layers & exist now.

1) Gibbs sampling.

2) Ancestral sampling -
   - Gibbs, ancestral sampling is done on remaining layers.
   - generates samples from visible units.

3) Greedy pretraining -
   - used to determine values of hidden/latent variables.
   - done layer by layer
   - Main adv of DBN → Learn features in unsupervised manner
     → DBNs resistant to overfitting.

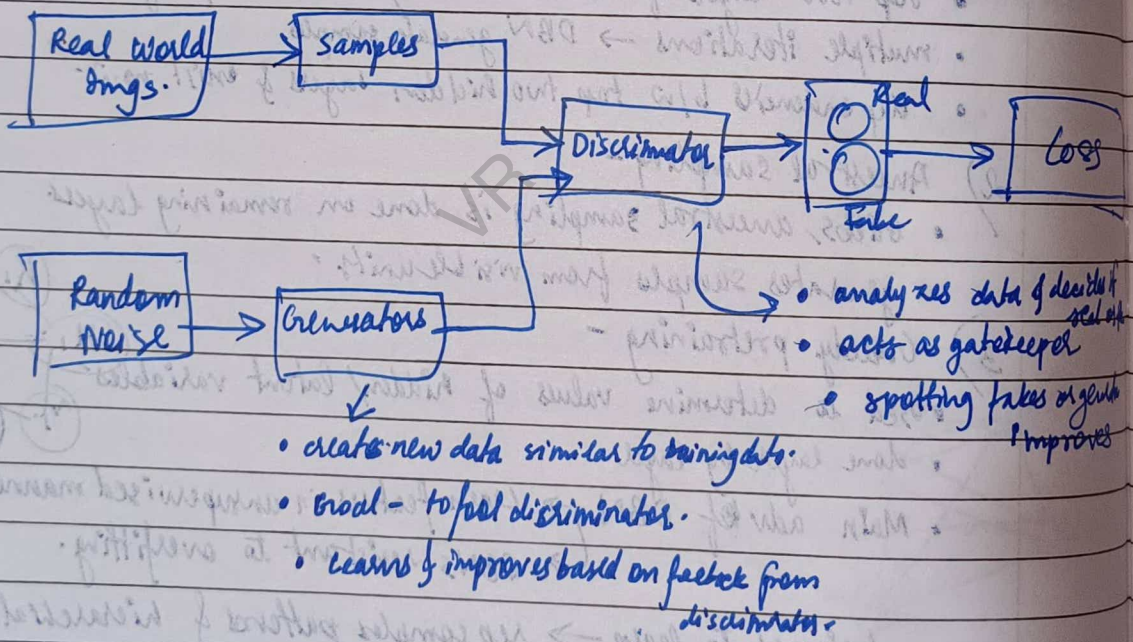


- designed to learn → rep complex patterns & hierarchical in data.
- multiple layers → rep complex patterns & hierarchial in data.
- each layer trained → learn most imp features / patterns.
- RBM → trained → DBN goes through fine-tuning.
- Adv → • learns meaningful features → raw data
  - eff in img recog & speech processing
  - capture hierarchical reps.
  - gen models → new data samples → reliable training data.
  - img recog, NLP, rec sys.
- disadv → • computationally expensive
  - RNN & CNN are pref due to eff & ↑ performance.

Page No.
Date

Generative Adversarial Network [GAN] →

- way to make generative models by having two neural networks compete with each other.
- exceptionaly potent neural networks → unsup learning.
- GAN → analyse → 2 neural n/w → compete with one another.
- GAN uses CNN + other DL techniques
- unsupervised learning technique → uses pattern in i/p data → find & learn them.
- applied to fresh instances.
- GAN training → framed task as sup learning utilising GANs
- discriminator → compares images with real world egs to classify



Page No.
Date

How GANs Work?

1. Basic approach — gen modelling that generates a new set of data based on training data → looks like training data.
2. 2 main blocks → NN → compete with each other (Gen & discrim).
3. can copy, capture & analyze variations dataset.
4. GAN term breakup -
   - Generative - explains how data gen statistically (how data gen is term of prob. model)
   - Adversarial - training of model is done in adv. setting.
   - Network - use deep NN for training purposes.
5. Generator → generates fake samples of data (img, audio, etc) & tries to fool discr.
6. Discriminator → distinguish b/w real & fake samples.
7. Both G & D are NN → run in competition with each other in training.
8. repeated several times both G & D get better at their jobs.
9. Gen model → ① captures distri of data ② trained in manner that max. prob. of discr.
10. Discrim. model → ② estimates of prob that sample recd is entered wrongly.
11. GANs formulates as minimax game.

ie G → max. D's reward ⇒ V(D,G) : ie maximise loss.
→ max V(D,G).

$$V(D,G) = E_{x \sim p_{data}(x)} \cdot [\log D(x)] + E_{z \sim p_z} [\log (1 - D(G(z)))]$$

Training Data → 2 sources → Real data → real picture
→ fake data → instances created by G.

Training the Disc-

1. connects 2 loss func.
2. ignores gloss & focus only on Dloss
3. Gloss used by G training.

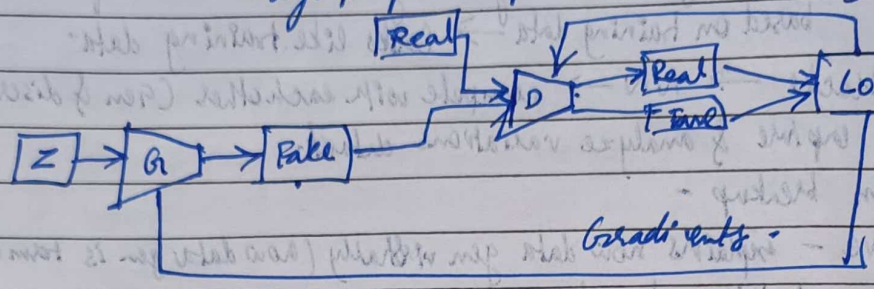
During D training → ① classifies ② penalises ③ updates.

Page No.
Date

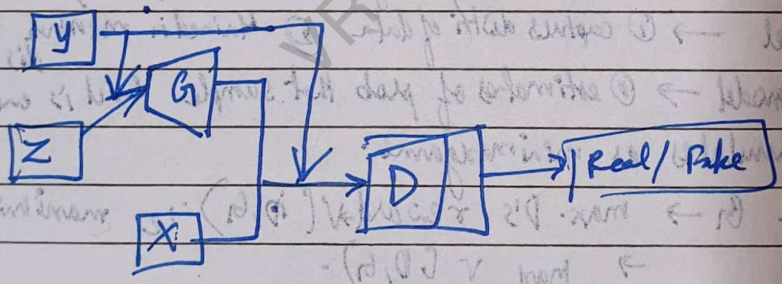
P

✱ TYES OF GAN -

① Vanilla GAN - • simplest GAN • G & D are simple •
• algo simple → optimizes mat eqn using stochastic gradient



② Conditional GAN (CGAN) -
- DL method in which cond° params are considered.
- CGAN → add° param(y) → added to G for gen corresponding data.
- Labels also put in D in order for D to help distng. fake & real data.



③ Deep Conditional GAN (DCGAN) -
- most popular + most successful implementation of GAN
- composed of convnets of multilayer preceptions.
- convnets implemented w/o max pooling infact replaced by convstrides
- layers not fully connected

④ Laplacian Pyramid GAN (LAPGAN).
- Lapplacian Pyramid is linear invertible img rep consisting of a set of band-pass imgs, spaced as octave apart + low freq residual uses multiple Gs & Ds -
- approach becoz it's produced high quality imgs.
- img is down-sampled at first at each layer of pyr.
- upscaled at each layer in backward path. When img acquires

Page No.
Date

(12) Super Resolution GAN (SR GAN) -

- DNN + adversial network in order to produce higher res imgs.
- useful in optimally scaling up native low-resoln imgs to enhance its details minimizes while doing so.

Applns of GAN -

- Img synthesis
- Img editing & style transfer
- super resolution.
- Data Augmentation
- Anomaly detection
- Text to Img Synthesis.