Lecture 5

Generative Models (Explicit Density(Approximate Density(Variational Autoencoders, Energy Based Models, Diffusion Models),Tractable Density(Autoregressive Models, Normalizing Flow Models)),Implicit Density (Generative Adversarial Models)) Autoregressive models condition predictions on previous values in the sequence, rather than on a latent random variable.

Query, Key, Value / Q, K, V, X is input embeddings, Q = X \* WQ K = X \* WK, V = X \* WV X is 1 x n, Attention scores = Q \* KT. Scaled Attention Scores = Q \* KT./ SQRT(DIM) Dot product measures similarity between query and key vectors a. If query and key are similar (point in same direction), score will be high b. If dissimilar (orthogonal), score will be low 2. Scaling by √dk improves numerical stability 3. Without scaling, dot products can grow large in magnitude for large dk, pushing softmax into regions with small gradients / Z = W · V, where Z has shape (n, dk) 4. Result is a matrix of shape (n, n) containing attention scores for each pair of elements

Graphical user interface, diagram

Description automatically generated

Diagram

Description automatically generated with low confidence

A picture containing text, clock

Description automatically generated

RNN – Recurrent Neural Network

Back Propagation Through Time – BPTT - Backward Pass: This is where BPTT diverges from standard backpropagation. The error is propagated back through each time step. Gradients are calculated not just with respect to the weights at each time step, but also with respect to the recurrent connections. The gradient with respect to each parameter must be summed across all places that the parameter occurs in the unrolled net. The total loss is often the sum or average of the losses at each relevant time step.

Create embeddings for multiple modalities (e.g., text and image) in the same vector space. Adaptive input representations assign vector embeddings of different sizes to words based on their frequency, rather than fixing the size for all words. / More frequent words get larger vector representations with more parameters, while less frequent words get smaller vectors.

Diagram

Description automatically generated with low confidence

LSTM – Long Short Term Memory – a variant of an RNN, have an Input Gate, a Forget Gate, and an Output Gate, they use Sigmoid

GRU – Gated Recurring Unit - A single gating unit simultaneously controls the forgetting factor and the decision to update the state unit.

PixelCNN – Use a mask with 1 for all previous pixels and 0 for all future pixels, generate pixels sequentially conditioned only on previously generated pixels, use skip connections

Lecture 6 – Multi Modal Large Language Models

ViT Vision Transformers - Patches with 1D, 2D, or relative embeddings. All performed about the same. 1D are computationally less expensive. Early layers have short to long mean distance for attention. Later layers have greater mean attention. Starts focusing on fine details and moves to overall

Quadrangle Attention (QA) – form multiple overlapping quadrangles, QA learns to transform default square windows into adaptive quadrangles using a projective transformation matrix, plain QFormer vs Hierarchical QFormer – Benefits: Adaptive Attention regions, Cross window information exchange, computational efficiency

Contrastive Language–Image Pre-training (CLIP), Web Image Text (WIT) – 400 Million Image/text, using 500,000 web queries that returned 20,000 image/text pairs each. CLIP's flexible text encoder enables "zero shot" transfer to downstream datasets without using any of their training examples. On average across 27 datasets, zero shot CLIP outperforms a ResNet50 trained with labeled examples from each dataset. CLIP is trained to maximize the cosine similarity of the Image / Text pairs that match and minimize the cosine similarity of the pairs that do not match.

Decoder Only Autoregressive Language and Image Synthesis (Dall E) – Input Prompt - Text Encoder – Text Embedding – Prior – Image Embedding – Decoder – Generated Image. The Decoder takes both the Input Prompt and Image Embedding / uses Text Encoder uses Clip / The autoregressive prior of DALL.E 2 is an encoder-decoder Transformer. It is trained to reproduce the CLIP image embedding given a CLIP text embedding.

BLIP2: Bootstrapping Language Image Pretraining with Frozen Image Encoders and Large Language Models, pre trained vision encoder and large language model. Trained Q-Former

STOPPED AT SLIDE 63

Lecture 7 – Diffusion Models

These are probabilistic models that define a nonlinear mapping from latent variables to the observed data where both quantities have the same dimension. all the learned parameters are in the decoder

Z1 = sqrt(1-beta1 \* X)+sqrt(beta1 \* epsilon1) / Zn = sqrt(1 – betan \* Zn-1) +sqrt(betan \* epsilonn) for all n from 2 to T. Betas are from [0, 1] and determine how quickly noise is added. Epsilons are noise drawn from a standard normal distribution. / the diffusion kernel is a handy mathematical shortcut that lets us sample the corrupted in one step, without having to do all the intermediate steps. To reverse we use Bayes Rule but this is intractable, when building the decoder we approximate the reverse process using a normal distribution

The formula turns out to be: *q(zt-1|zt , x)*= Gaussian over *zt-1 /*  Mean = weighted combination of zt and x / Variance = gets smaller as we diffuse backwards / To summarize: we learn a series of neural networks to reverse the blurring, approximated as Gaussians. / We then generate clear samples by walking back step-by-step. The integral is intractable so we lower bound the likelihood using Jensen’s Inequality, do it in batches / To fit the model, we maximize the ELBO with respect to the parameters φ1...T . / We recast this as a minimization by multiplying with minus one and approximating the expectations / In summary, reparameterization changes the model to predict the noise rather than the diffused image, leading to a simpler and more effective loss function.

Implementation – cosine diffusion schedule / UNet consists of two halves. Down sampling – images are compressed and channels expanded – up sampling where images are expanded spatially and channels reduced. Skip connections between blocks. Predicts the noise added to the image and has the same shape.

Graphical user interface

Description automatically generated with low confidence

Lecture 8 – Generative Adversarial Networks (GANS)

Disregarding additive and multiplicative constants, this is the Jensen-Shannon divergence between the synthesized distribution Pr(x∗) and the true distribution Pr(x) and where DKL[•||•] is the Kullback-Leibler divergence / The first KL term penalizes regions with samples x∗ but no real examples x; it enforces quality. / The second KL term penalizes regions with real examples but no samples. It enforces coverage. / Mode Collapse / GAN loss function encourages quality but lacks explicit coverage term, potentially contributing to mode collapse. Quality penalty (pr(x\*) > pr(x)) Coverage penalty (pr(x) > pr(x\*)). / Gan training / The discriminator parameters *φ* are modified to assign high probability to the real examples and low probability to the generated samples. / The generator parameters *θ* are modified to “fool” the discriminator into assigning the generated samples a high probability. / Generator goes from low dimensions and high channels to high dimensions and low channels. Discriminator goes from high dimensions and low channels to low dimensions and high channels. Mode dropping is leaving off bits such as beards. Mode collapse is an extreme version when only one or a few points are included. If generated and real images are easy to distinguish, nothing changes WGAN / work, earth movers, distance to move / Lipschitz constrains how fast output of a function can change based on input / In the context of WGANs (Wasserstein GANs), the critic function is required to be 1-Lipschitz, meaning that the Lipschitz constant K should be less than or equal to 1. / Weight Clipping or Gradient Penalty, alternative add a penalty term, estimate for interpolated points between the real and fake image / CGAN / includes a condition with the image / describes some aspect of the image / ACGAN / Probability is real and probability is in class / InfoGAN / Probability is real and estimate of c / CycleGAN / StyleGAN

Lecture 9 – Normalizing Flow Models

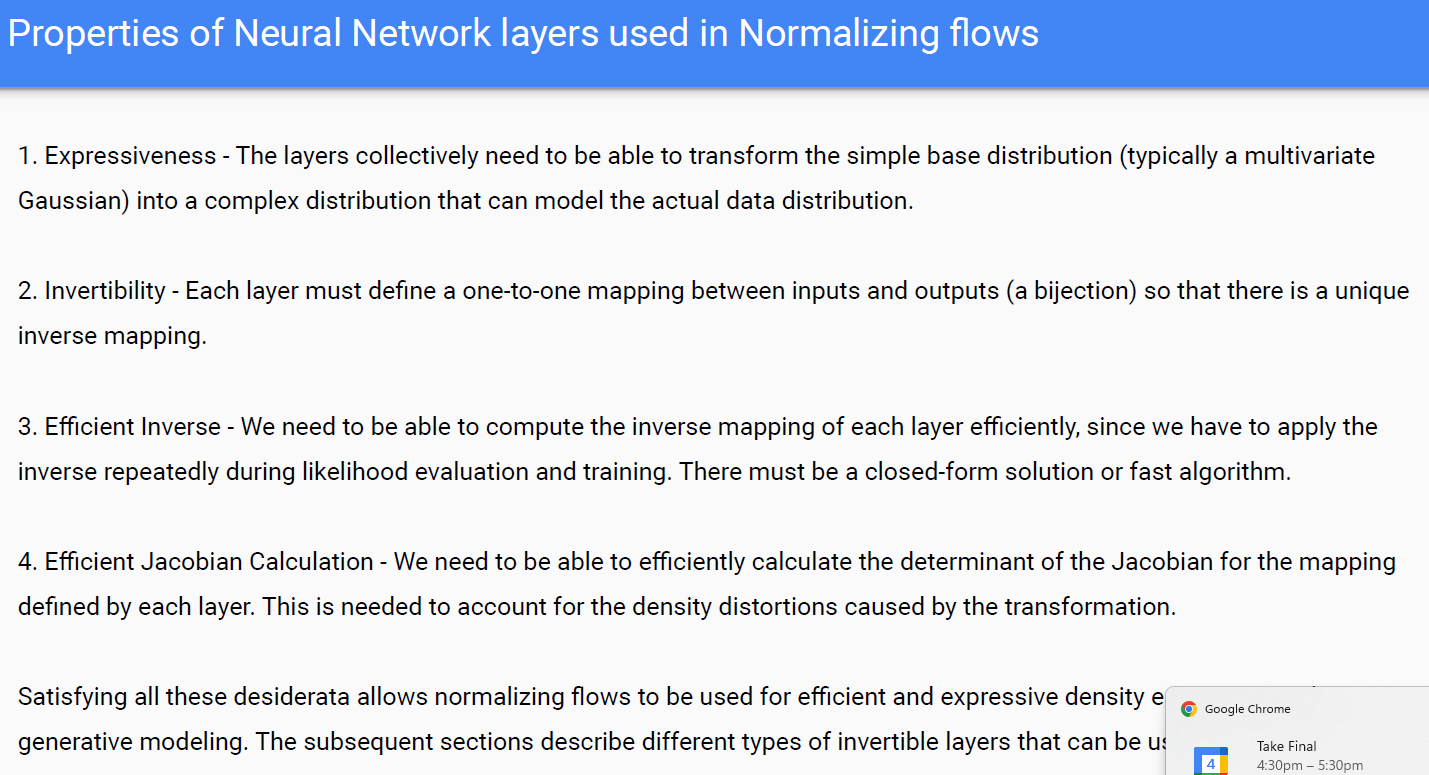
Each layer must be invertible / like autoregressive models can tractably model the data generating distribution / lime VAEs attempt to map to a simpler distribution such as a Gasian / One to One (at most one x s.t. f(x) = y) and Onto (at least one x s.t. f(x) = y) / inverse exists. Preserve information, can compose function and Determinant of Jacobian exists / Affine X 🡪 Ax + b, Elementwise nonlinearities X 🡪 f(x) where f is monotonically increasing, coupling layers x = (x1, x2) 🡪 (x1, x2 \* exp(s(x1)) + t(x1)) / Base Density in Z 🡪 x = f(Z, Theta) 🡪 Model Density in X. Not that Z = f-1(X, Theta) because f is invertible. Base density is usually the normal standard distribution / training minimize the log likelihood / partial of f with respect to z is the determinant of the Jacobian. / multiple layers / So in summary, we compose invertible neural network layers to transform a simple base distribution to a complex one, with the Jacobian accounting for the distortions.

The Jacobian determinant is a single number that summarizes the overall effect of the distortion / It tells you how much the function expands or shrinks the volume (or area in 2D) around the point / Greater than 1: expanding the space, making the volume larger / Between 0 and 1: contracting the space, making the volume smaller / Exactly 1: preserving the volume, keeping it the same / 0: collapsing the space, mapping it to a lower-dimensional space / Negative: flipping the orientation of the space, turning it inside out

Change of variables / Find a g that maps the probability to a unit space. It is invertible. / it is the absolute value of the Jacobian determinant of the transformation. / this scaling factor gets s back to P(x) = 1 for all x.

Text

Description automatically generated



Timeline

Description automatically generated with low confidence

Text, letter

Description automatically generated

S is scale and t is translation / det(J) = exp(sum over j(s(x1:d)j]

Stacking coupling layers, alternate masking pattern, updates previously unchanged part

1:d rows go through unchanged, rows d througD are changed