Deep Generative Models

Chapter 7: Multimodal Models and Conditional Generation

Ali Bereyhi

ali.bereyhi@utoronto.ca

Department of Electrical and Computer Engineering

University of Toronto

Summer 2025

Multimodal Model

+ *We studied a lot, but did not hear too much about multimodal models! Where do we learn about them?*

– Well! You already know whatever you need to know about them

Multimodal Model

Multimodal models combine inputs from different modalities, e.g., visual and textual, to make outputs that are more informed than unimodal models

In practice, we always think about multimodal generative models

Multimodal Generative Model

A multimodal generative model learns learns to sample data of one modality upon observing samples of data in another modality

Multimodal Model: *Example*

There are various forms of multimodal generation: *text-to-image conversion*

Text Prompt: an armchair in the shape of an avocado. . . .

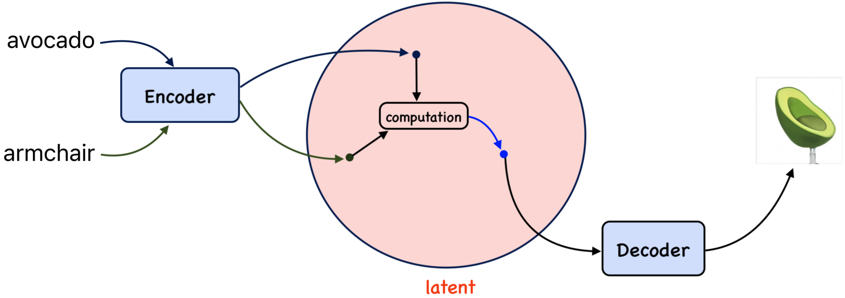
AI Generated images

or video question answering

Modality: *Latent Space*

+ *How do these models work in practice?*
– Basically, by connecting modalities in the latent space

Different modalities can be related in the same latent space



Statistical Viewpoint

+ *But, how can we integrate multimodality into the statistical framework?*
– It's all about conditioning

Say we see a sample of some modality: let's call it u

↳ *For instance, u could be a text describing an image we are looking for*

To change the modality, we should sample x of different nature

↳ *In our example, x is the image described by u*

The sample x should depend on u, so we know that

It's not simply coming from distribution $P(x)$, but it's coming from

$$P(x|u)$$

which describe the density of samples related to u

Statistical Viewpoint: *Conditional Generation*

Conditional Generative Model

A conditional generative model is a model that upon being prompted by u samples from the conditional data distribution $P(x|u)$

Example: *Say we are converting text to image*

$$u = \texttt{give me an image of a dog with glasses}$$

$P(x|u)$ *then describes the distribution of all dogs with glasses*

large $P(x|u)$

$P(x|u) \approx 0$

Learning Conditional Distribution

+ *Can we use our approaches to learn conditional distributions?*
– Absolutely!

We are now aiming to learn $P(x|u)$: we consider a computational model

$$u \longrightarrow \boxed{f_{\mathbf{w}}(u)} \longrightarrow P_{\mathbf{w}}(x|u)$$

This model describes the conditional distribution for an input prompt u

↳ *This can be any of models we had*

↳ *We now also include the conditioning in our computation*

Learning Conditional Distribution

+ *What about data samples?*
– We collect samples from the conditional distribution

$u =$ `give me an image of a dog with glasses`

+ *Do we have many samples for any u?*
– We would be fine with as small as one!

Learning Joint Distribution

+ *With only one pair (u, x) how do we know that the model is not learning thejoint distribution?*
– Let it learn, it's still OK!

Bayes' rule indicates that

$$P(x|u) = \frac{P(x, u)}{P(u)}$$

For a fixed u: $P(x|u)$ is a normalized version of $P(x, u)$

↳ *$P(x^1, u)$ and $P(x^2, u)$ are relatively the same as $P(x^1|u)$ and $P(x^2|u)$*

↳ *Learning $P(x, u)$ still do the job for us!*

Naive Approach: *Label Sampling*

+ *How can we learn the conditional distribution in practice then?*
– Most naively, we can train our model on down-sampled data

Say we want to train sample on MNIST with u being the digit label

- *We can make a subset of MNIST for each digit u*

$$\mathbb{D}_u = \{x \in \mathbb{D} : \textit{label}\,(x) = u\}$$

- *Training any generative model on $\mathbb{D}_u$ learns $P\,(x|u)$*

+ *You mean train 10 separate models?!*
– *Theoretically yes, but practically not really!*

Practical Implementation: *Label Embedding*

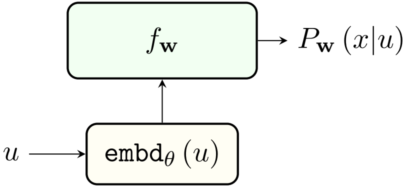
In practice, we can embed the label into our model

Let $\texttt{embd}_\theta(\cdot)$ *be an embedding model returning* $\texttt{embd}_\theta(u)$ *for a given* u

↳ *We did this in LMs to embed a token*

↳ *We did this in DPMs to embed the time*

We can modify our generative model $P_{\mathbf{w}}$ *as*



and use both MNIST images and their labels to train this model

Generic Solution: *Condition Embedding*

In practice, we always condition a generative model by embedding

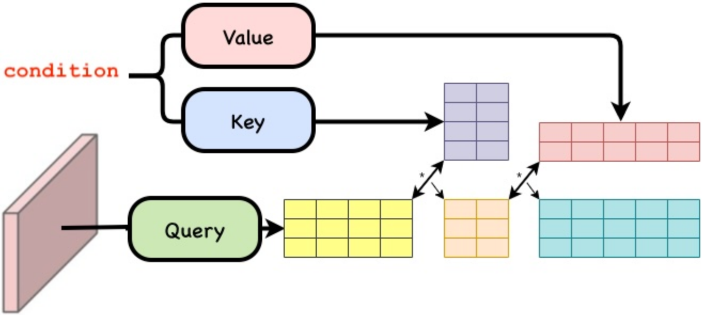
Condition Embedding

In condition embedding, we compute a rich representation of condition u via a computational sub-model, e.g., an embedding layer or an encoder, and include it in the generation process in rather dense way

+ *What do you mean by dense way?!*
– *Simply speaking, the condition embedding should show up every layer*
+ *How can we do it?!*
– *We can simply have it in every layer; there are other approaches as well*

Practical Implementation: *Cross Attention*

One popular approach is to use cross attention



Practical Implementation: *FiLM*

Another approach is to use feature-wise linear modulation (FiLM): say we have a particular layer $L_{\mathbf{w}}$ in our basic model

it computes features $y_\ell = L_{\mathbf{w}}(y_{\ell-1})$ from outputs of previous layer $y_{\ell-1}$

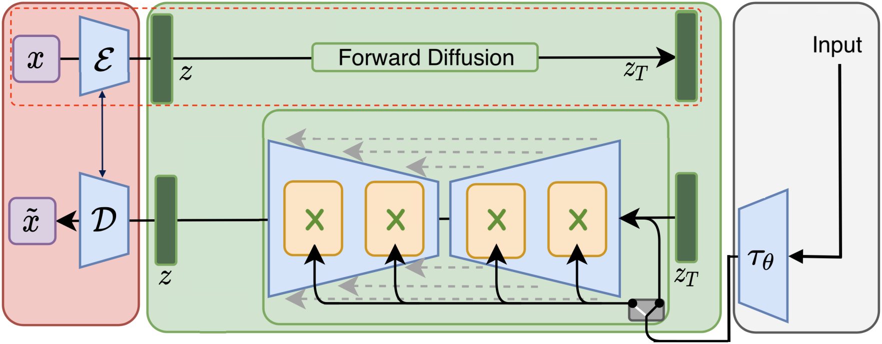
FiLM applies conditioning as follows

1. *it computes embeddings $\{a, b\} = \texttt{embd}_\theta(u)$*
2. *it conditions all layers in the model as*

$$\mathrm{FiLM}\{L_{\mathbf{w}} | u\} = a \circ L_{\mathbf{w}} + b$$

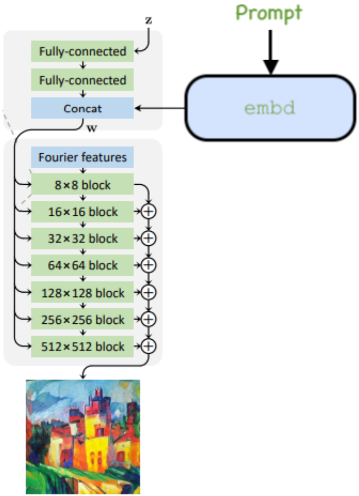
Stable Diffusion: *Cross Attention*

A well-known example of conditioning by cross attention is Stable Diffusion



Conditional StyleGAN: *FiLM*

Conditional StyleGAN was one of famous models using FiLM



Final Notes

All we need for multimodality is conditioning

- *Conditioning is implemented by embedding the prompt*
- *To keep the conditioning strong, the embedding should integrate densely*

We can make all generative models: you could take a look as

- *Conditional AR Models*
- *Conditional EBMs*
- *Conditional GANs*
- *Conditional VAEs*
- *Conditional Diffusions*

You Know a Lot . . .

It was a great pleasure to learn generative models together in this semester!

! Important

Any knowledge comes hand-in-hand with responsibility!

You know a lot about Generative AI: *this means that*

- *You should be self-confident to use if for problem solving*
- *You are responsible to inform people about what it can do*
  - ↳ *An LLM can learn the writing style of an author from its books*
  - ↳ *The author has the right to know this before sharing their books!*
- *You should educate yourself about privacy and ethical aspects*
  - ↳ *We are still on the beginning and many things have not been regulated!*