Study Material - Youtube

Document Information

- **Generated:** 2025-08-26 07:11:50
- **Source:** https://www.youtube.com/watch?v=kJCgO7rwo3Y
- **Platform:** Youtube
- **Word Count:** 2,354 words
- **Estimated Reading Time:** ~11 minutes
- **Number of Chapters:** 5
- **Transcript Available:** Yes (analyzed from video content)

Table of Contents

Video Overview

This lecture, titled "Guided Diffusion Models," is part of the "Mathematical Foundations of Generative AI" series. It builds upon the concepts of Denoising Diffusion Probabilistic Models (DDPMs) and score functions to introduce methods for **conditional generation**. The instructor, Prof. Prathosh A P, explains how to modify a standard DDPM, which generates samples from a marginal data distribution, to generate samples conditioned on specific attributes like a class label or a text description. Two primary techniques are discussed in detail: **Classifier Guidance** and the more advanced **Classifier-Free Guidance**, which is the state-of-the-art method used in many modern generative models.

Learning Objectives

Upon completing this lecture, students will be able to: - Understand the fundamental difference between unconditional and conditional generation in the context of diffusion models. - Explain the goal of guided diffusion: to sample from a conditional probability distribution $p(x_0|y)$. - Derive the core mathematical principle behind **Classifier Guidance**, which decomposes the conditional score into an unconditional score and a classifier gradient. - Understand the implementation and limitations of Classifier Guidance, particularly the need for a separate, pre-trained classifier. - Grasp the concept and motivation behind **Classifier-Free Guidance** as a more elegant and effective solution. - Detail the training and inference procedures for Classifier-Free Guidance, including the use of a null conditioning token and the guidance scale hyperparameter λ. - Appreciate the mathematical connection between score functions, Bayes' law, and the guidance mechanisms in diffusion models.

Prerequisites

To fully understand the concepts presented in this video, students should have a solid grasp of: - **Denoising Diffusion Probabilistic Models (DDPMs):** The forward (noising) and reverse (denoising) processes. - **Score Functions:** The definition of a score function as the gradient of the log probability density, $\nabla_x \log p(x)$. - **Score Matching:** The concept of training a model to estimate the score function. - **DDPM as Score Predictors:** The crucial insight that a DDPM trained to predict noise is implicitly predicting the score function of the noisy data distribution. - **Probability Theory:** Concepts like conditional probability, marginal probability, and Bayes' Law. - **Calculus:** Understanding of gradients (∇) and basic calculus rules.

Key Concepts Covered

- Conditional Generation
- Guided Diffusion
- Classifier Guidance
- Classifier-Free Guidance
- Conditional Score
- Unconditional Score
- Classifier Gradient
- Null Conditioning Token
- Guidance Scale (λ)

Guided Diffusion Models for Conditional Generation

The lecture begins by establishing the context for guided diffusion. While standard DDPMs are powerful for unconditional generation (sampling from the overall data distribution $p(x_0)$), most practical applications require **conditional generation**. This means we want to guide the generation process based on some input condition, `y`.

Examples of Conditional Generation (01:05): - **Text-to-Image:** Given a text prompt `y` (e.g., "a photo of an astronaut riding a horse in space"), generate a corresponding image `x_0`. - **Class-Conditional Generation:** Given a class label `y` (e.g., "cat"), generate an image `x_0` that belongs to that class.

The fundamental goal is to modify the diffusion model to sample from the **conditional distribution** $p(x_0|y)$ instead of the marginal distribution $p(x_0)$.

Data for Conditional Generation (01:38)

To train a conditional model, the dataset must consist of pairs or tuples of the form $(\mathbf{x_0}, \mathbf{y})$, where: - $\mathbf{x_0}$ is the data sample (e.g., an image). - $\mathbf{y}$ is the corresponding conditioning variable.

The conditioning variable `y` can take various forms, but it is typically represented as a vector: - **Class Label:** For a dataset with `C` classes, `y` can be a one-hot encoded vector of length `C`. - **Text Embedding:** For text-to-image models, `y` is a high-dimensional vector embedding of the input text, often generated by a separate pre-trained language model like a Transformer.

The Core Problem: Estimating the Conditional Score (04:12)

Recall that the reverse process in a DDPM is guided by the score function. - An **unconditional DDPM** learns to estimate the score of the marginal distribution: $\nabla_{x_t} \log p(x_t)$. - A **conditional DDPM** must learn to estimate the score of the conditional distribution: $\nabla_{x_t} \log p(x_t|y)$.

The central question is how to modify a DDPM to compute or approximate this conditional score.

Method 1: Classifier Guidance (05:30)

Classifier Guidance is the first approach to solving this problem. It leverages a separate, pre-trained classifier to inject the conditional information into the denoising process.

Mathematical Derivation

The derivation elegantly connects the conditional score to the unconditional score using Bayes' Law.

1. **Start with the target conditional score:** We want to find an expression for $\nabla_{x_t} \log p(x_t|y)$.

2. **Apply Bayes' Law:** The conditional probability $p(x_t|y)$ can be expressed as:

$$p(x_t|y) = \frac{p(y|x_t)p(x_t)}{p(y)}$$

   - **Intuition:** This rule relates the probability of `x_t` given `y` to the probability of `y` given `x_t`.

3. **Take the Logarithm:** Applying the logarithm to both sides separates the terms:

$$\log p(x_t|y) = \log p(x_t) + \log p(y|x_t) - \log p(y)$$

4. **Take the Gradient:** We differentiate with respect to x_t. The term $\log p(y)$ is constant with respect to x_t, so its gradient is zero.

$$\nabla_{x_t} \log p(x_t|y) = \nabla_{x_t} \log p(x_t) + \nabla_{x_t} \log p(y|x_t)$$

Conceptual Breakdown of the Result

This equation is the cornerstone of classifier guidance. It shows that the desired **conditional score** is the sum of two components:

1. $\nabla_{x_t} \log p(x_t)$: This is the **unconditional score**. It is exactly what a standard DDPM is trained to predict. It guides the sample to look like a plausible image from the overall dataset.
2. $\nabla_{x_t} \log p(y|x_t)$: This is the **classifier gradient**. It is the gradient of the log-likelihood of a classifier that predicts the label `y` from the noisy input `x_t`. This term "pushes" the sample x_t in a direction that makes it more recognizable as class `y` to the classifier.

   Key Insight: To generate a conditional sample, we can start with the guidance from an unconditional model and add a "nudge" from a classifier that knows about the condition.

Implementation and Drawbacks

The implementation of Classifier Guidance can be visualized as follows:

```
graph TD
    subgraph Denoising Step at time t
        direction LR
        U["U-Net (DDPM)<br/>Predicts Unconditional Score<br>  log p(x_t)"]
        C["Pre-trained Classifier<br/>Predicts p(y|x_t)"]

        xt["Noisy Sample x_t"] --> U
        xt --> C

        U -->|Unconditional Score| A["+"]
        C -->|Compute Gradient<br>  log p(y|x_t)| A

        A -->|Conditional Score<br>  log p(x_t|y)| D["Use score to compute<br> _ (x_t, y)"]
        D --> xt_minus_1["Denoised Sample x_{t-1}"]
    end
```

This flowchart illustrates that at each denoising step, the unconditional score from the DDPM and the gradient from a separate classifier are combined to produce the conditional score, which guides the generation of the next, less noisy sample.

Drawbacks (22:51): 1. **Requires a Separate Classifier:** You must train and maintain an additional model alongside the diffusion model. 2. **Classifier Must Handle Noise:** The classifier needs to be robustly trained on noisy inputs x_t for all possible noise levels `t`. This is a challenging training task in itself. 3. **Performance Dependency:** The quality of the final generated image is highly dependent on the quality of the external classifier. A poor classifier will provide poor guidance.

Method 2: Classifier-Free Guidance (24:16)

Classifier-Free Guidance is a more modern and effective technique that achieves conditional generation **without needing an external classifier**. It cleverly uses a single neural network for both conditional and unconditional score prediction.

The Core Idea and Training

The key innovation is in the training procedure of the U-Net.

1. **Joint Training:** The U-Net is trained on the paired dataset `(x_0, y)`.
2. **Conditional Dropout:** During training, the conditioning vector `y` is randomly replaced with a special **null token** with a certain probability (e.g., 10-20% of the time).
3. **Dual-Purpose Model:** This process trains a single model, parameterized by θ, to perform two tasks:
   - When given a valid `y`, it learns to predict the **conditional score**: $s_\theta(x_t, y) \approx \nabla_{x_t} \log p(x_t|y)$.
   - When given the null token , it learns to predict the **unconditional score**: $s_\theta(x_t, \emptyset) \approx \nabla_{x_t} \log p(x_t)$.

Mathematical Formulation and Inference

During inference, the model performs two forward passes to get both score estimates. These are then combined to form an extrapolated score that guides the generation.

The final guided score, $\tilde{s}_\theta(x_t, y)$, is a linear combination of the conditional and unconditional predictions:

$$\tilde{s}_\theta(x_t, y) = s_\theta(x_t, \emptyset) + \lambda \left(s_\theta(x_t, y) - s_\theta(x_t, \emptyset) \right)$$

- **Intuition:** This formula starts with the unconditional score (the general direction of a plausible sample) and adds a vector pointing from the unconditional to the conditional prediction. The hyperparameter λ scales this guiding vector.
- **Guidance Scale λ:**
  - If $\lambda = 0$, $\tilde{s}_\theta = s_\theta(x_t, \emptyset)$, resulting in **unconditional generation**.
  - If $\lambda = 1$, $\tilde{s}_\theta = s_\theta(x_t, y)$, resulting in standard **conditional generation**.
  - If $\lambda > 1$, the guidance is **amplified** or **extrapolated**. This pushes the generation more strongly towards the condition `y`, often improving sample quality and adherence to the prompt, at the risk of reducing diversity.

The formula can be rearranged into a weighted average:

$$\tilde{s}_\theta(x_t, y) = (1 - \lambda) s_\theta(x_t, \emptyset) + \lambda s_\theta(x_t, y)$$

Implementation

The inference process for Classifier-Free Guidance is as follows:

```
flowchart TD
    subgraph Denoising Step at time t
        direction LR
        xt["Noisy Sample x_t"]
        y["Condition y"]
        null["Null Token  "]

        subgraph UNet
            direction LR
            xt_y_input["x_t, t, y"] --> Model["U-Net ( )"] --> s_cond["Conditional Score<br>s_ (x_t, y)"]
```

```
        xt_null_input["x_t, t,  "] --> Model --> s_uncond["Unconditional Score<br>s_ (x_t,  )"]
    end

    s_cond --> C{Combine Scores}
    s_uncond --> C

    C --> |"s_uncond +  (s_cond - s_uncond)"| D["Final Guided Score<br>s̃_ (x_t, y)"]
    D --> E["Use score to compute<br> _ (x_t, y)"]
    E --> xt_minus_1["Denoised Sample x_{t-1}"]
end
```

This flowchart shows that for each denoising step, the same U-Net model is run twice: once with the condition y *and once with a null token . The resulting conditional and unconditional scores are combined with a guidance scale to produce the final score that directs the next step of the generation.*

Key Mathematical Concepts

This lecture revolves around a few central mathematical ideas.

1. **DDPM Score Equivalence (0:11):** The true score is equivalent to negatively scaled noise.

   $$\nabla_{x_t} \log p(x_t) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}}\epsilon_t$$

   This establishes the link between noise prediction in DDPMs and score matching.

2. **Classifier Guidance Formula (9:11):** The conditional score is the sum of the unconditional score and the classifier gradient.

   $$\underbrace{\nabla_{x_t} \log p(x_t|y)}_{\text{Conditional Score}} = \underbrace{\nabla_{x_t} \log p(x_t)}_{\text{Unconditional Score}} + \underbrace{\nabla_{x_t} \log p(y|x_t)}_{\text{Classifier Gradient}}$$

3. **Classifier-Free Guidance Formula (26:21):** The final guided score is an extrapolation between the model's conditional and unconditional predictions.

   $$\tilde{s}_\theta(x_t, y) = (1-\lambda)s_\theta(x_t, \emptyset) + \lambda s_\theta(x_t, y)$$

   where $s_\theta(x_t, y)$ is the model's prediction for the conditional score and $s_\theta(x_t, \emptyset)$ is its prediction for the unconditional score.

Visual Elements from the Video

The instructor uses several handwritten diagrams to illustrate the concepts.

- **Classifier Guidance Architecture (13:00 - 19:18):** A diagram showing two separate models. A U-Net predicts the unconditional score (or equivalent noise), and a separate pre-trained classifier/regressor provides the gradient $\nabla_{x_t} \log p(y|x_t)$. These two outputs are combined to guide the diffusion process.
- **Classifier-Free Guidance Architecture (29:07 - 31:45):** A diagram showing a single U-Net model being used twice.
  1. **Conditional Pass:** The U-Net takes `x_t`, `t`, and the condition `y` as input to produce the conditional score.

2. **Unconditional Pass:** The same U-Net takes `x_t`, `t`, and a null token as input to produce the unconditional score. The two resulting scores are then linearly combined to create the final guidance for the denoising step.

Self-Assessment for This Video

1. **Question:** What is the primary goal of guided diffusion, and how does it differ from unconditional diffusion?
   - **Answer:** The goal of guided diffusion is to generate samples from a conditional distribution $p(x_0|y)$, meaning the output is controlled by a condition `y`. Unconditional diffusion generates samples from the marginal distribution $p(x_0)$ without any specific control.
2. **Question:** Explain the two main components of the conditional score in the Classifier Guidance framework. What does each component represent intuitively?
   - **Answer:** The conditional score $\nabla_{x_t} \log p(x_t|y)$ is composed of:
     1. The **unconditional score** $\nabla_{x_t} \log p(x_t)$, which guides the sample to be a plausible data point from the entire dataset.
     2. The **classifier gradient** $\nabla_{x_t} \log p(y|x_t)$, which pushes the sample to be more recognizable as belonging to the condition `y`.
3. **Question:** What is the main drawback of the Classifier Guidance method?
   - **Answer:** Its main drawback is the need to train a separate classifier on noisy data for all time steps, which is difficult and makes the overall system's performance dependent on this external model.
4. **Question:** How does Classifier-Free Guidance eliminate the need for a separate classifier? Describe the training process.
   - **Answer:** It uses a single U-Net model trained to predict both conditional and unconditional scores. This is achieved by randomly replacing the conditioning vector `y` with a null token during training, forcing the model to learn both tasks.
5. **Question:** In Classifier-Free Guidance, what is the role of the guidance scale λ? What happens when $\lambda = 0$, $\lambda = 1$, and $\lambda > 1$?
   - **Answer:** The guidance scale λ controls the strength of the conditioning.
     - $\lambda = 0$: The model performs unconditional generation.
     - $\lambda = 1$: The model performs standard conditional generation.
     - $\lambda > 1$: The model's guidance towards the condition is amplified, often improving sample quality at the cost of some diversity.

Key Takeaways from This Video

- DDPMs can be extended from unconditional to conditional generators by guiding the reverse diffusion process.
- This guidance is achieved by modifying the score function to be conditional on an input `y`.
- **Classifier Guidance** accomplishes this by adding the gradient from a separate, pre-trained classifier to the unconditional score.
- **Classifier-Free Guidance** is a more efficient and powerful method that trains a single model to predict both conditional and unconditional scores, which are then combined during inference.
- The state-of-the-art approach (Classifier-Free Guidance) is used in many large-scale commercial models for tasks like text-to-image generation.