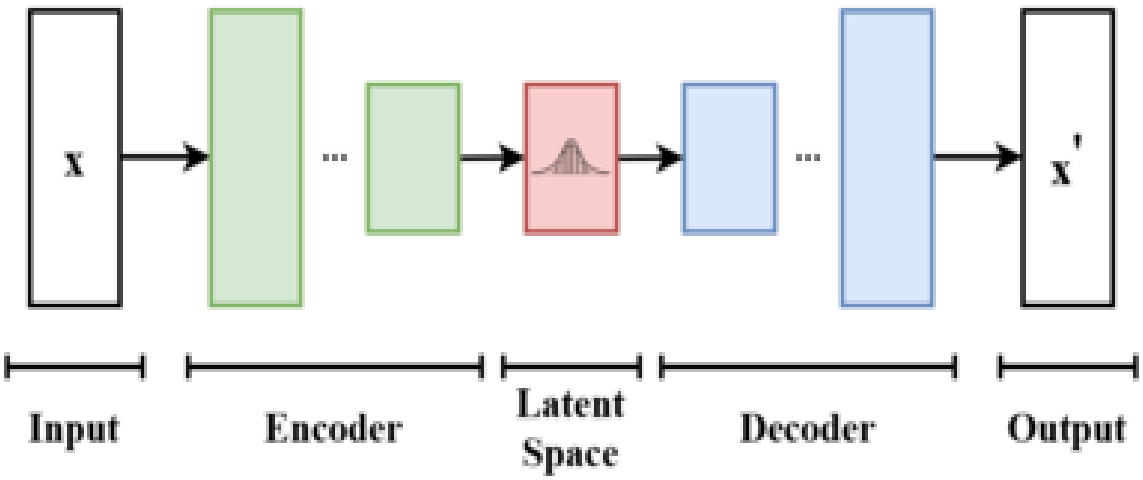
Introduction:

Stable diffusion is a technique employed in text-to-image generation where an initial random noise image undergoes iterative refinement guided by learned gradients and textual prompts. This process aims to generate high-quality images that align with the given textual description. Stability ensures smooth and consistent changes across iterations, leading the image towards convergence with the provided description. Quality control mechanisms are employed to monitor fidelity and coherence, ensuring alignment with predefined criteria throughout the refinement process.

Variational Auto-Encoder:

In text-to-image generation, we prefer using variational autoencoders (VAEs) over regular autoencoders due to their ability to capture the underlying distribution of images in a more structured and probabilistic manner. While a regular autoencoder compresses the input image into a fixed latent representation, a VAE learns a distribution of possible latent representations, allowing for greater diversity and flexibility in the generated images.

This means that with VAEs, we can generate a wider range of images, each with subtle variations, leading to more realistic and diverse outputs. Additionally, VAEs offer the advantage of latent space interpolation, allowing for smooth transitions between different image features, which can be useful for tasks like image editing or manipulation. Overall, VAEs provide a more sophisticated and nuanced approach to image generation compared to regular autoencoders, making them a preferred choice for text-to-image generation tasks.



The above is an example of the variational autoencoder architecture. The model receives x as input. The encoder compresses it into the latent space. The decoder receives as input the information sampled from the latent space and produces x' as similar as possible to

x. tt By applying VAE we can make the U-Net process faster compared with the autoencoder.

Stable Diffusion Architecture for Text-to-Image:

Stable Diffusion is an approach to generate images from textual descriptions. Here's adetailed breakdown of the process, step by step:

1. **Input:**

- **Noise:** A random noise image selected from the training data. This noise acts as a starting point for the image creation process.
- **Text Prompt:** We Provide a textual description of the desired image. This prompt serves as a guide for the model to shape the random noise towards our vision.

2. **Encoding:**

- **Encoder:** This component takes the random noise image and transforms it into alatent representation. This representation is a compressed version of the image containing essential information in a mathematical format.
- **CLIP Encoder:** Meanwhile, the text prompt enters the CLIP encoder. This encoder processes the text and extracts key features, converting it into "prompt embeddings." These embeddings capture the semantic meaning of the prompt.

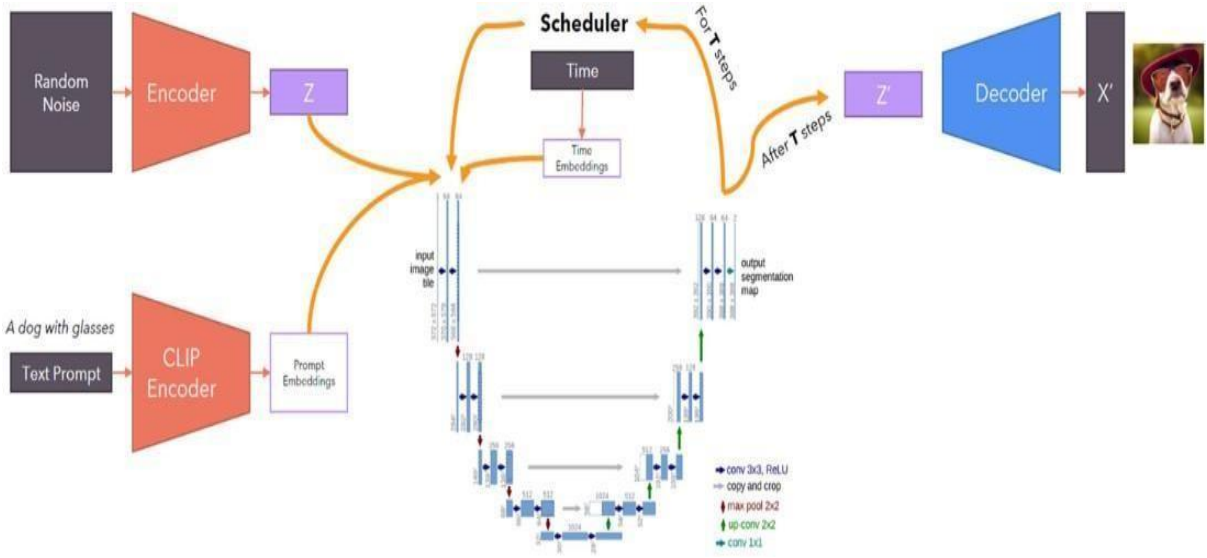
3. **U-Net and Diffusion Process:**

- **U-Net:** Now comes the core part, the U-Net. This convolutional neural network acts like a sculptor, iteratively refining the latent image based on the prompt embeddings.
- **Time Embeddings:** The U-Net incorporates an additional input called "time embeddings." These embeddings gradually increase with each iteration, signifying theprogress of the noise reduction process.
- **Noise Removal:** At each step, the U-Net analyses the current latent representation and the prompt embeddings. It then identifies and removes noise from the latent image, pushing it closer to resembling the prompt's description. This process continues for a predetermined number of iterations (typically up to 1000).
- **Pure Image Latent Representation:** As the number of iterations increases, the amount of noise in the latent representation progressively decreases. Ideally, after 1000 iterations, the latent representation becomes a "pure" version, essentially encoding the final image without any noise.

4. **Decoding and Output:**

- **Decoder:** Finally, the "pure" latent image representation is fed into the decoder.

  This decoder acts like an image generator, translating the mathematical representation back into a complete image.
- **Final Image:** The resulting image is then presented to you, reflecting the essence of your text prompt.



The input comprises random noise sampled from the training dataset, while concurrently considering a prompt for extracting the necessary features of the output image. The random noise is fed into an encoder, which converts it into a mathematical or latent representation. Simultaneously, the prompt is processed by a CLIP encoder to derive prompt embeddings, capturing its semantic essence. These embeddings and the latent representation are then forwarded to a U-Net model. The primary objective of the U-Net is to discern and mitigate noise levels based on the provided prompt, facilitating the transformation into the desired output image. The U-Net incorporates time embeddings, which dictate periodicreiterations of the refinement process. After several iterations, yielding a maximum of 1000 time embeddings, a purified image latent representation is attained. This latent representation is subsequently decoded to generate the final image, visible to the user. This is the architecture of the Text to Image generation using Stable diffusion.

Conclusion:

Text-to-image generation using the Stable Diffusion model effectively combines random noise with a semantic prompt to create coherent and visually appealing images. By leveraging a U-Net model, enhanced with time and prompt embeddings, the system iteratively refines the noise into a detailed image, accurately reflecting the prompt's content. This sophisticated process enables high-quality image synthesis from textual descriptions.