

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر

گزارش تحقیق درس یادگیری عمیق

مدل‌های انتشاری در شبکه عصبی ترانسفورمر

نگارش

رضا آدینه پور

استاد درس

جناب آقای دکتر صفابخش

تیر ۱۴۰۳

چکیده

چالش اصلی این تحقیق، بررسی و مقایسه کارایی ترنسفرمرها در برابر معماری‌های سنتی مانند U-Net در مدل‌های انتشار بوده است. نتایج به دست آمده نشان می‌دهند که ترنسفرمرها با مقیاس‌پذیری بهتر و کارایی بالاتر می‌توانند جایگزین مناسبی برای معماری‌های کنونی در مدل‌های انتشار باشند و به بهبود کیفیت تصاویر تولید شده کمک کنند. در این راستا، ترنسفرمرهای انتشار (DiTs) با افزایش عمق و عرض ترنسفرمرها و افزایش تعداد توکن‌های ورودی، توانسته‌اند بهبود قابل توجهی در معیار FID نشان دهن و نتایج برتری در بسته‌های محک ImageNet به دست آورند.

این تحقیق نشان می‌دهند که استفاده از ترنسفرمرها در مدل‌های انتشار می‌تواند راهکارهای نوینی برای بهبود کیفیت و کارایی در تولید تصاویر ارائه دهد. این رویکردها، علاوه بر بهبود عملکرد، می‌توانند به کاهش پیچیدگی‌های محاسباتی و افزایش سرعت فرآیندهای تولید تصویر کمک کنند. با توجه به این نتایج، ترنسفرمرها به عنوان یک جایگزین قوی برای معماری‌های سنتی در مدل‌های انتشار مطرح می‌شوند و می‌توانند به طور گسترده در کاربردهای مختلف مورد استفاده قرار گیرند.

کلیدواژه‌ها: یادگیری عمیق، مدل‌های انتشار، ترنسفرمر، یو-نت، کدگذار، کدگشا

فهرست مطالب

۱	۱	مقدمه
۱	۱-۱	تعريف مسئله
۱	۱-۱-۱	انتشار چیست؟
۲	۱-۱-۲	مدل‌های انتشار در یادگیری ماشین چیستند؟
۳	۲-۱	اهمیت پژوهش
۴	۳-۱	ساختار پژوهش
۵	۲	مفاهیم اولیه
۵	۱-۲	مقدمه‌ای بر مدل‌ها انتشار
۶	۱-۱-۲	فرآیند انتشار پیشرو
۷	۱-۱-۲	فرایند انتشار معکوس
۱۰	۲-۲	شبکه عصبی کانولوشنی U-Net
۱۰	۱-۲-۲	ساختار شبکه
۱۱	۳-۲	ویژن ترنسفورم
۱۲	۴-۲	نحوه عملکرد ویژن ترنسفورها
۱۲	۱-۴-۲	بینیان ترنسفورم
۱۲	۲-۴-۲	انطباق ترنسفورم برای تصاویر
۱۳	۳-۴-۲	سوگیری استقرایی و ویژن ترنسفورم

۱۴	۴-۴-۲	معماری هیبریدی
۱۴	۵-۲	مدل‌های انتشار نهان
۱۴	۱-۵-۲	معایب شبکه کانولوشنی U-Net
۱۵	۲-۵-۲	حرکت به سوی ترانسفورمرها
۱۵	۳-۵-۲	تکامل پچ‌های نهان
۱۶	۶-۲	ترانسفورمرهای انتشار (DiT) در مقابل ویژن ترانسفورمرها (ViT)
۱۶	۱-۶-۲	ترانسفورمرهای انتشار (DiT)
۱۶	۲-۶-۲	ویژن ترانسفورمرها (ViT)
۱۷	۷-۲	معماری عمومی ترانسفورمرهای انتشار
۱۷	۱-۷-۲	نمایش‌های فضایی
۱۷	۲-۷-۲	جاسازی‌های موقعیتی
۱۷	۳-۷-۲	طراحی بلوک DiT
۱۸	۳	کارهای پیشین
۲۰	۴	مقایسه مقاله‌های [۱، ۲] و نتیجه‌گیری
۲۳		مراجع

فهرست جداول

فهرست تصاویر

۱-۱	ساختار مدل‌های مولد	۳
۲-۱	نحوه عملکرد مدل‌های DDPMs	۳
۲-۲	نمونه‌ای از آموزش مدل انتشار برای داده‌های ۲ بعدی [۴]	۸
۲-۲	ساختار شبکه U-Net	۱۱
۳-۲	ساختار شبکه ViT	۱۲
۴-۲	ساختار مدل LDM	۱۴
۵-۲	ساختار مدل DiT	۱۷
۱-۴	خروجی‌های زمانی الگوریتم DREAMPlace برای دو بنچ‌مارک مختلف	۲۱
۲-۴	خروجی‌های زمانی الگوریتم OpenPARF برای بنچ‌مارک ISPД ۲۰۱۷	۲۱

فصل ۱

مقدمه

هوش مصنوعی^۱ به طور مداوم در حال تکامل است تا مشکلات سخت و پیچیده را حل کند. تولید تصویر یکی از این مشکلات پیچیده برای مدل‌های هوش مصنوعی است. VAE^۲ها، GAN^۳ها و مدل‌های جریان عملکرد خوبی داشته‌اند اما در تولید تصاویر با وضوح بالا و دقت زیاد دچار مشکل بوده‌اند. از سوی دیگر، مدل‌های انتشار^۴ در تولید تصاویر با وضوح بالا و کیفیت متنوع با دقت بالا بسیار خوب عمل می‌کنند. در حال حاضر، آن‌ها در خط مقدم انقلاب هوش مصنوعی مولد (GenAI) قرار دارند که در همه جا دیده می‌شود. مدل‌هایی مانند DALL-E-3، GLIDE، OpenAI Imagen توسط گوگل، و Stable Diffusion از جمله مدل‌های انتشار پرطرفدار هستند. در ادامه به معرفی مسئله و برخی از پیش‌نیازها می‌پردازیم.

۱-۱ تعریف مسئله

۱-۱-۱ انتشار چیست؟

انتشار یک پدیده طبیعی اساسی است که در سیستم‌های مختلف از جمله فیزیکی، شیمیایی و زیست‌شناسی مشاهده می‌شود.

این پدیده در زندگی روزمره به وضوح قابل مشاهده است. به عنوان مثال، در نظر بگیرید که عطر را اسپری می‌کنید. در ابتدا، مولکول‌های عطر به طور متراکم در نزدیکی نقطه اسپری قرار دارند. با گذشت زمان، مولکول‌ها

¹Artificial Intelligence

²Generative Adversarial Networks

³Variational Autoencoder

⁴Diffusion Models

در محیط اطراف منتشر می‌شوند.

انتشار فرآیندی است که طی آن ذرات، اطلاعات یا انرژی از ناحیه‌ای با غلظت پایین‌تر حرکت می‌کنند. این اتفاق به این دلیل رخ می‌دهد که سیستم‌ها تمایل دارند به تعادل^۵ برسند، جایی که غلظت‌ها در سراسر سیستم یکسان می‌شود.

در یادگیری ماشین^۶ و تولید داده^۷، انتشار به یک رویکرد خاص برای تولید داده‌ها با استفاده از یک فرآیند تصادفی مشابه با زنجیره مارکوف^۸ اشاره دارد. در این زمینه، مدل‌های انتشار نمونه‌های جدیدی از داده‌ها را با استفاده از داده‌های ساده‌تر ایجاد می‌کنند و به تدریج داده‌های پیچیده‌تر و واقعی‌تر تولید می‌شود.

۲-۱-۱ مدل‌های انتشار در یادگیری ماشین چیستند؟

مدل‌های انتشار مولد^۹ هستند، به این معنی که داده‌های جدیدی بر اساس داده‌هایی که بر روی آن‌ها آموزش دیده‌اند تولید می‌کنند. به عنوان مثال، یک مدل انتشار که بر روی مجموعه‌ای از داده‌های چهره‌های انسان آموزش دیده است، می‌تواند چهره‌های انسانی جدید و واقع‌گرایانه با ویژگی‌ها و حالت‌های مختلف تولید کند، حتی اگر آن چهره‌های خاص در مجموعه داده‌های اولیه وجود نداشته باشند.

برخلاف سایر مدل‌های مولدی مثل GAN، VAE و ... این مدل بر مدل‌سازی تکامل مرحله به مرحله توزیع داده‌ها از یک نقطه شروع ساده به یک توزیع پیچیده‌تر مرکز دارند. مفهوم اساسی مدل‌های انتشار این است که یک توزیع ساده مثل توزیع گاوی^{۱۰}، را از طریق یک سری عملیات‌های قابل برگشت^{۱۱} به یک توزیع داده پیچیده‌تر تبدیل کنند.

پس از اینکه مدل، فرآیند تبدیل را آموزش می‌بیند، می‌تواند نمونه‌های جدیدی را با شروع از یک نقطه در توزیع ساده و به تدریج "انتشار" آن به توزیع داده پیچیده مطلوب تولید کند.

در مدل‌های DDPMs^{۱۲} با افروzen تدریجی نویز گاوی به داده‌های اصلی در فرآیند انتشار پیشرو^{۱۳} و سپس یادگیری حذف نویز در فرآیند انتشار معکوس^{۱۴} کار می‌کنند. «شکل ۲-۱»

⁵Equilibrium

⁶Machine Learning

⁷Data Generation

⁸Markov Chain

⁹Generative

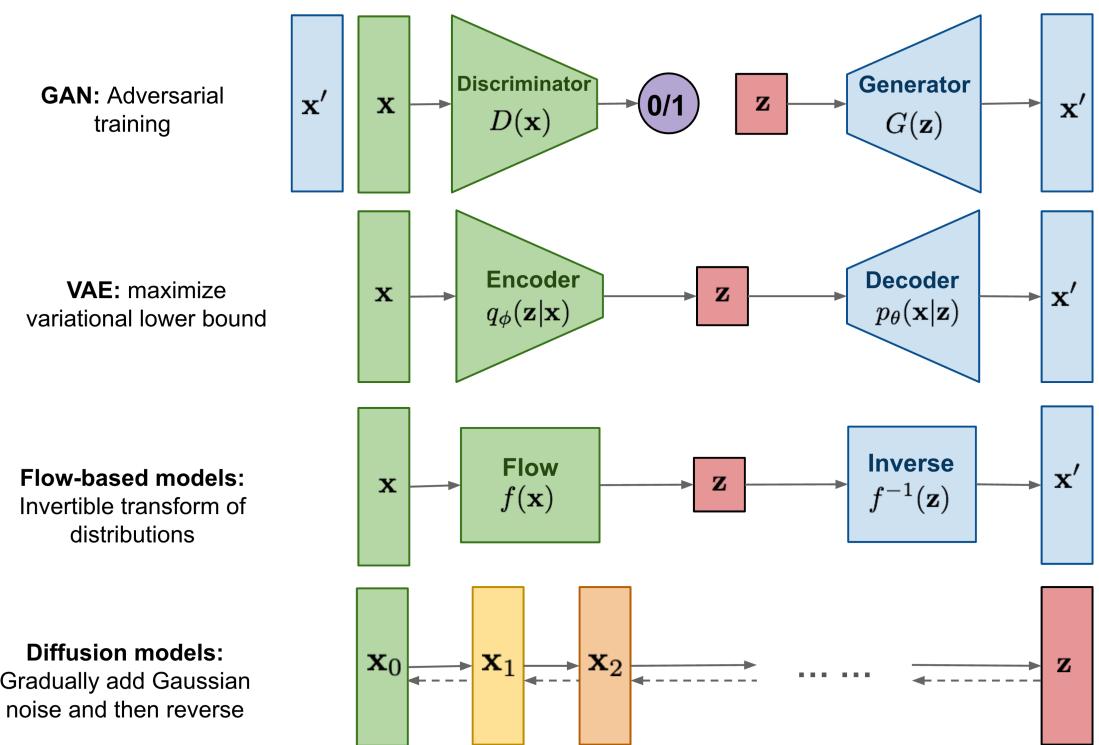
¹⁰Gaussian Distribution

¹¹Invertible Operations

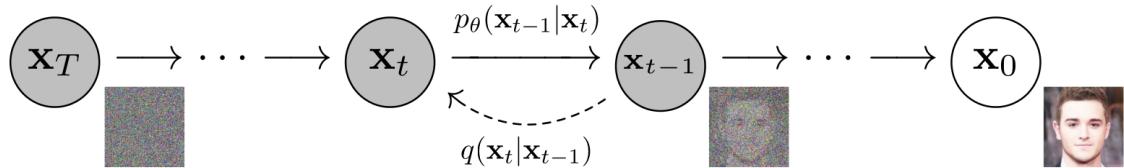
¹²Denoising Diffusion Probabilistic Models

¹³Forward Diffusion

¹⁴Reverse Diffusion



شکل ۱-۱: ساختار مدل‌های مولد



شکل ۲-۱: نحوه عملکرد مدل‌های DDPMs [۵]

۲-۱ اهمیت پژوهش

مدل‌های انتشار با شبیه‌سازی فرآیندهای تصادفی قادر به تولید نمونه‌های واقعی‌تری هستند که در کاربردهای مختلفی مانند ترمیم تصاویر، نویززدایی و تولید تصاویر با وضوح بالا استفاده می‌شوند. به ویژه، ترانسفورمرهای انتشار (DiT) به عنوان یک نوآوری در این حوزه با استفاده از معماری ترانسفورمر، قدرت بیشتری در مدل‌سازی پیچیدگی‌های داده‌ها دارند. DiT‌ها با توانایی مقیاس‌پذیری بالا و کاهش مداوم معیار FID، بهبود قابل توجهی در کیفیت و واقع‌گرایی نمونه‌های تولیدی ارائه می‌دهند. این ویژگی‌ها DiT‌ها را به ابزار قدرتمندی برای وظایف پیچیده‌تری مانند سنتز تصویر و بهبود کیفیت تصویر تبدیل می‌کنند.

۳-۱ ساختار پژوهش

این پژوهش در ۵ فصل انجام شده است. در فصل ۱ به مقدمه و اهمیت موضوع پژوهش پرداخته شده است. در فصل ۲ به مفاهیم اولیه و پیش‌نیازها پرداخته شده است. در ادامه در فصل ۳ پژوهش به بررسی کارهای پیشین انجام شده در این زمینه پرداخت شده است. در فصل چهارم به بررسی دقیق و جزئی مقالات مطالعه شده در این پژوهش پرداخته شده است و در فصل پایانی، جمع‌بندی و نتیجه‌گیری پژوهش ارائه شده است.

فصل ۲

مفاهیم اولیه

همانطور که در فصل ۱ بیان شد، برای درک بهتر مدل‌های انتشار، ابتدا می‌بایست پیش‌نیازها و مفاهیم اولیه این زمینه را بیان کنیم. در ادامه به معرفی و بررسی مدل‌های انتشار و ساختارهای معروف و مورد استفاده در مقالات مرتبط می‌پردازیم:

۱-۲ مقدمه ای بر مدل‌ها انتشار

مدل‌های انتشار نوعی مدل مولد هستند که یک زنجیره مارکوف را شبیه‌سازی می‌کنند تا از یک توزیع اولیه ساده به توزیع داده‌ها پیچیده انتقال یابند. این فرآیند شبیه به حرکت پراونی یک ذره است که هر مرحله آن یک حرکت تصادفی کوچک است. به همین دلیل به آن‌ها "مدل‌های انتشار" گفته می‌شود.

مدل‌های انتشار در کاربردهای مختلفی مانند نویز‌زدایی^۱، افزایش وضوح تصویر^۲ و ترمیم تصاویر^۳ استفاده می‌شود. یکی از مزایای کلیدی مدل‌های انتشار توانایی آن‌ها در تولید نمونه‌های با کیفیت بالا است که آن‌ها را به‌ویژه برای وظایفی مانند سنتر تصویر بسیار مفید می‌سازد.

به‌طور کلی فرآیند انتشار از دو مرحله تشکیل می‌شود:

۱. انتشار پیش‌رو

۲. انتشار پس‌رو

¹Denoising

²Super-resolution

³Inpainting

که در ادامه به توضیح هر کدام می‌پردازیم:

۱-۱-۲ فرآیند انتشار پیشرو

فرآیند انتشار پیشرو یک زنگیره مارکوف از مراحل انتشار است که در آن به تدریج و به صورت تصادفی نویز به داده‌های اصلی اضافه می‌کنیم.

با فرض اینکه یک نقطه داده از یک توزیع داده واقعی $(x \sim q)$ نمونه‌برداری شده باشد، یک فرآیند انتشار پیشرو را تعریف کنیم که در آن مقدار کمی نویز گاووسی به نمونه در T مرحله اضافه می‌کنیم، به طوری که یک دنباله از نمونه‌های نویزی x_T, \dots, x_1 تولید می‌شود. اندازه مراحل با یک برنامه تغییرپذیری $\{\beta_t \in (0, 1)\}_{t=1}^T$ کنترل می‌شود.

$$q(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t \mathbf{I}) \quad (1-2)$$

$$q(x_{1:T} | x_0) = \prod_{t=1}^T q(x_t | x_{t-1}) \quad (2-2)$$

با بزرگتر شدن t ، نمونه داده x به تدریج ویژگی‌های قابل تشخیص خود را از دست می‌دهد در نهایت، زمانی که $T \rightarrow \infty$ معادل یک توزیع گاووسی همسانگرد^۴ خواهد بود.

یکی از ویژگی‌های جالب فرآیند فوق این است که می‌توانیم x_t را در هر لحظه زمانی دلخواه t با استفاده از ترفند بازپارامتری‌سازی^۵ به صورت بسته نمونه‌برداری کنیم. اگر $\bar{\alpha}_t = \prod_{i=1}^t \alpha_i$ و $\bar{\beta}_t = 1 - \beta_t$ باشد داریم:

⁴Isotropic

⁵Reparameterization

$$\begin{aligned}
\mathbf{x}_t &= \sqrt{\alpha_t} \mathbf{x}_{t-1} + \sqrt{1 - \alpha_t} \boldsymbol{\epsilon}_{t-1} \\
&= \sqrt{\alpha_t \alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{1 - \alpha_t \alpha_{t-1}} \tilde{\boldsymbol{\epsilon}}_{t-2} \\
&= \dots \\
&= \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \boldsymbol{\epsilon}
\end{aligned} \tag{3-2}$$

$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I})$

توجه به این نکته مهم است که وقتی دو توزیع گاوی با واریانس‌های مختلف $\mathcal{N}(\mathbf{0}, \sigma_1^2 \mathbf{I})$ و $\mathcal{N}(\mathbf{0}, \sigma_2^2 \mathbf{I})$ را باهم ترکیب می‌کنیم، توزیع جدید $\mathcal{N}(\mathbf{0}, (\sigma_1^2 + \sigma_2^2) \mathbf{I})$ است. در اینجا انحراف استاندارد ترکیبی برابر است با:

$$\sqrt{(1 - \alpha_t) + \alpha_t(1 - \alpha_{t-1})} = \sqrt{1 - \alpha_t \alpha_{t-1}} \tag{4-2}$$

معمولًاً، می‌توانیم گام به روزرسانی بزرگتری داشته باشیم وقتی که نمونه‌ها نویز بیشتری پیدا می‌کنند، بنابراین $\cdot \bar{\alpha}_T < \beta_1 < \beta_2 < \dots < \beta_T$ و در نتیجه

۴-۱-۲ فرایند انتشار معکوس

فرایند انتشار معکوس سعی می‌کند فرایند انتشار پیشرو را به صورت معکوس انجام دهد و از داده‌های نویزی، داده‌های اصلی را با کیفیت بالا تولید کند.

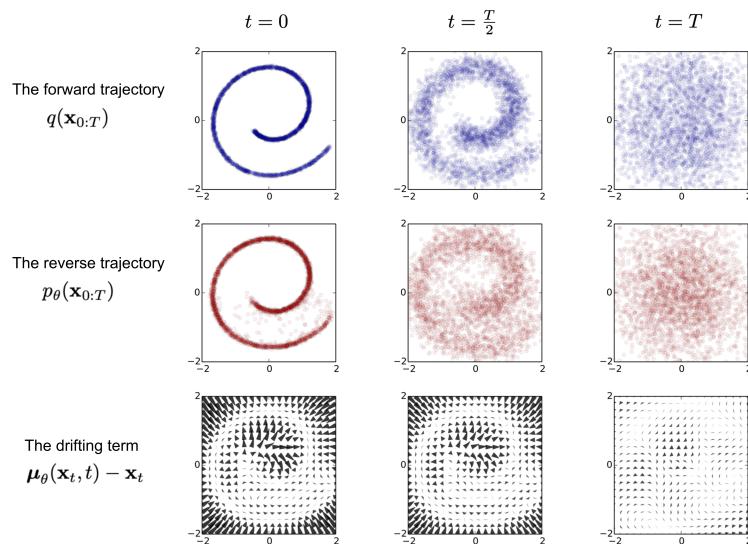
اگر بتوانیم فرایند فوق را معکوس کنیم و از $q(\mathbf{x}_{t-1} | \mathbf{x}_t)$ نمونه‌برداری کنیم، قادر خواهیم بود نمونه واقعی را از یک ورودی نویز گاوی بازسازی کنیم:

$$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{5-2}$$

اگر مقدار β_t به اندازه کافی کوچک باشد، $q(\mathbf{x}_{t-1} | \mathbf{x}_t)$ نیز گاوی خواهد بود. متاسفانه، نمی‌توانیم به راحتی $q(\mathbf{x}_{t-1} | \mathbf{x}_t)$ را تخمین بزنیم زیرا نیاز به استفاده از کل مجموعه داده‌ها دارد و بنابراین باید یک مدل p_θ را بیاموزیم تا این احتمالات شرطی را تقریب بزند تا بتوانیم فرایند انتشار معکوس را اجرا کنیم.

$$p_{\theta}(\mathbf{x}_{\circ:T}) = p(\mathbf{x}_T) \prod_{t=1}^T p_{\theta}(\mathbf{x}_{t-1} | \mathbf{x}_t) \quad (8-2)$$

$$p_{\theta}(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_{\theta}(\mathbf{x}_t, t), \Sigma_{\theta}(\mathbf{x}_t, t)) \quad (8-3)$$



شکل ۱-۲: نمونه‌ای از آموزش مدل انتشار برای داده‌های ۲ بعدی [۴]

قابل توجه است که احتمال شرطی معکوس زمانی که بر روی \mathbf{x}_\circ شرطی شود، به صورت زیر قابل محاسبه است:

$$q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_\circ) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\mu}(\mathbf{x}_t, \mathbf{x}_\circ), \tilde{\beta}_t \mathbf{I}) \quad (8-4)$$

با استفاده از قاعده بیز^۶، می‌توان نوشت:

⁶Bayes' rule

$$\begin{aligned}
q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_o) &= \frac{q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_o) q(\mathbf{x}_{t-1} | \mathbf{x}_o)}{q(\mathbf{x}_t | \mathbf{x}_o)} \\
&\propto \exp \left(-\frac{1}{2} \left(\frac{(\mathbf{x}_t - \sqrt{\alpha_t} \mathbf{x}_{t-1})^T}{\beta_t} + \frac{(\mathbf{x}_{t-1} - \sqrt{\bar{\alpha}_t} \mathbf{x}_o)^T}{1 - \alpha_{t-1}^-} \right. \right. \\
&\quad \left. \left. - \frac{(\mathbf{x}_t - \sqrt{\alpha_t} \mathbf{x}_o)^T}{1 - \bar{\alpha}_t} \right) \right) \\
&= \exp \left(-\frac{1}{2} \left(\frac{\mathbf{x}_t^T - 2\sqrt{\alpha_t} \mathbf{x}_t \mathbf{x}_{t-1} + \alpha_t \mathbf{x}_{t-1}^T}{\beta_t} \right. \right. \\
&\quad \left. \left. + \frac{\mathbf{x}_{t-1}^T - 2\sqrt{\bar{\alpha}_t} \mathbf{x}_{t-1} \mathbf{x}_o + \bar{\alpha}_t \mathbf{x}_o^T}{1 - \alpha_{t-1}^-} \right. \right. \\
&\quad \left. \left. - \frac{\mathbf{x}_t^T - 2\sqrt{\alpha_t} \mathbf{x}_t \mathbf{x}_o + \bar{\alpha}_t \mathbf{x}_o^T}{1 - \bar{\alpha}_t} \right) \right) \\
&= \exp \left(-\frac{1}{2} \left(\left(\frac{\alpha_t}{\beta_t} + \frac{1}{1 - \alpha_{t-1}^-} \right) \mathbf{x}_{t-1}^T \right. \right. \\
&\quad \left. \left. - \left(\frac{2\sqrt{\alpha_t}}{\beta_t} \mathbf{x}_t + \frac{2\sqrt{\bar{\alpha}_t}}{1 - \alpha_{t-1}^-} \mathbf{x}_o \right) \mathbf{x}_{t-1}^T \right. \right. \\
&\quad \left. \left. + \frac{2\sqrt{\bar{\alpha}_t}}{1 - \bar{\alpha}_t} \mathbf{x}_t + C(\mathbf{x}_t, \mathbf{x}_o) \right) \right) \\
\end{aligned} \tag{9-2}$$

با پیروی از تابع چگالی گاوی استاندارد^۴، میانگین و واریانس به صورت زیر بیان می‌شوند. (به یاد داریم که $\bar{\alpha}_t = \prod_{i=1}^t \alpha_i$ و $\alpha_t = 1 - \beta_t$)

$$\tilde{\beta}_t = \left(\frac{\alpha_t}{\beta_t} + \frac{1}{1 - \alpha_{t-1}^-} \right)^{-1} = \left(\frac{\alpha_t - \bar{\alpha}_t + \beta_t}{\beta_t(1 - \alpha_{t-1}^-)} \right)^{-1} = \beta_t \frac{1 - \alpha_{t-1}^-}{1 - \bar{\alpha}_t} \tag{10-2}$$

$$\begin{aligned}
\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_o) &= \left(\frac{\sqrt{\alpha_t}}{\beta_t} \mathbf{x}_t + \frac{\sqrt{\alpha_{t-1}^-}}{1 - \alpha_{t-1}^-} \mathbf{x}_o \right) \left(\frac{\alpha_t}{\beta_t} + \frac{1}{1 - \alpha_{t-1}^-} \right)^{-1} \\
&= (\sqrt{\alpha_t} \mathbf{x}_t + \sqrt{\bar{\alpha}_t} \mathbf{x}_o) \left(\frac{1 - \alpha_{t-1}^-}{1 - \bar{\alpha}_t} \right) \\
&= \frac{\sqrt{\alpha_t(1 - \bar{\alpha}_t)}}{1 - \bar{\alpha}_t} \mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_t \beta_t}}{1 - \bar{\alpha}_t} \mathbf{x}_o.
\end{aligned} \tag{11-2}$$

^۴Standard Gaussian Density Function

۲-۲ شبکه عصبی کانولوشنی U-Net

در معماری‌های رایج شبکه‌های کانولوشنی، کانال‌های ویژگی به منظور استخراج اطلاعات معنایی به طور متناوب از لایه‌های ادغام^۸ عبور می‌کنند. لایه‌های ادغام با کاهش اطلاعات فضایی میدان دید لایه‌های بعدی را افزایش می‌دهند و از این طریق شبکه می‌تواند اطلاعات معنایی را از تصاویر استخراج کند. هرچه از ورودی یک شبکه کانولوشنی به سمت خروجی آن حرکت کنیم، اطلاعات فضایی کاسته شده و بر اطلاعات معنایی افزوده می‌شود؛ بنابراین به نظر می‌رسد بین اطلاعات فضایی و معنایی در این شبکه‌ها مصالحه^۹ ای وجود دارد. که افزایش هریک، موجب کاهش دیگری می‌شود. شبکه یو-نت به این چالش بزرگ از طریق معماری خاص خود و اتصالات پرشی رسیدگی می‌کند. این شبکه دارای یک ساختار متقارن-کدگذار-کدگشا می‌باشد (شکل ۲-۲) که در آن کانال‌های ویژگی در سطوح مختلفی از بخش کدگذار، از طریق اتصال پرشی، به کانال‌های ویژگی کدگشا الحق^{۱۰} می‌شود. این ساختار برخلاف ساختار شبکه تمام کانولوشنی [۹]، که خروجی نهایی را در یک مرحله نمونه افزایی^{۱۱} ایجاد می‌کند، باعث می‌شود بازیابی اطلاعات فضایی بهتر صورت بگیرد. تفاوت دیگر این دو شبکه این است که اتصالات پرشی در شبکه یو-نت به جای استفاده از عمل جمع، از عمل الحق استفاده می‌کند. این ساختار متقارن، به همراه تکنیک‌های افرونگی داده، باعث می‌شود شبکه یو-نت بتواند علاوه بر کسب دقت بالا، از تعداد بسیار کمی داده‌های آموزشی را یاد بگیرد.

۱-۲-۲ ساختار شبکه

شکل (۲-۲) ساختار شبکه یو-نت را نشان می‌دهد. این شبکه از یک بخش کدگذار «سمت چپ» و یک بخش کدگشا «سمت راست» تشکیل شده است. هر مرحله از بخش کدگذار، شامل دو لایه کانولوشنی 3×3 با یک تابع فعالسازی^{۱۲} ReLU در هر کدام و یک لایه ادغام حداقل^{۱۳} با سایز 2×2 و گام ۲ می‌باشد. در هر مرحله از بخش کدگشا، ابتدا یک لایه نمونه افزایی به همراه یک لایه کانولوشنی 2×2 «کانولوشن افزاینده^{۱۴}» تعداد کانال‌های ویژگی را نصف و ابعاد آن را دو برابر می‌کند. سپس کانال‌های ویژگی با کانال‌های برش داده شده متناظر از بخش کدگذار الحق می‌شوند. در ادامه دولایه کانولوشنی، مشابه آنچه در بخش کدگذار وجود دارد، روی کانال‌های ویژگی اعمال می‌شود. در لایه‌های کانولوشنی این شبکه عمل گسترش مرز با صفر^{۱۵} انجام نمی‌شود و بنابراین برای الحق کانال‌های ویژگی، همانطور که اشاره شد، کانال‌های ویژگی بخش کدگذار بریده می‌شوند.

⁸Pooling

⁹Trade-off

¹⁰Concatenate

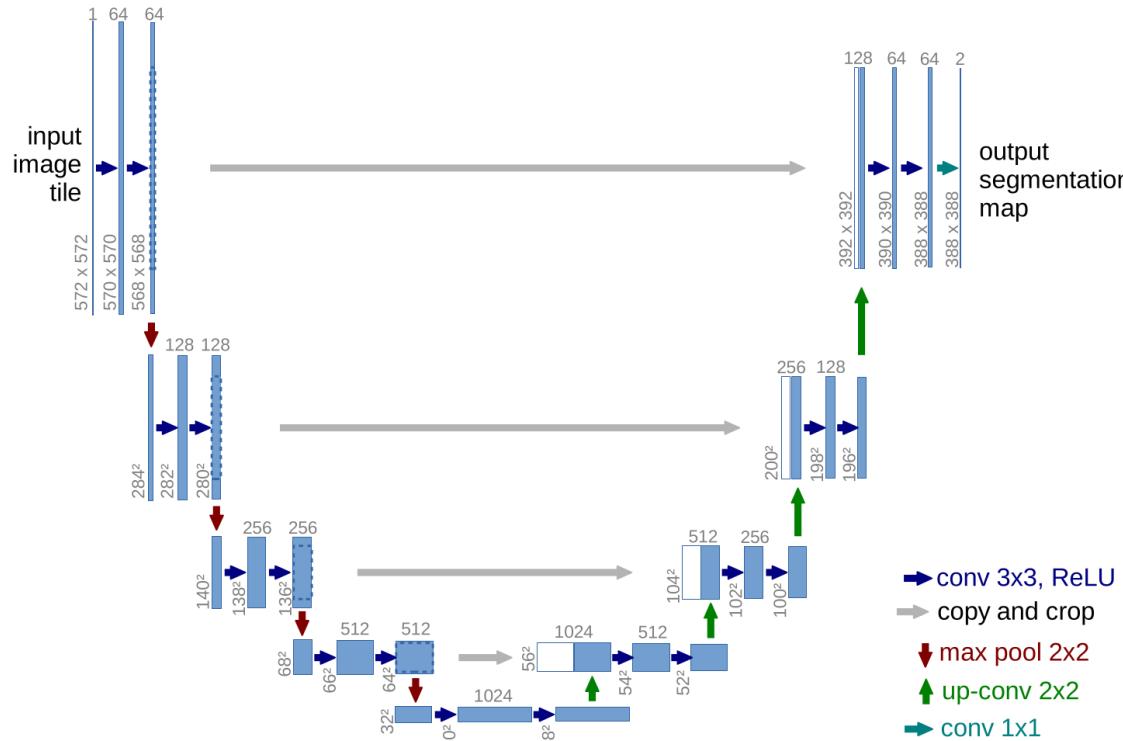
¹¹Up-Sampling

¹²Activation Function

¹³Max Pooling

¹⁴Up-Convolution

¹⁵Zero Padding



شکل ۲-۲: ساختار شبکه [۵] U-Net

در انتهای بخش کدگشا یک لایه کانولوشنی 1×1 تعداد کانال‌های ویژگی «۶۴» را به تعداد کلاس‌های مسئله تبدیل می‌کند.

۳-۲ ویژن ترنسفورمر

ویژن ترنسفورمرها (ViT) مدلی جدید در زمینه بینایی کامپیوترا^{۱۶} هستند که مدل‌های ترنسفورمر را که در اصل برای وظایف پردازش زبان طبیعی طراحی شده‌اند، به وظایف طبقه‌بندی تصاویر اعمال می‌کنند. برخلاف شبکه‌های عصبی کانولوشنی سنتی که تصاویر را به صورت سلسله مراتبی پردازش می‌کنند، ViT ها به صورت موازی تصاویر را پردازش می‌کنند.

شبکه‌های ViT تصاویر را به عنوان دنباله‌ای از تکه‌ها در نظر می‌گیرند و وابستگی‌های جهانی^{۱۷} بین آن‌ها را ثبت می‌کنند. این ویژگی به آن‌ها امکان مدل‌سازی تعاملات پیکسلی با برد بلند را می‌دهد. یکی از مزایای کلیدی ViT ها قابلیت مقیاس‌پذیری آن‌ها است. آن‌ها می‌توانند بر روی مجموعه داده‌های بزرگ آموزش بینند و تصاویر بزرگ را به عنوان ورودی قبول کنند.

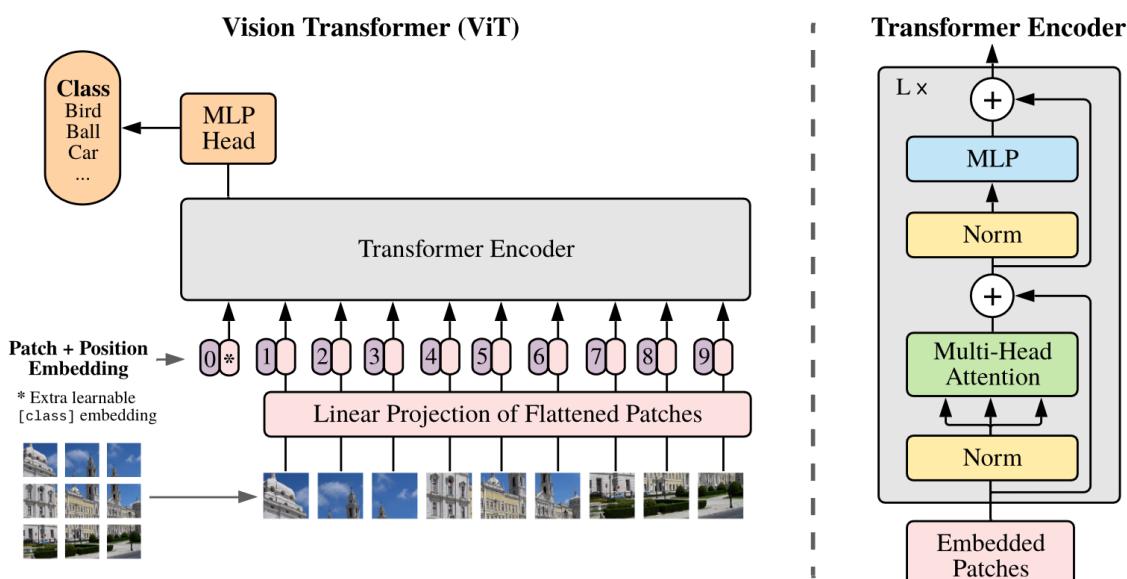
¹⁶Computer Vision

¹⁷Global Dependencies

۴-۲ نحوه عملکرد ویژن ترنسفرها

۱-۴-۲ بینیان ترنسفرمر

برای درک چگونگی عملکرد ویژن ترنسفرمرها، لازم است مفاهیم پایه‌ای معماری ترنسفرمر مانند توجه به خود^{۱۸} را درک کنیم. توجه به خود یک مکانیزم است که به مدل اجازه می‌دهد تا هنگام پیش‌بینی، اهمیت عناصر مختلف در یک دنباله را وزن‌دهی کند و در نتیجه به نتایج چشمگیری در وظایف مبتنی بر دنباله دست یابد.



شکل ۳-۲: ساختار شبکه [۶] ViT

۲-۴-۲ انطباق ترنسفرمر برای تصاویر

مفهوم توجه به خود برای پردازش تصاویر با استفاده از ویژن ترنسفرمرها تطبیق داده شده است. برخلاف داده‌های متنی، تصاویر به طور ذاتی دو بعدی هستند و شامل پیکسل‌هایی هستند که در ردیف‌ها و ستون‌ها قرار گرفته‌اند. برای مقابله با این چالش، ویژن ترنسفرمرها تصاویر را به دنباله‌هایی تبدیل می‌کنند که می‌توانند توسط ترانسفورمر پردازش شوند.

مراحل پردازش در ViT ها را می‌توان به صورت زیر تقسیم‌بندی نمود:

- تقسیم تصویر به پچ‌ها:

اولین مرحله در پردازش یک تصویر با ویژن ترنسفرمر، تقسیم آن به پچ‌های کوچکتر و با اندازه ثابت است. هر پچ نمایانگر یک ناحیه محلی از تصویر است.

¹⁸Self-Attention

- تخت کردن پچ‌ها:

درون هر پچ، مقادیر پیکسل‌ها به یک بردار واحد تخت می‌شوند. این فرآیند تخت کردن به مدل اجازه می‌دهد تا پچ‌های تصویر را به عنوان داده‌های دنباله‌ای پردازش کند.

- تولید جاسازی‌های خطی با بعد کمتر:

این بردارهای پچ تخت شده سپس با استفاده از تبدیل‌های خطی قابل آموزش به یک فضای با بعد کمتر نگاشت می‌شوند. این مرحله بعد داده‌ها را کاهش می‌دهد در حالی که ویژگی‌های مهم را حفظ می‌کند.

- اضافه کردن رمزگذاری‌های موقعیتی:

برای حفظ اطلاعات در مورد ترتیب فضایی پچ‌ها، رمزگذاری‌های موقعیتی اضافه می‌شوند. این رمزگذاری‌ها به مدل کمک می‌کنند تا موقعیت نسبی پچ‌های مختلف در تصویر را درک کند.

- تغذیه دنباله به رمزگذار ترانسفورمر:

ورودی به یک رمزگذار ترانسفورمر استاندارد شامل دنباله‌ای از جاسازی‌های پچ و جاسازی‌های موقعیتی است. این رمزگذار از لایه‌های متعددی تشکیل شده است که هر کدام شامل دو جزء مهم هستند: مکانیزم‌های توجه به خود چندسری (MSP‌ها)، که مسئولیت محاسبه وزن‌های توجه برای اولویت دادن به عناصر دنباله ورودی در حین پیش‌بینی‌ها را دارند، و پرسپترون چندلایه (MLP) بلوک‌ها. قبل از هر بلوک، نرمال‌سازی لایه (LN) برای مقیاس‌بندی و مرکزیت داده‌ها درون لایه اعمال می‌شود، که پایداری و کارایی آموزش را تضمین می‌کند. در طول آموزش، یک بهینه‌ساز نیز برای تنظیم ابرپارامترهای مدل در پاسخ به از دست رفتن محاسبه شده در هر تکرار آموزشی استفاده می‌شود.

- طبقه‌بندی:

برای فعال کردن طبقه‌بندی تصویر، یک "توکن طبقه‌بندی" ویژه به دنباله جاسازی‌های پچ اضافه می‌شود. حالت نهایی این توکن در خروجی رمزگذار ترانسفورمر به عنوان نماینده کل تصویر عمل می‌کند.

۳-۴-۲ سوگیری استقرایی و ویژن ترانسفورمر

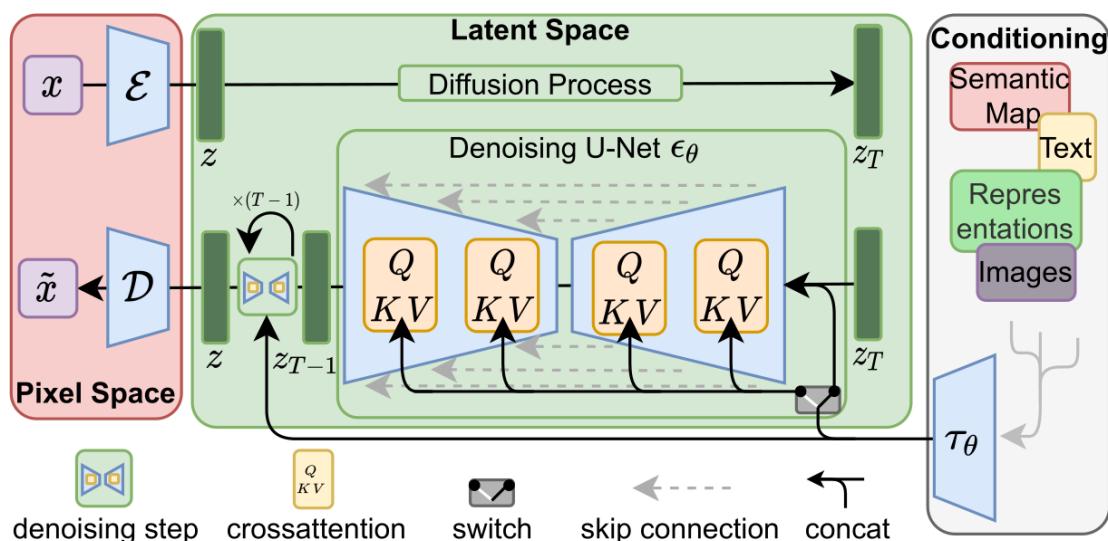
ویژن ترانسفورمرها نسبت به CNN‌ها سوگیری استقرایی خاص تصویر کمتری دارند. در CNN‌ها، مفاهیمی مانند محلی بودن، ساختار همسایگی دو بعدی و تعادل انتقال در هر لایه از مدل تعییه شده‌اند. اما ویژن ترانسفورمرها بر لایه‌های توجه به خود برای زمینه‌یابی جهانی تکیه می‌کنند و فقط از ساختار همسایگی دو بعدی در مراحل اولیه استخراج پچ استفاده می‌کنند. این بدان معنی است که ویژن ترانسفورمرها بیشتر به یادگیری روابط فضایی از ابتدا تکیه دارند و دیدگاه متفاوتی از درک تصویر ارائه می‌دهند.

۴-۴-۲ معماری هیبریدی

علاوه بر استفاده از پچ‌های تصویر خام، ویژن ترانسفورمرها همچنین گزینه‌ای برای یک معماری هیبریدی فراهم می‌کنند. با این روش، دنباله‌های ورودی می‌توانند از نقشه‌های ویژگی استخراج شده توسط یک CNN تولید شوند. این سطح از انعطاف‌پذیری اجازه می‌دهد تا نقاط قوت CNN‌ها و ترانسفورمرها را در یک مدل ترکیب کرد و امکانات بیشتری برای بهینه‌سازی عملکرد را ارائه داد.

۵-۲ مدل‌های انتشار نهان

مدل‌های انتشار نهان^{۱۹} نوعی مدل مولد هستند که با مدل سازی داده‌ها به عنوان یک فرآیند انتشار، یاد می‌گیرند داده‌ها را تولید کنند. این فرآیند با یک توزیع اولیه ساده، مانند نویز گاووسی، آغاز می‌شود و به تدریج از طریق یک سری مراحل کوچک آن را به توزیع هدف تبدیل می‌کند. هر مرحله توسط یک شبکه عصبی هدایت می‌شود که برای معکوس کردن فرآیند انتشار آموزش داده شده است. مدل‌های LDMs در تولید نمونه‌های با کیفیت بالا در حوزه‌های مختلف، از جمله تصاویر، متن و صدا موفق بوده‌اند. [۷]



شکل ۴-۲: ساختار مدل LDM [۷]

۱-۵-۲ معایب شبکه کانولوشنی U-Net

شبکه‌های U-Net در بسیاری از وظایف بینایی کامپیوتری به دلیل توانایی آن‌ها در استخراج ویژگی‌های محلی و حفظ وضوح فضایی به یک اصل اساسی تبدیل شده‌اند. با این حال، این شبکه‌ها محدودیت‌هایی دارند. برای

¹⁹Latent Diffusion Models

مثال، آن‌ها اغلب در به دست آوردن وابستگی‌های بلندمدت و زمینه جهانی در داده‌های ورودی مشکل دارند. این به این دلیل است که میدان پذیرش یک لایه کانولوشنی محلی و محدود است و افزایش آن نیاز به شبکه‌های عمیق‌تر و فیلترهای بزرگ‌تر دارد که خود چالش‌های جدیدی را به همراه دارند.

علاوه بر این، عملیات کانولوشن در Net-U‌ها نسبت به انتقال مکان مقاوم است، به این معنی که یک ویژگی را بدون توجه به موقعیت آن در تصویر به همان صورت پردازش می‌کند. این می‌تواند در وظایفی که موقعیت مطلق ویژگی‌ها اهمیت دارد، یک عیب حساب شود.

۲-۵-۲ حرکت به سوی ترانسفورمرها

ترانسفورمرها که در اصل برای وظایف پردازش زبان طبیعی طراحی شده‌اند، پتانسیل بالایی در وظایف بینایی کامپیوتری نشان داده‌اند. برخلاف شبکه‌های کانولوشنی، ترانسفورمرها می‌توانند وابستگی‌های بلندمدت را بدون نیاز به شبکه‌های عمیق یا فیلترهای بزرگ مدل‌سازی کنند. این به دلیل استفاده از مکانیزم‌های توجه به خود است که به هر عنصر ورودی اجازه می‌دهد با تمام عناصر دیگر، بدون توجه به فاصله‌شان، تعامل داشته باشد. علاوه بر این، ترانسفورمرها نسبت به انتقال مکان مقاوم نیستند، به این معنی که می‌توانند موقعیت مطلق ویژگی‌ها را دریافت کنند. این از طریق استفاده از رمزگذاری‌های موقعیتی محقق می‌شود که اطلاعاتی در مورد موقعیت هر عنصر در ورودی اضافه می‌کند.

۳-۵-۲ تکامل پچ‌های نهان

مفهوم پچ‌های نهان از نیاز به ایجاد کارایی محاسباتی ترانسفورمرها برای تصاویر با وضوح بالا نشأت گرفته است. اعمال ترانسفورمرها مستقیماً به پیکسل‌های خام تصاویر با وضوح بالا به دلیل پیچیدگی توجه به خود که به تعداد عناصر به صورت درجه دوم افزایش می‌یابد، محاسباتی پرهزینه است. برای غلبه بر این مشکل، تصویر به پچ‌های کوچکی تقسیم می‌شود و به ترانسفورمرها اعمال می‌شوند. این کار به طور قابل توجهی تعداد عناصر و در نتیجه پیچیدگی محاسباتی را کاهش می‌دهد. این روش به ترانسفورمرها اجازه می‌دهد تا هم ویژگی‌های محلی درون هر پچ و هم زمینه جهانی بین پچ‌ها را دریافت کنند.

۶-۲ ترانسفورمرهای انتشار (DiT) در مقابل ویژن ترانسفورمرها (ViT)

در حالی که هر دو DiT و ViT از ترانسفورمرها به عنوان معماری و بدنه اصلی خود استفاده می‌کنند و بر روی پچ‌های نهان عمل می‌کنند، تفاوت آن‌ها در نحوه تولید تصاویر و جزئیات معماری خاص آن‌هاست.

۱-۶-۲ ترانسفورمرهای انتشار (DiT)

مدل DiT از ترانسفورمرها در یک فرآیند انتشار نهان استفاده می‌کند، جایی که یک توزیع اولیه ساده (مانند نویز گاوسی) به تدریج به تصویر هدف تبدیل می‌شود. این کار با معکوس کردن فرآیند انتشار که توسط یک شبکه ترانسفورمر هدایت می‌شود، انجام می‌گیرد. یکی از جنبه‌های مهم DiT مفهوم بازه‌های زمانی انتشار است. این بازه‌های زمانی نمایانگر مراحل فرآیند انتشار هستند و شبکه ترانسفورمر در هر مرحله به بازه زمانی شرطی می‌شود. این ویژگی به شبکه اجازه می‌دهد تا ویژگی‌های مختلف را در مراحل مختلف فرآیند انتشار تولید کند. DiT همچنین می‌تواند به "برچسب‌های کلاس" شرطی شود و بدین ترتیب تصاویر مربوط به کلاس‌های خاصی را تولید کند.

۲-۶-۲ ویژن ترنسفرمرها (ViT)

ویژن ترنسفرمرها از ترنسفرمرها برای تولید مستقیم تصویر به صورت خودرگرسیو^{۲۰} استفاده می‌کنند، جایی که هر پچ یکی پس از دیگری تولید می‌شود و به پچ‌های قبل^{۲۱} تولید شده شرطی می‌شود. یکی از اجزای کلیدی ViT استفاده از لایه‌های نرم‌افزاری تطبیقی^{۲۱} است. این لایه‌ها به طور تطبیقی ویژگی‌ها را بر اساس آمار دسته فعلی مقیاس‌بندی و تغییر می‌دهند که به پایداری آموزش و بهبود عملکرد مدل کمک می‌کند. در حالی که هر دو رویکرد نقاط قوت و ضعف خود را دارند، آن‌ها دو جهت امیدوارکننده برای استفاده از ترنسفرمرها در مدل‌سازی مولد تصاویر را نشان می‌دهند. انتخاب بین DiT و ViT به نیازهای خاص مورد نظر بستگی دارد.

به عنوان مثال، اگر کاربرد مورد نظر تولید تصاویر از کلاس‌های خاصی باشد، DiT ممکن است به دلیل توانایی آن در شرطی شدن بر روی برچسب‌های کلاس انتخاب بهتری باشد. از سوی دیگر، اگر وظیفه نیاز به تولید تصاویر با وضوح بالا داشته باشد، ViT ممکن است به دلیل استفاده از لایه‌های adaLN که می‌توانند به پایداری آموزش مدل‌های بزرگ کمک کنند، مناسب‌تر باشد.

²⁰ Autoregressive

²¹ Adaptive Layer Norm

۷-۲ معماری عمومی ترانسفورمرهای انتشار

۱-۷-۲ نمایش‌های فضایی

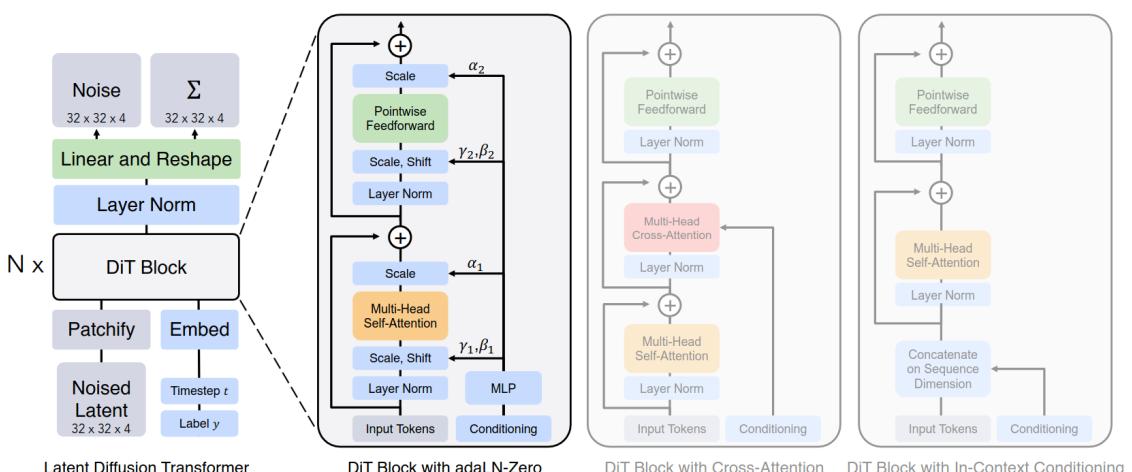
مدل ابتدا نمایش‌های فضایی را از طریق یک لایه شبکه ورودی می‌گیرد و ورودی‌های فضایی را به دنباله‌ای از توکن‌ها تبدیل می‌کند. این فرآیند به مدل اجازه می‌دهد تا اطلاعات فضایی موجود در داده‌های تصویر را پردازش کند. این یک گام حیاتی است زیرا داده‌های ورودی را به فرمتی تبدیل می‌کند که ترانسفورمر بتواند به طور مؤثر پردازش کند.

۲-۷-۲ جاسازی‌های موقعیتی

جاسازی‌های موقعیتی یک جزء حیاتی از معماری ترانسفورمر هستند. آن‌ها مدل را با اطلاعاتی درباره موقعیت هر توکن در دنباله فراهم می‌کنند. در DiT‌ها، جاسازی‌های موقعیتی استاندارد ویژن ترانسفورمر به همه توکن‌های ورودی اعمال می‌شود. این فرآیند به مدل کمک می‌کند تا موقعیت‌های نسبی و روابط بین بخش‌های مختلف تصویر را درک کند.

۳-۷-۲ طراحی بلوك DiT

در یک مدل انتشار معمولی، یک شبکه عصبی کانولوشنی یاد می‌گیرد که نویز را از تصویر حذف کند. DiT‌ها این U-Net را با یک ترانسفورمر جایگزین می‌کنند. این جایگزینی نشان می‌دهد که سوگیری استقرایی برای عملکرد مدل‌های انتشار ضروری نیست.



شکل ۲-۵-۲: ساختار مدل DiT

فصل ۳

کارهای پیشین

ترنسفورمرها (Transformers) به عنوان جایگزینی برای معماری‌های خاص حوزه‌ها در زمینه‌های مختلفی از جمله زبان، بینایی، یادگیری تقویتی و فرآگیری متأثر معرفی شده‌اند. این مدل‌ها نشان داده‌اند که با افزایش اندازه مدل، توان محاسباتی و داده‌های آموخته مقياس‌پذیری قابل توجهی دارند. در خارج از حوزه زبان، ترانسفورمرها برای پیش‌بینی خودرگرسیو پیکسل‌ها و کدبک‌های گستاخ آموخته شده‌اند و در مدل‌های تولیدی ماسک شده نیز به کار رفته‌اند. همچنین در مدل‌های انتشار نویزدایی (DDPM) برای تولید داده‌های غیرمکانی مانند تولید جاسازی‌های تصویر CLIP در DALL-E ۲ استفاده شده‌اند. در این مقاله، خواص مقياس‌پذیری ترانسفورمرها را هنگامی که به عنوان ستون فقرات مدل‌های انتشار تصویر استفاده می‌شوند، بررسی می‌کنیم.

مدل‌های انتشار نویزدایی احتمالاتی (DDPMs) در تولید تصاویر موفق بوده‌اند و در بسیاری از موارد از شبکه‌های مولد تقابلی (GANs) پیشی گرفته‌اند. بهبودها در مدل DDPM عمدهاً توسط تکنیک‌های نمونه‌گیری بهبود یافته، هدایت بدون استفاده از طبقه‌بند، بازفرمول‌بندی مدل‌های انتشار برای پیش‌بینی نویز به جای پیکسل‌ها و استفاده از پایپ‌لاین‌های DDPM آبشاری با مدل‌های انتشار پایه با وضوح پایین و نمونه‌بردارهای افزایش‌دهنده انجام شده است. در تمام مدل‌های انتشار ذکر شده، های U-Net کانولوشنی به عنوان معماری ستون فقرات به کار رفته‌اند. کار همزمان نیز یک معماری جدید و کارآمد مبتنی بر توجه برای مدل DDPM معرفی کرده است. در این مقاله، ما به بررسی ترانسفورمرهای خالص می‌پردازیم.

هنگام ارزیابی پیچیدگی معماری در ادبیات تولید تصویر، استفاده از تعداد پارامترها یک عمل رایج است. با این حال، تعداد پارامترها می‌تواند نشانگر ضعیفی برای پیچیدگی مدل‌های تصویر باشد، زیرا به عنوان مثال، وضوح تصویر که به طور قابل توجهی بر عملکرد تأثیر می‌گذارد را در نظر نمی‌گیرد. در عوض، بسیاری از تحلیل‌های پیچیدگی مدل در این مقاله از منظر Gflops نظری است. این روش با ادبیات طراحی معماری که در آن Gflops به طور گستره‌ای برای سنجش پیچیدگی استفاده می‌شود، هم خوانی دارد. در عمل، معیار طلایی

پیچیدگی هنوز مورد بحث است زیرا اغلب به سناریوهای کاربردی خاص بستگی دارد. کارهای اصلی Nichol و Dhariwal در بهبود مدل‌های انتشار بیشتر به ما مرتبط است، جایی که آن‌ها مقیاس‌پذیری و ویژگی‌های U-Net را تحلیل کرده‌اند. در این مقاله، ما بر کلاس ترانسفورمرها تمرکز می‌کنیم.

مدل‌های انتشار به عنوان مدل‌های مولد عمیق قدرتمند اخیراً برای تولید تصاویر با کیفیت بالا معرفی شده‌اند. آن‌ها به سرعت رشد کرده و در تولید متن به تصویر، تصویر به تصویر، تولید ویدئو، سنتز گفتار و سنتز سه‌بعدی کاربرد یافته‌اند. به همراه توسعه الگوریتم‌ها، انقلاب ستون فقرات نقش مرکزی در مدل‌های انتشار دارد. مثالی بر جسته U-Net مبتنی بر شبکه عصبی کانولوشنی (CNN) است که در کارهای پیشین استفاده شده است. U-Net مبتنی بر CNN با گروهی از بلوک‌های نمونه‌برداری پایین، گروهی از بلوک‌های نمونه‌برداری بالا و اتصالات بلند بین دو گروه مشخص می‌شود که در مدل‌های انتشار برای وظایف تولید تصویر غالب است. از سوی دیگر، ویژن ترانسفورمرها (ViT) در وظایف مختلف بینایی نویدبخش بوده‌اند، جایی که ViT در مقایسه با رویکردهای مبتنی بر CNN قابل مقایسه یا حتی برتر بوده‌اند. بنابراین، یک سؤال طبیعی مطرح می‌شود: آیا وابستگی به U-Net مبتنی بر CNN در مدل‌های انتشار ضروری است؟

در این مقاله، ما یک معماری ساده و عمومی مبتنی بر ViT به نام U-ViT طراحی کرده‌ایم. U-ViT تمام ورودی‌ها از جمله زمان، شرط و پچ‌های تصویر نویزی را به عنوان توکن‌ها در نظر می‌گیرد. به طور حیاتی، U-ViT اتصالات بلند بین لایه‌های سطحی و عمیق را الهام‌گرفته از U-Net به کار می‌گیرد. به طور شهودی، ویژگی‌های سطح پایین برای هدف پیش‌بینی در سطح پیکسل در مدل‌های انتشار مهم هستند و این اتصالات می‌توانند آموزش شبکه پیش‌بینی مربوطه را آسان‌تر کنند. علاوه بر این، U-ViT به طور انتخابی یک بلوک کانولوشن 3×3 اضافی قبل از خروجی برای کیفیت بصری بهتر اضافه می‌کند.

ما U-ViT را در سه وظیفه محبوب ارزیابی می‌کنیم: تولید تصویر بدون شرط، تولید تصویر شرطی بر اساس کلاس و تولید متن به تصویر. در همه تنظیمات، U-ViT قابل مقایسه با U-Net مبتنی بر CNN با اندازه مشابه است و در برخی موارد برتر نیز می‌باشد. به ویژه، مدل‌های انتشار نهان با U-ViT به نمرات FID رکورددشکن ۲۹.۲ در تولید تصویر شرطی بر اساس کلاس در 256×256 ImageNet و ۴۸.۵ در تولید متن به تصویر در MS-COCO دست یافته‌اند. نتایج ما نشان می‌دهد که اتصالات بلند مهم هستند در حالی که عملگرهای نمونه‌برداری پایین/بالا در U-Net مبتنی بر CNN همیشه برای مدل‌های انتشار تصویر ضروری نیستند. ما باور داریم که U-ViT می‌تواند برای تحقیقات آینده در مورد ستون فقرات مدل‌های انتشار بینشی ارائه دهد و به مدل‌سازی مولد در مجموعه‌های داده‌های بزرگ و چندوجهی کمک کند.

فصل ۴

مقایسه مقاله های [۱، ۲] و نتیجه گیری

هر دو مقاله بررسی شده، به ارائه ساختاری متن باز جهت انجام دو فرایند مهم در طراحی به نام های چینش و مسیریابی با استفاده از تکنیک های یادگیری عمیق پرداخته است. با این تفاوت که در مقاله [۲] محدودیتی بر روی مدار یا ساختار مورد بخث گذاشته نشده است و می‌توان الگوریتم را برای هر مدار فشرده و خیلی فشرده «VLSI» ای آموزش داد و از آن در فرایند طراحی استفاده نمود. اما در مقاله [۱] الگوریتم ارائه شده فقط مخصوص چینش و مسیریابی FPGA هایی با منابع داخلی فراوان است. از این رو، کاربرد و عمومیت مقاله [۲] نسبت به مقاله [۱] بیشتر است.

هر دو مقاله برای انجام فرایند آموزش از سخت‌افزار های پیشرفته و گران قیمتی استفاده کرده‌اند.

در [۲] از سروری ۴۰ هسته لینوکسی با سی‌پی‌یو Intel E5-۲۶۹۸ با فرکانس کاری ۲/۲ گیگاهرتز و کارت‌گرافیک NVIDIA Tesla V۱۰۰ برای فرایند آموزش استفاده شده است.

و در [۱] از سروری لینوکس بیس که CPU آن Intel Xeon ۶۲۳۰ با فرکانس کاری ۱۰/۲ گیگاهرتز و ۴۰ هسته هسته، با کارت‌گرافیک NVIDIA RTX ۲۰۸۰ Ti و حافظه رم ۵۱۲ گیگابایتی استفاده شده است.

در هر دو مقاله سیستم‌های استفاده شده، سیستم‌های قوی و خاص منظوره ای هستند که کمتر در دسترس عموم مردم است. به همین دلیل ممکن است نتوان این الگوریتم‌ها را با سیستم‌های معمولی توسعه داد و این شاید به نوعی یکی از عیوب‌های این دو الگوریتم به حساب آید.

در [۲] الگوریتم با دو بنچ‌کارک اکادمیک و صنعتی تست شده است که خروجی‌های آن به صورت زیر است:
«شکل ۱-۴»

در [۱] نیز خروجی به ازای بنچ‌کارک ISPD ۲۰۱۷ ۲۰۴ مشاهده شده است. «شکل ۲-۴» همانطور که در «شکل ۱-۴» مشاهده می‌شود، در الگوریتم DREAMPlace با افزایش تعداد سلول‌های

Table 2: Experimental results on ISPD 2005 benchmarks [25].

Design	#cells	#nets	RePIPlace [6] (40 Threads)					DREAMPlace					
			HPWL	GP (s)	LG (s)	DP (s)	Total (s)	HPWL	GP (s)	LG (s)	DP (s)	IO (s)	Total (s)
adaptec1	211K	221K	73.26	106	5	24	139	73.30	5	0.5	25	5	37
adaptec2	255K	266K	81.87	194	7	30	236	82.19	6	0.5	32	6	47
adaptec3	452K	467K	193.20	382	22	53	466	194.12	10	1	58	10	82
adaptec4	496K	516K	175.23	434	21	61	524	174.43	11	2	65	11	92
bigblue1	278K	284K	89.85	168	3	30	206	89.43	7	0.3	35	6	51
bigblue2	558K	577K	138.09	383	22	90	505	136.69	11	9	90	12	125
bigblue3	1097K	1123K	304.83	967	46	138	1171	303.99	22	3	145	24	198
bigblue4	2177K	2230K	743.73	2037	56	329	2463	743.75	43	9	336	54	446
ratio	-	-	1.002	34.8	10.6	0.9	5.0	1.000	1.0	1.0	1.0	-	1.0

Table 3: Experimental results on industrial benchmarks.

Design	#cells	#nets	RePIPlace [6] (40 Threads)					DREAMPlace					
			HPWL	GP (s)	LG (s)	DP (s)	Total (s)	HPWL	GP (s)	LG (s)	DP (s)	IO (s)	Total (s)
design1	1345K	1389K	340.42	974	46	155	1216	340.87	24	4	179	34	244
design2	1306K	1355K	275.46	1001	44	152	1238	275.76	24	5	180	32	244
design3	2265K	2276K	524.35	1767	84	257	2189	522.79	37	14	305	55	414
design4	1525K	1528K	455.22	1130	55	187	1424	454.38	26	8	207	37	281
design5	1316K	1364K	287.24	955	46	153	1193	288.41	22	4	186	33	248
design6	10504K	10747K	NA	>9100	NA	NA	NA	2356.88	282	73	1681	276	2323
ratio	-	-	1.000	43.1	9.5	0.9	5.0	1.000	1.0	1.0	1.0	-	1.0

شکل ۱-۴: خروجی های زمانی الگوریتم DREAMPlace برای دو بنچمارک مختلف

Design	#LUT/#FF/#BRAM/#DSP	#Clock	RippleFPGA [13]			OpenPARF		
			PRT	RRT	RWL	PRT	RRT	RWL
CLK-FPGA01	211K/324K/164/75	32	278	10	238.54	131	10	205.44
CLK-FPGA02	230K/280K/236/112	35	250	15	261.85	127	14	246.65
CLK-FPGA03	410K/481K/850/395	57	537	24	648.69	206	24	594.00
CLK-FPGA04	309K/372K/467/224	44	346	18	440.09	157	19	420.30
CLK-FPGA05	393K/469K/798/150	56	501	25	560.18	201	23	510.62
CLK-FPGA06	425K/511K/872/420	58	545	28	678.43	218	28	617.28
CLK-FPGA07	254K/309K/313/149	38	288	13	276.29	136	13	256.62
CLK-FPGA08	212K/257K/161/75	32	235	10	213.06	119	10	196.67
CLK-FPGA09	231K/358K/236/112	35	312	13	297.02	148	14	250.98
CLK-FPGA10	327K/506K/542/255	47	465	23	544.07	195	14	451.28
CLK-FPGA11	300K/468K/454/224	44	421	24	516.67	182	30	421.52
CLK-FPGA12	277K/430K/389/187	41	378	18	403.59	167	20	336.03
CLK-FPGA13	339K/405K/570/262	47	393	21	464.78	178	19	428.41
Ratio			2.251	1.037	1.128	1.000	1.000	1.000

شکل ۲-۴: خروجی های زمانی الگوریتم OpenPARF برای بنچمارک ISPD ۲۰۱۷

طراحی، زمان مراحل GP و LG و ... افزایش پیدا می‌کند و سلوول ها تا ۲ میلیون و ۱۷۷ هزار عدد افزایش داده شده است که برای این مقدار سلوول، مرحله GP ، ۴۳ ثانیه طول کشیده است.

برای الگوریتم OpenPARF هم هرچقدر دیزاین ها پیچیده تر می‌شود، هم تعداد منابع مصرفی داخلی FPGA بیشتر مصرف می‌شود و هم زمانبندی فاز چینش و مسیریابی نیز زیاد تر می‌شود.

نتیجه گیری ای که این جانب از این مطالعات دارم بدین صورت است:

با پیشرفت ابزارهای هوش مصنوعی، کم کم ابزارهای قدیمی طراحی دارند جای خود را به ابزارهای هوشمند می‌دهند. هرچند که همچنان راه زیادی وجود دارد تا جایگذینی کامل این ابزارها با هم اما بلاخره روزی فراخواهد رسید که قرایнд طراحی از مرحله طراحی شماتیک تا Layout به صورت اتوماتیک و بدون دخالت انسان انجام

می شود.

این دو مقاله به پیاده سازی ۲ مرحله مهم از طراحی، یعنی چینش قطعات و مسیر یابی آنها گام کوچ و موثری در تحقق این هدف برداشته اند.

امید است که بتوانیم با تحقیقات بیشتر و ارائه ساختارهای جدید به پیشرفت تکنولوژی در این مسیر کمک کنیم.

Bibliography

- [1] J. Mai, J. Wang, Z. Di, G. Luo, Y. Liang, and Y. Lin. Openparf: An open-source placement and routing framework for large-scale heterogeneous fpgas with deep learning toolkit. *IEEE 15th International Conference on ASIC (ASICON)*, 2023.
- [2] Y. Lin, S. Dhar, W. Li, H. Ren, B. Khailany, and D. Z. Pan. Dreamplace: Deep learning toolkit-enabled gpu acceleration for modern vlsi placement. *Proceedings of the 56th Annual Design Automation Conference*, (117):1 – 6, 2019.
- [3] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [4] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.
- [5] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.
- [6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- [7] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.
- [8] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

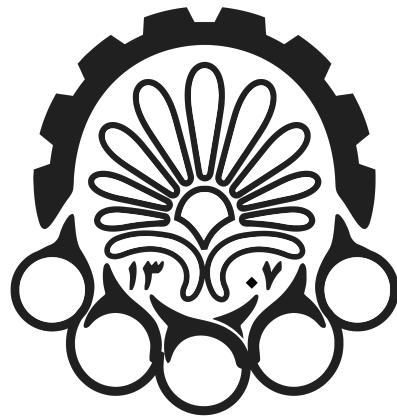
- [9] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [10] K. E. Murray, S. Whitty, S. Liu, J. Luu, , and V. Betz. Timing-driven titan: Enabling large benchmarks and exploring the gap between academic and commercial cad. *ACM TRETS*, pages 1 – 18, 2015.

Abstract

The main challenge of this research has been to examine and compare the efficiency of transformers against traditional architectures like U-Net in diffusion models. The obtained results indicate that transformers, with better scalability and higher efficiency, can be a suitable replacement for current architectures in diffusion models and help improve the quality of generated images. In this regard, Diffusion Transformers (DiTs), by increasing the depth and width of transformers and increasing the number of input tokens, have shown significant improvement in FID scores and achieved superior results in the ImageNet benchmark sets.

This research demonstrates that using transformers in diffusion models can provide innovative solutions for improving the quality and efficiency of image generation. These approaches, in addition to enhancing performance, can help reduce computational complexities and increase the speed of image generation processes. Given these results, transformers emerge as a strong alternative to traditional architectures in diffusion models and can be widely used in various applications.

Keywords: Deep Learning, Diffusion Models, Transformer, U-Net, Encoder, Decoder



Amirkabir University of Technology

(Tehran Polytechnic)

Department of Computer Engineering

Deep Learning Final Research Report Thesis

Diffusion Models with Transformers Neural Network

By:

Reza Adinepour

Supervisor:

Prof. Safabakhsh

Jun 2024