

Generatywna Sztuczna Inteligencja

Uczenie maszynowe II



Włodzisław Duch

Katedra Informatyki Stosowanej UMK

Google: Włodzisław Duch

Model Isinga

100 lat modelu Isinga: nadal ≈ 800 publikacji co roku! Praca doktorska z 1924 roku.

Mechanika statystyczna, model ferromagnetyzmu, oddziaływujące dipole magnetyczne (spiny) $\sigma_i = \pm 1$. W 1944 Lars Onsager (Nobel z chemii, 1968) podał rozwiązanie w 2D, w pracy o statystyce kryształów, zmiany od fazy nieuporządkowanej do uporządkowanej.

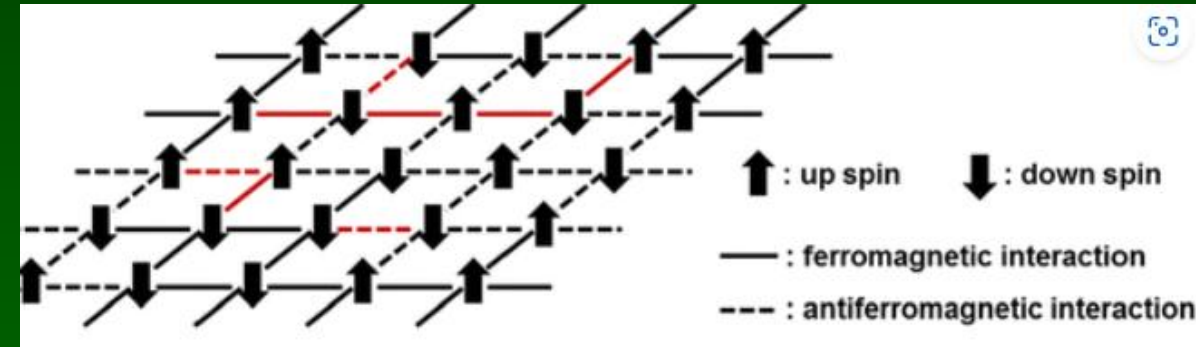
Wielu słynnych fizyków próbowało rozwiązać model 3D analitycznie ale się to nie udało.

S.F. Edwards, P.W. Anderson (1975) *Theory of spin glasses*.

Spin glass random Heisenberg Hamiltonian:

$$H(\sigma) = - \sum_{ij} J_{ij} \sigma_i \sigma_j$$

$J_{ij} > 0$ or $J_{ij} < 0$, $\sigma_i = \pm 1$



Giorgio Parisi, 2021 Nobel z fizyki, za badania fluktuacji i uporządkowania w sieciach oddziaływujących elementów. Zastosowania w sociofizyce, ekonofizyce, ewolucji języka, modelowaniu raka, rynkach finansowych, głosowaniu w wyborach, polaryzacji przekonań politycznych, wahaniach nastroju ...

Macy, M.W., Szymanski, B.K., & Hołyst, J.A. (2024). The Ising model celebrates a century of interdisciplinary contributions. Npj Complexity 1, 1–8

Więcej w referatach o Nagrodzie Nobla 2025 <https://is.umk.pl/~duch/ref.html>

Sieci Hopfielda

J.J. Hopfield (1982). *Neural networks and physical systems with emergent collective computational abilities*, PNAS 79(8), 2554

Standardowy model oddziałujących na siebie elementów studiowany w ramach fizyki statystycznej, dający się zaimplementować sprzętowo.

Sieć Hopfielda: binarne element (spiny)

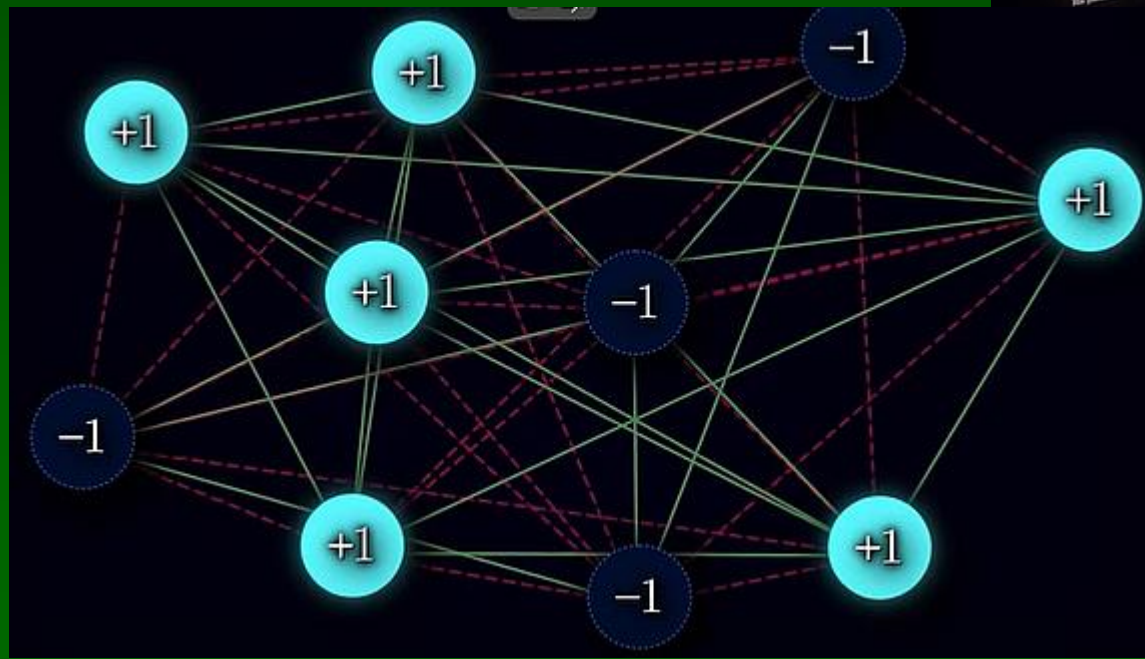
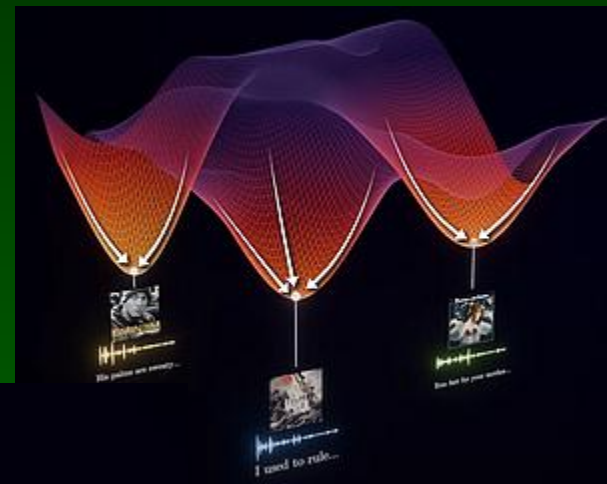
W pełni połączone,
symetryczne połączenia:

$$w_{ij} = w_{ji}$$

$$\min E = - \sum_{ij} w_{ij} x_i x_j$$

Uczenie: adaptacja w_{ij} tworzy lokalne minima.

Wnioskowanie: znajdź wzorce rozkładów x_i minimalizujące energię.



Pamięci asocjacyjne

Aby osiągnąć minimum, zaktualizuj wartości x_i sumując wszystkie wejścia docierające do tego węzła, co zmniejszy energię, iteracje doprowadzą do wzorca $\{x_i\}$ odpowiadającego lokalnemu minimum, w zależności od początkowego wzorca.

Prowadzi to do uzupełnienia wzorca, oczyszczenia z szumów i optymalizacji.

Uczenie: dla pojedynczego globalnego zestawu minimalnego $w_{ij} = x_i x_j$, więc dla N neuronów:

$$\min E = - \sum_{ij} (x_i x_j)^2 = - N^2$$

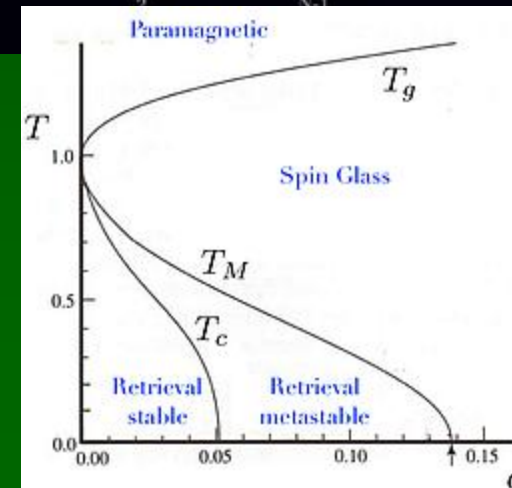
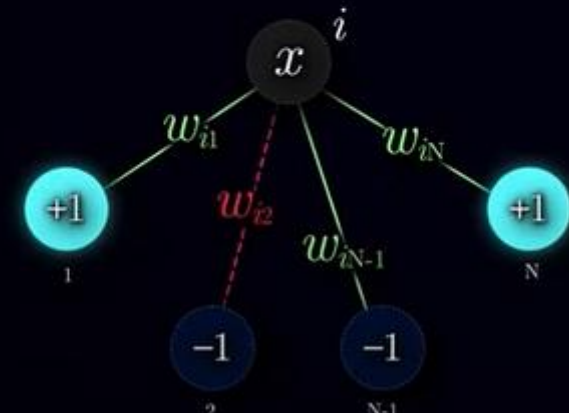
Dla kilku wzorców: zsumuj wszystkie wagi dla każdego

$$w_{ij}(k) = x_i(k) x_j(k)$$

Reguła Hebba: neurony, które pracują razem, łączą się razem, uzasadnia uśrednianie. Pojemność pamięci wynosiła: $N_p \approx 0.14 N$, ale teraz jest znacznie zwiększona. Użyj symulowanego wyżarzania do zbadania krajobrazu energetycznego $\Rightarrow$ 4 fazy na diagramie, w zależności od $\alpha = N_p/N$: stabilne, płytkie lokalne, kilka szerokich minimów (szkło spinowe) i jedno globalne minimum (paramagnetyczne).

[Good intro by Artem Kirsanov](#) (YouTube)

$$h_i = \sum_{j \neq i} w_{ij} x_j$$
$$x_i = \begin{cases} +1 & \text{if } h_i > 0 \\ -1 & \text{if } h_i < 0 \end{cases}$$



Maszyny Boltzmann

Spełnianie ograniczeń pozwala na budowę modeli generatywnych na wyższym poziomie abstrakcji.

- Ackley, D.H., Hinton, G.E., & Sejnowski, T.J. (1985). *A learning algorithm for Boltzmann machines*. Cognitive Science, 9(1), 147–169.

Dodaj losowość do obliczeń, znajdź rozkłady prawd. które uogólniają wzorce pamięci i tworzą wewnętrzne reprezentacje danych.

Sieć Hopfielda+jednostki ukryte+eksploracja stochastyczna.

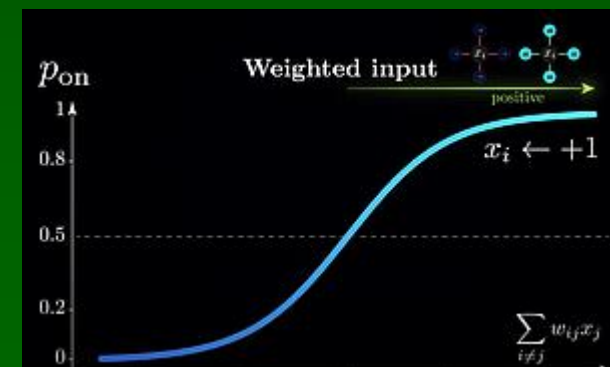
Prawd. przejścia między dwoma stanami energetycznymi:

$$P(\Delta E) = e^{-\Delta E/T} \quad P(\text{State}_E) = \frac{1}{Z} e^{-E/T}$$

Normalizacja funkcji partycji $Z = \sum_S e^{-E_S/T}$

Zaktualizuj stan neuronu x_i obliczając energię $E_i = -\sum_{i \neq j} w_{ij} x_i x_j$ z prawd. opartym na różnicy energii dla $x_i = \pm 1$ określonej przez funkcję sigmoidalną (funkcja transferu neuronu):

$$P(x = +1) = \frac{e^{-E_{i+}/T}}{e^{-E_{i+}/T} + e^{-E_{i-}/T}} = \frac{1}{1 + e^{-\Delta E_i/T}}$$



Generatory rozkład=>obraz

Oblicz sygnał dla neuronu x_i od połączonych neuronów $\sum_{j \neq i} w_{ij} x_j$

Oblicz prawd. aktualizacji z użyciem f. sigmoidalnej $P(x_i = +1) = \frac{1}{1 + e^{-\Delta E_i/T}}$



Wygeneruj liczbę losową: jeśli jest mniejsza niż $P(x_i)$ to brak zmiany, jeśli większa to zmiana.

Niskie T = bardziej deterministyczne, wysokie T bardziej losowa eksploracja lokalnych minimów.

Używane również w modelach LLM do kontrolowania kreatywności sieci.

Uczenie kontrastywne: zmiana prawdopodobieństwa jednego wzorca wpłynie na funkcję partycji.

Maksymalizacja łącznego prawdopodobieństwa danych treningowych $\{x^{(1)}, x^{(2)}, \dots, x^{(n)}\}$

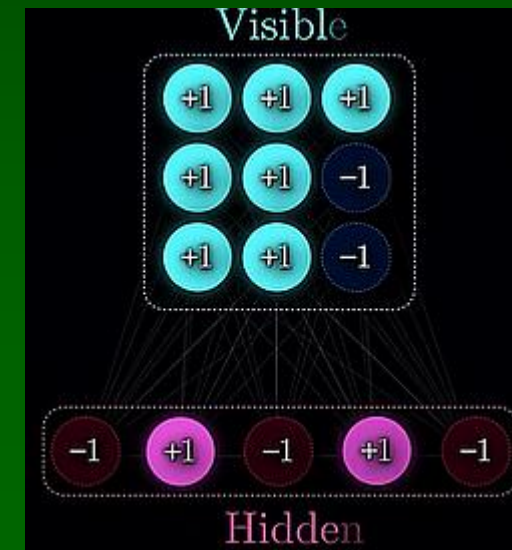
$$\log P(\text{Data}) = -\frac{1}{T} \sum_{i=1..n} \log P(x^{(i)}) - n \log Z$$

Minimalizacja funkcji partycji zwiększy energię fałszywych stanów.

Obliczanie gradientów prowadzi do kontrastowej reguły Hebbowskiej:

$$\Delta w_{ij} \propto \langle x_i x_j \rangle_{\text{data}} - \langle x_i x_j \rangle_{\text{free}}$$

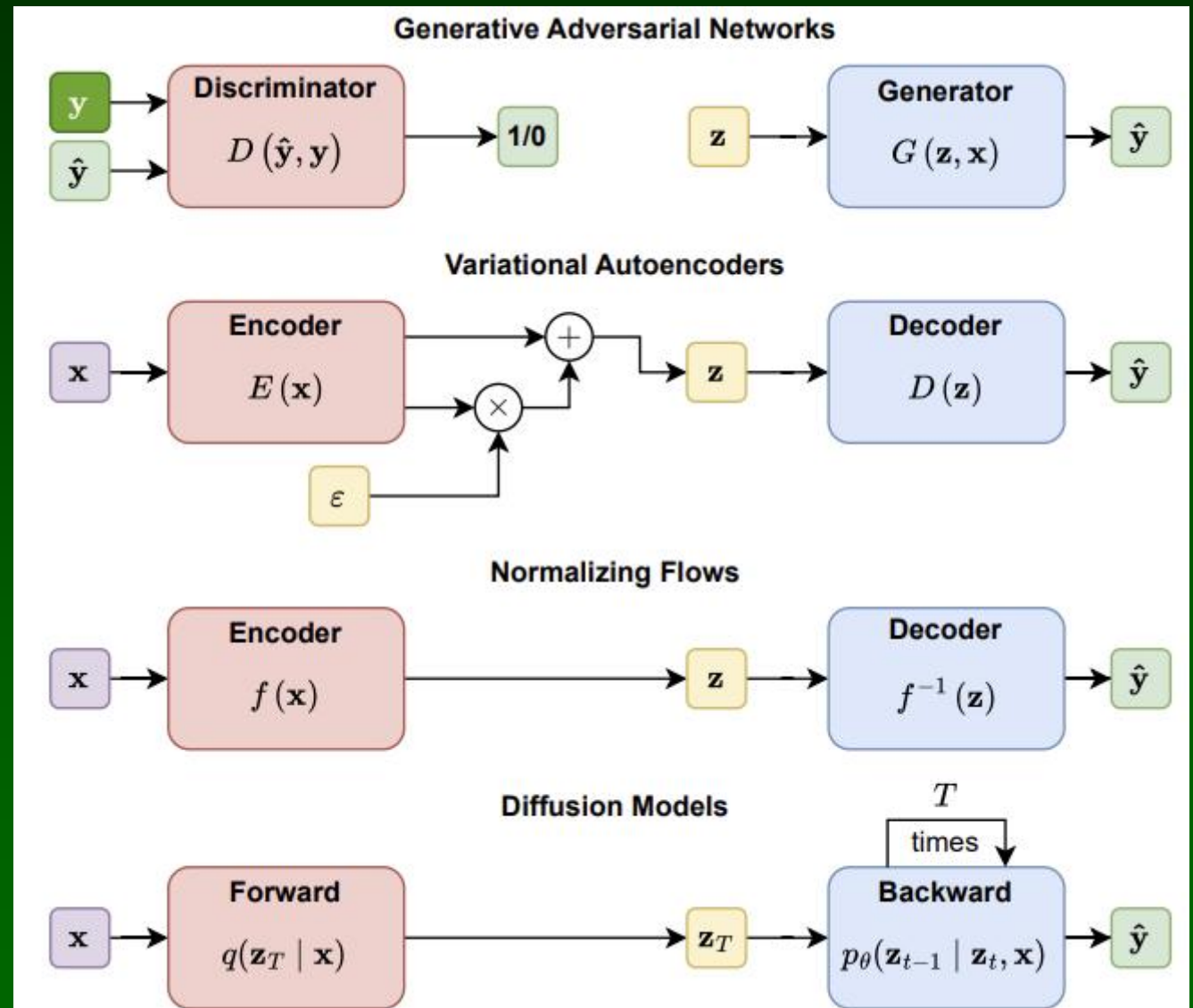
Ustal wejścia, uruchom sieć, wagi zakodują wzorce, druga część redukuje szum, zacznij od losowych wejść. RBM, Restricted Boltzmann machines, mają tylko połączenia między jednostkami ukrytymi na wejściu.



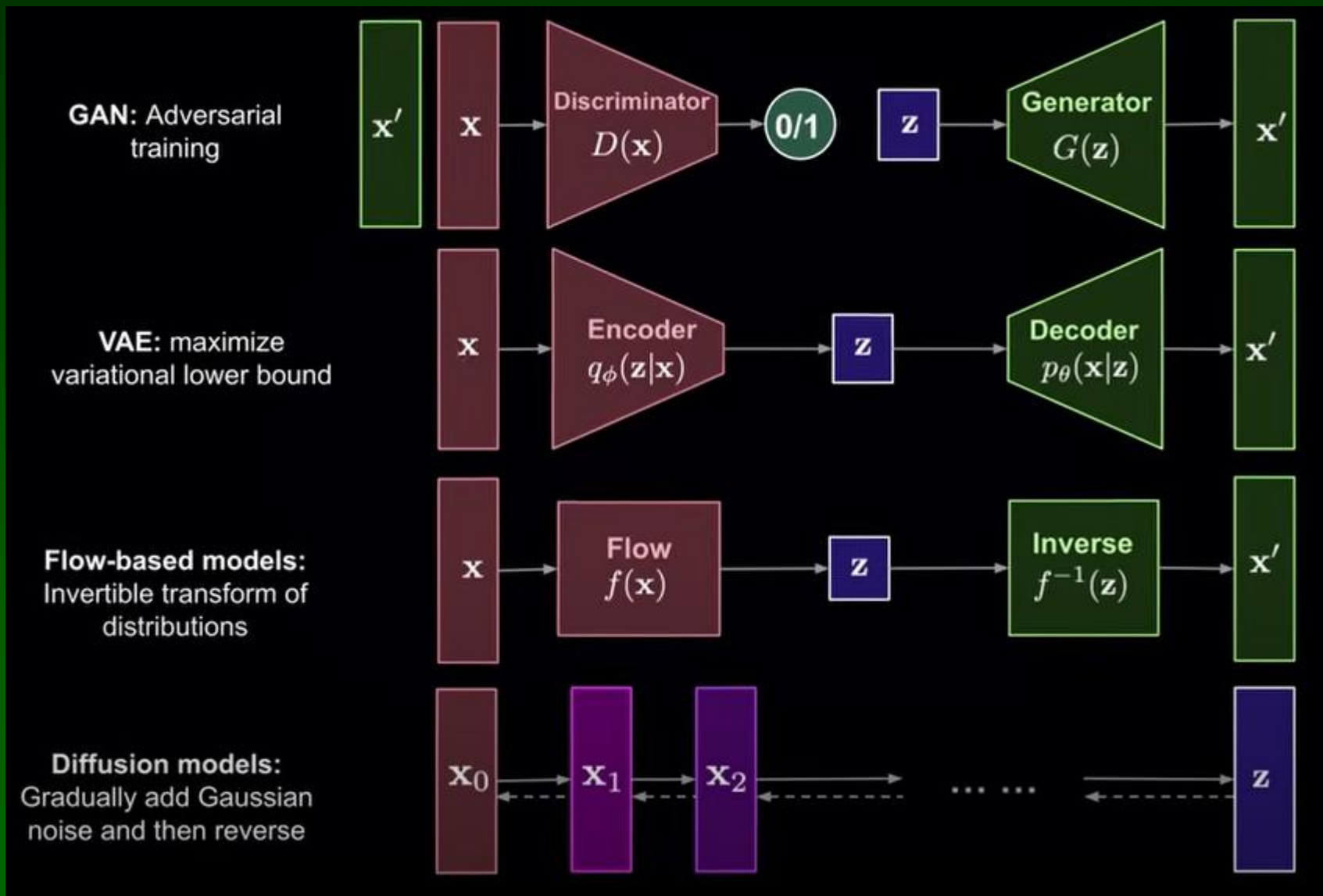
Modele generatywne

Główne kategorie modeli generatywnych.

GAN (Goodfellow 2014):
model generatywny G przechwytuje rozkład danych, oraz model dyskryminacyjny D, który szacuje prawdopodobieństwo, że próbka pochodzi z danych treningowych, a nie z G.
Model D: klasyfikator, sieć neuronowa.
Model G: próbki z rozkładu prawd. Z na przestrzeń obrazów.
o Maszyny Boltzmann i warianty.
o Deep Belief Networks.



Modele generatywne



Kompresja do ukrytych reprezentacji, $x \Rightarrow z$

[GAN](#) (2014), [VAE](#) (2022),

[Flows](#) (2019), [diffusion models](#) (2015).

[EBM, Energy-based models](#).

Rozkłady, a nie obrazy czy zapamiętane teksty. Polecenia (prompty) pozwalają na ułożenie skojarzonych z nimi fragmentów w nową sensowną całość.

Kluczem jest informacja algorytmiczna o niskiej złożoności.

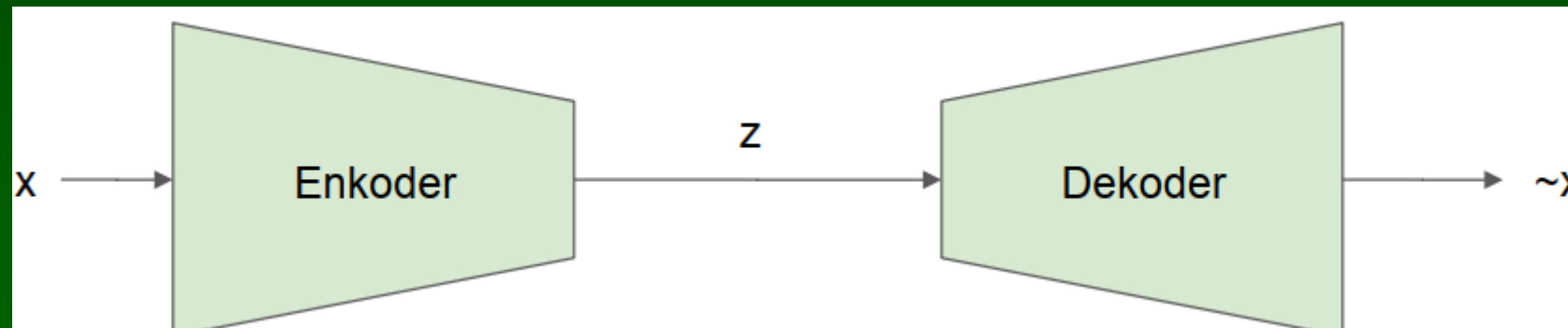
Autoenkodery

Autoasocjacja w modelu Hopfielda/Boltzmann: uszkodzone wzorce zbiegają się do nauczonych.



Ogólna architektura autoenkoderów (AE):

Transformacja $x \Rightarrow z$ i dekompresja $z \Rightarrow x$

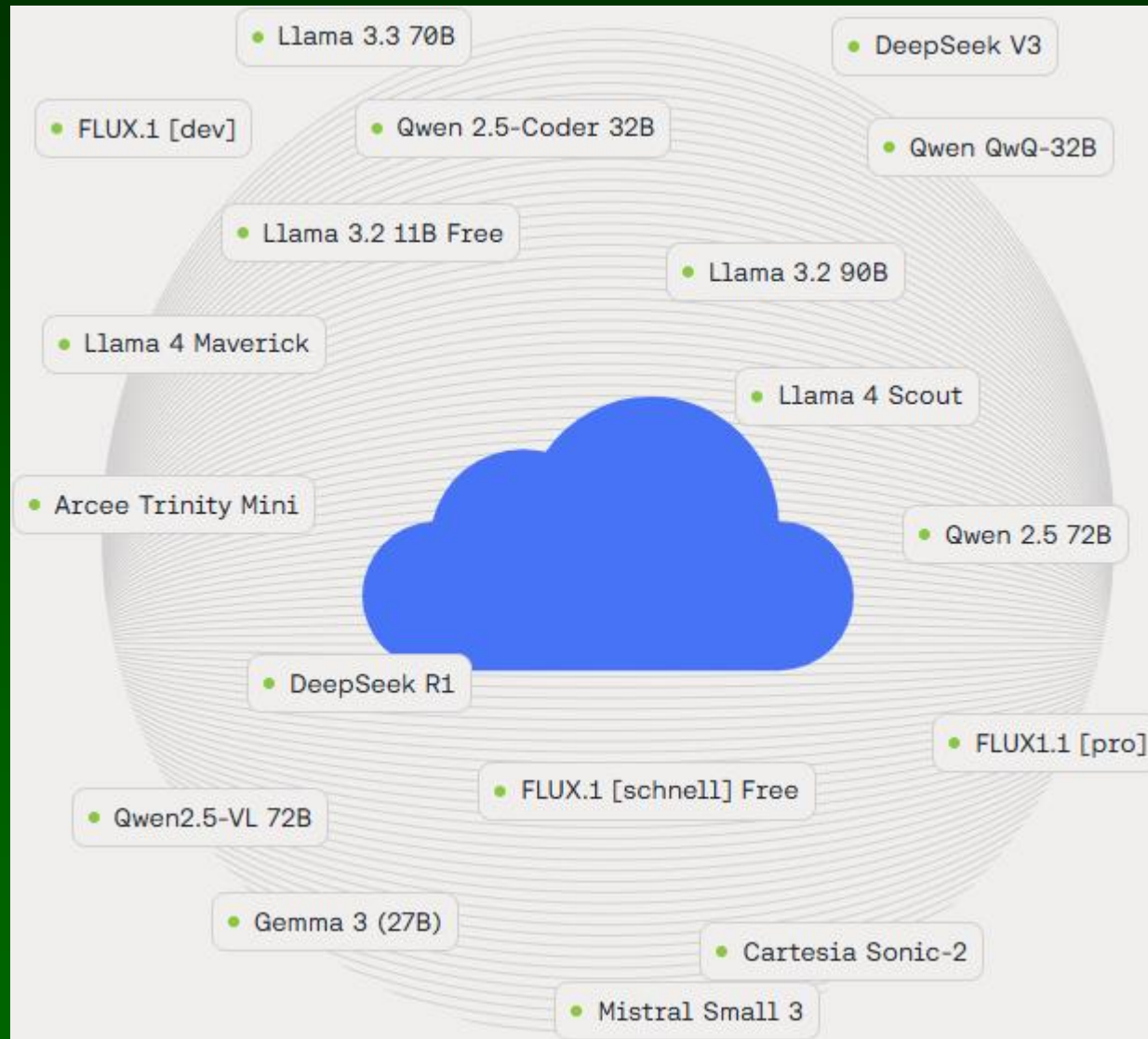


Autoenkoder działa jako generator tworząc ukryte uproszczone reprezentacje.

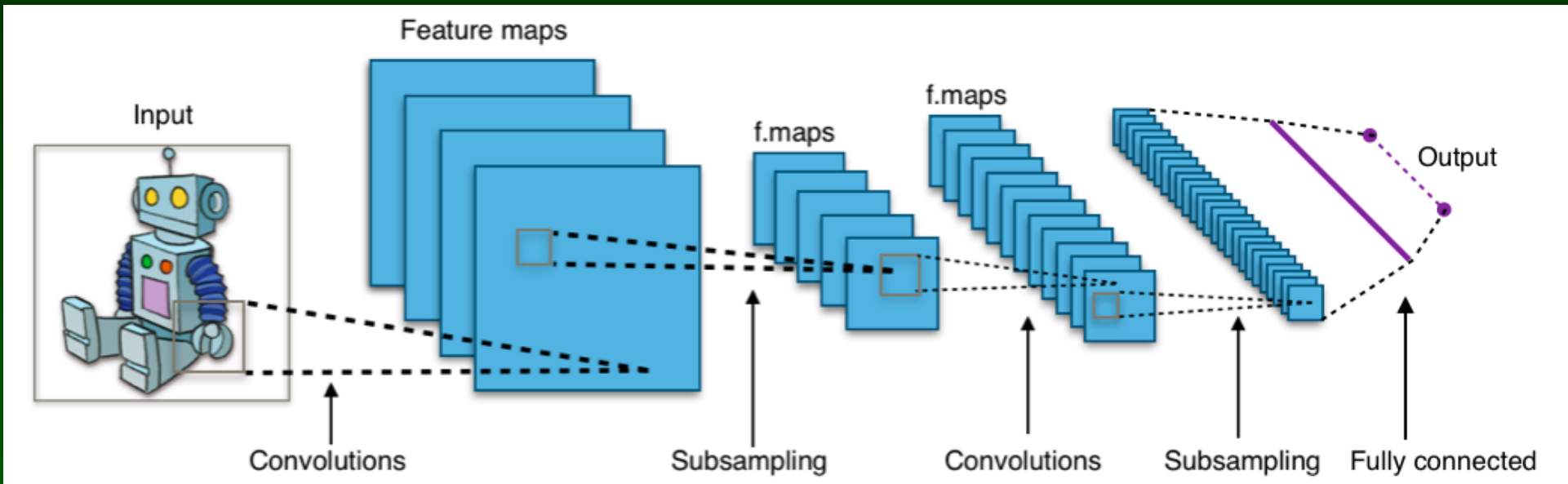
Autoasocjacja to forma uczenia bez nadzoru. [Denoising Autoencoders](#)

Proste wyjaśnienie jak [działają modele dyfuzyjne](#).

Modele open-source

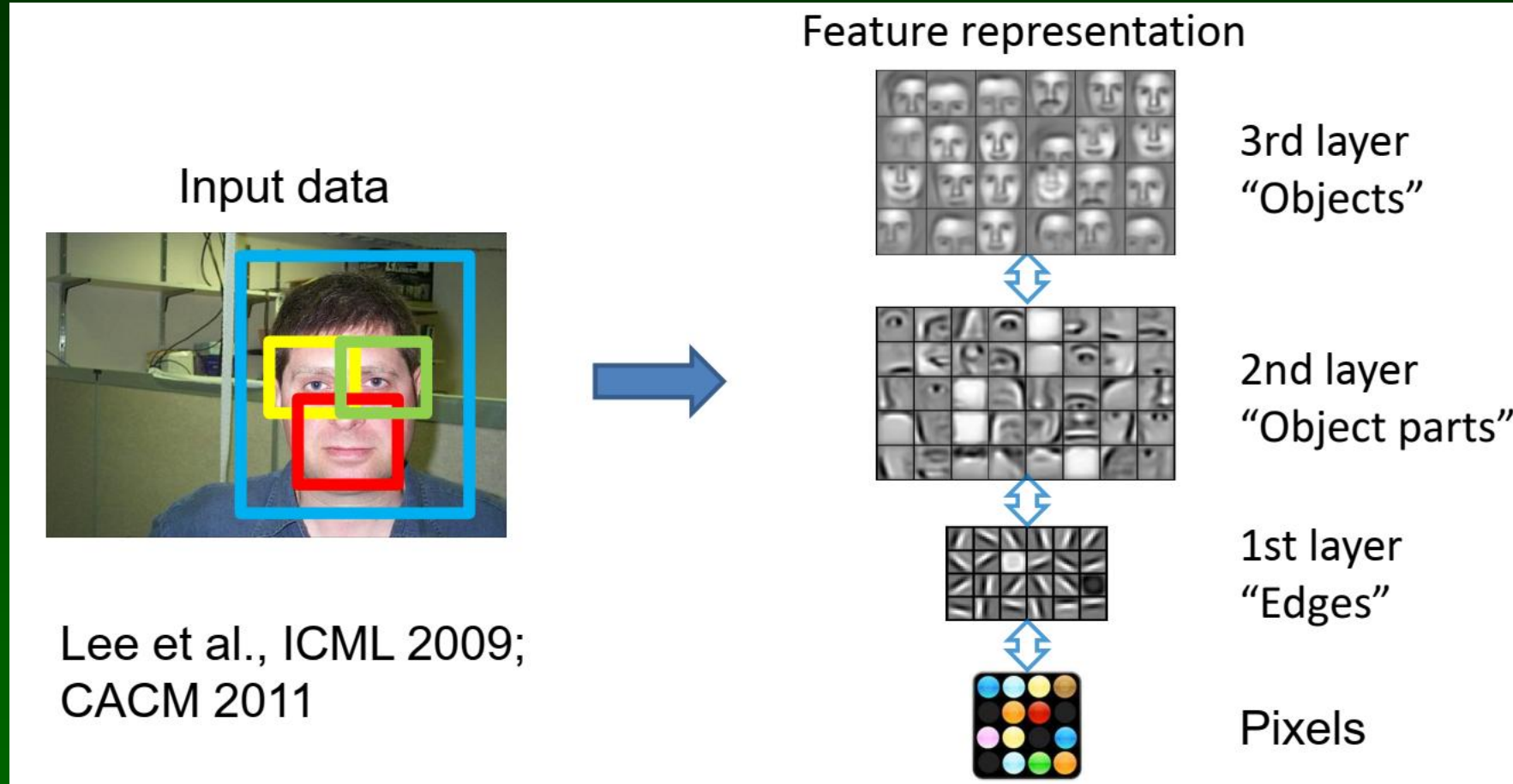


Metody: Głębokie sieci NN dla obrazów



Tokenizacja: małe fragmenty obrazu (np. 5x5 pikseli) są filtrowane przez pola receptywne w operacji splotu, tworząc elementarne cechy, łączone razem za pomocą próbkowania i ponownie filtrowane w celu wygenerowania bardziej złożonych cech. Ostatecznie wygenerowane cechy są analizowane w celu określenia kategorii obrazu.

Methods: NN for images



IBM an [introduction to deep learning](#)

Sieci konwolucyjne CNN

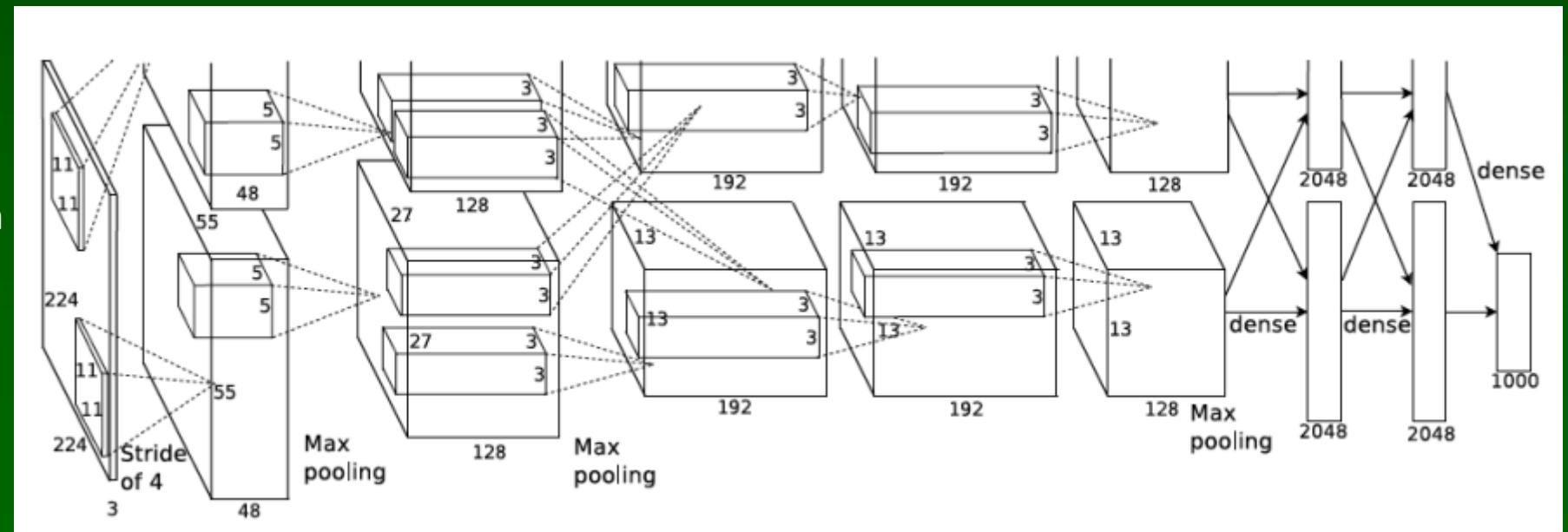
Czemu CNN zawdzięczają swoim możliwości?

- Nieliniowe transformacje pól recepcyjnych.
- Wiele filtrów pracujących równolegle, macierzowe operacje (tensorowe).
- Hierarchiczna (głównie przestrzenna) organizacja, końcowe warstwy są gęste.
- Pomocne: inicjalizacja, algorytmy uczenia (optymalizatory), normalizacja wsadowa (batch normalization) i inne usprawnienia.

Źródło: A. Krizhevsky, I. Sutskever, G.E. Hinton, [ImageNet Classification](#) with Deep Convolutional Neural Networks. NIPS 2012.

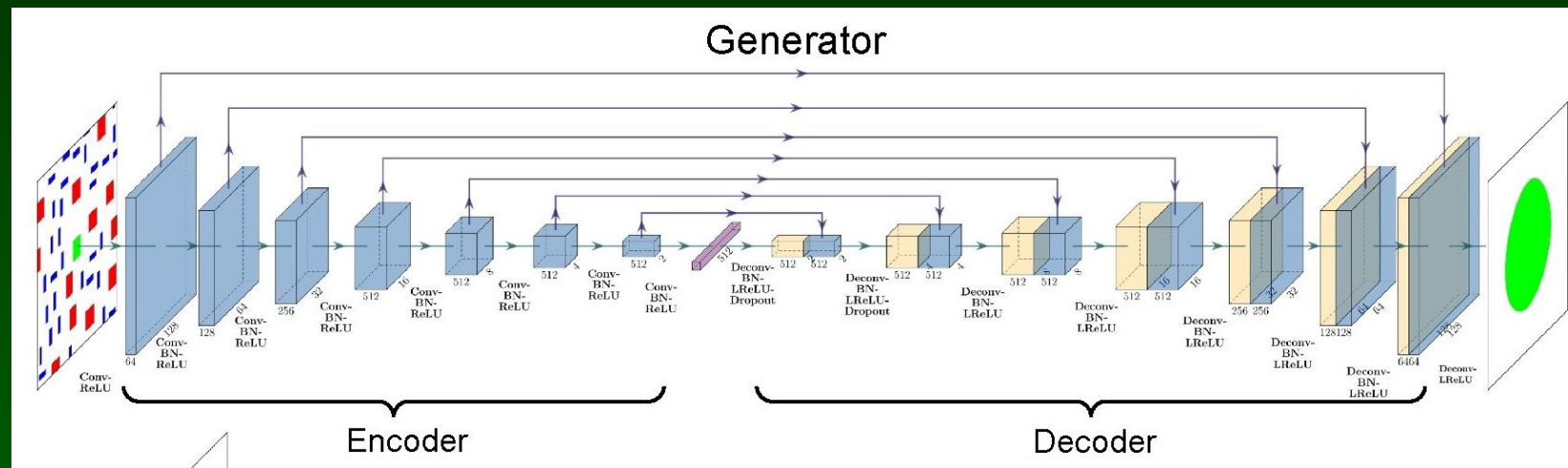
Sieć Alexnet, uczona na 1.2 mln obrazów, 60 mln parametrów, 1000 klas. Jednostki ReLu.

Wiele architektur CNN.



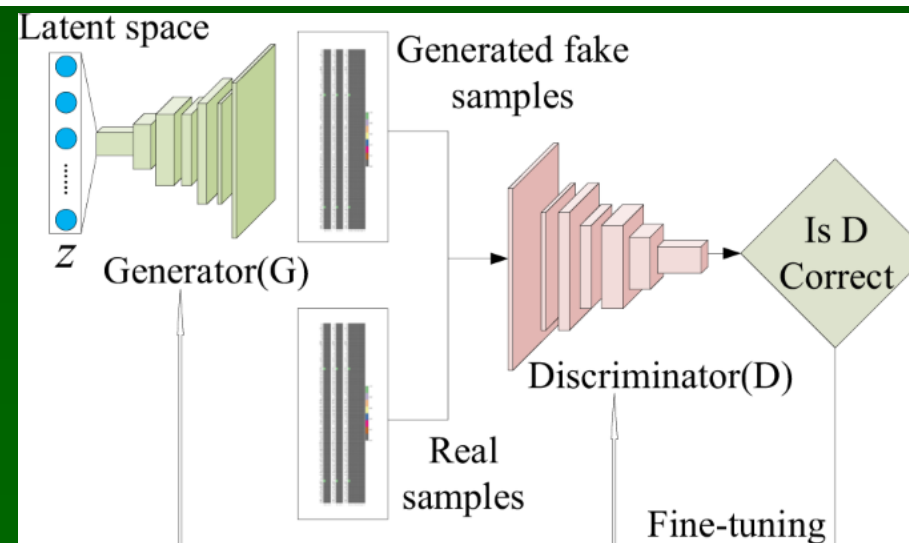
Obrazy i kreatywność

Autoenkodery – kompresja i dekompresja informacji, tworzy przestrzeń ukrytych parametrów.



Generative Adversarial Networks:
Generuj obrazy (G), porównuj z
rzeczywistymi (D), zmieniaj parametry by stawały
się coraz bardziej realistyczne.

[Przykłady na Youtube](#), [Dall-e2](#)



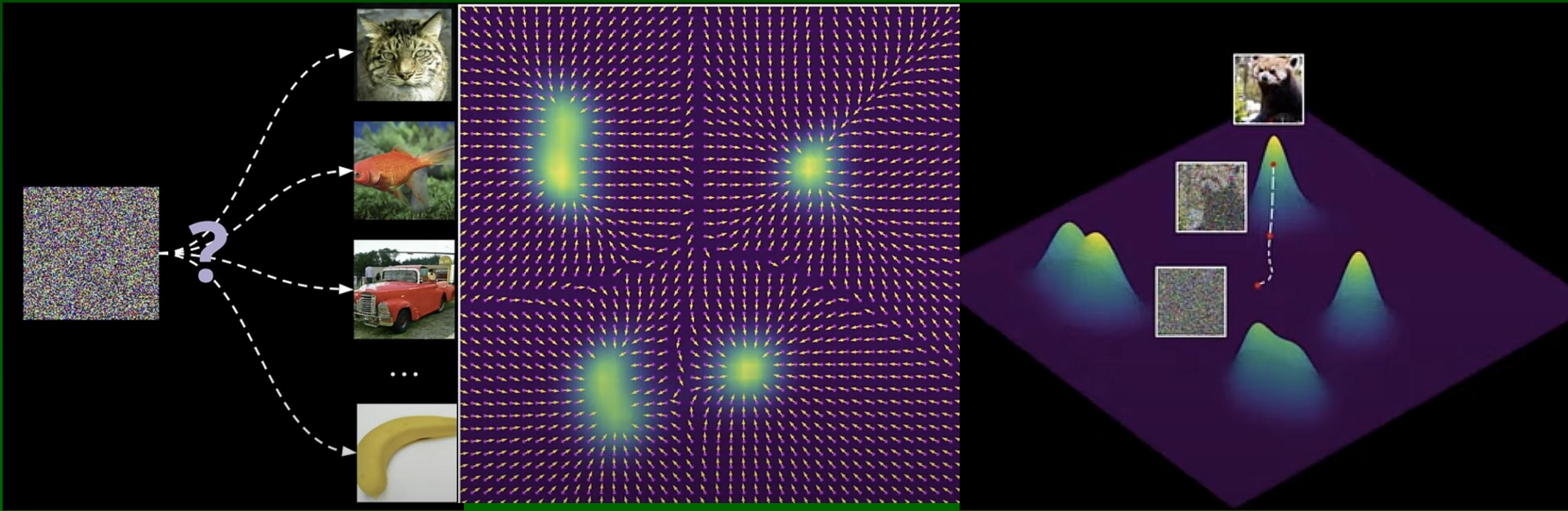
Modele dyfuzyjne

Jak z szumu powstaje obraz? Szum nie zawiera przecież żadnej informacji.

Proste wyjaśnienie jak działają modele dyfuzyjne i bardziej techniczne Denoising Autoencoders.

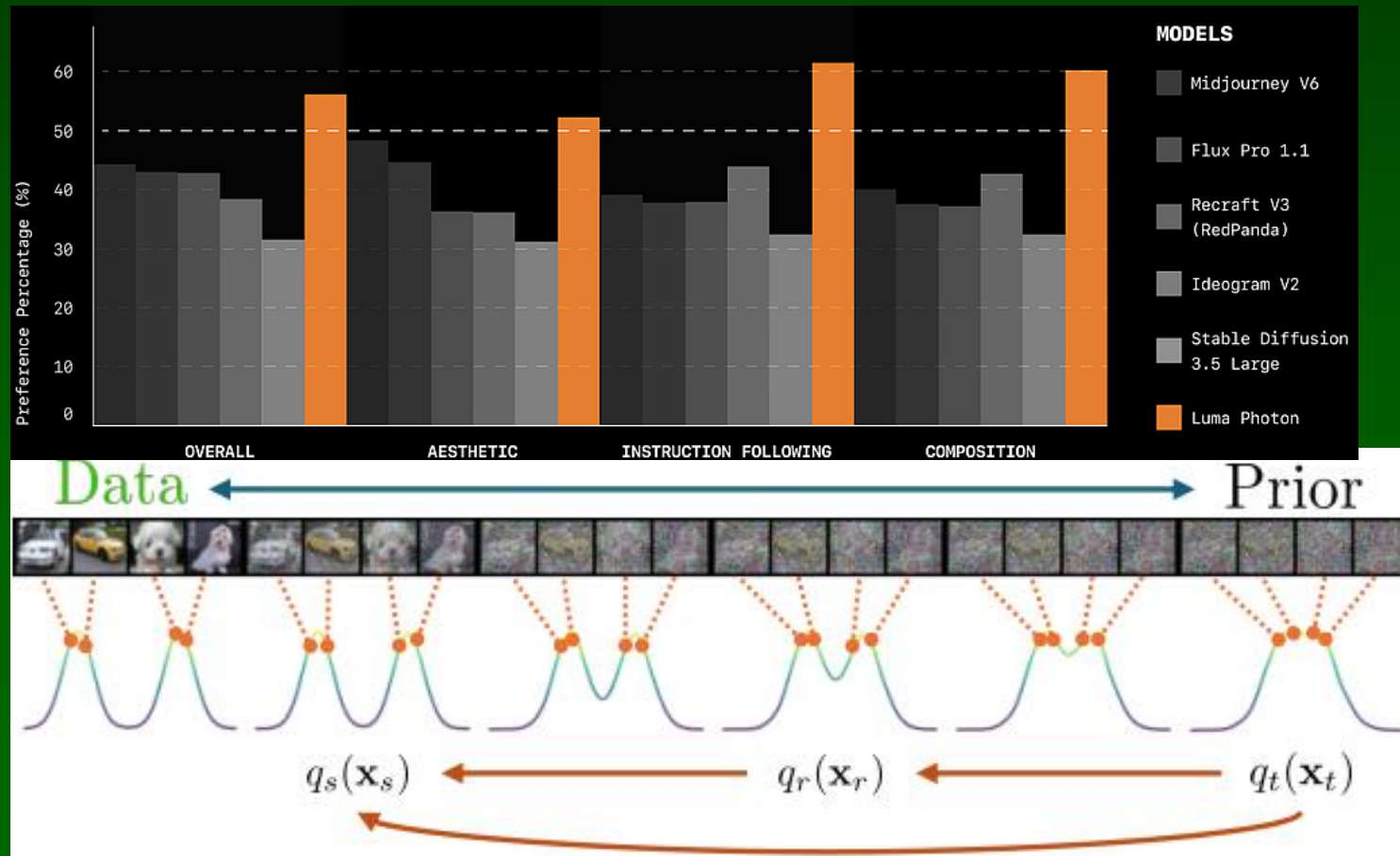
W przestrzeni obrazów 1000X1000 pikseli [0..255] uczymy się kierunku najbliższego klastra z obrazami zgodnymi z opisem, czyli szukamy gradientów rozkładów prawdopodobieństwa.

Dodajemy szum, dostajemy kierunek obraz czysty – zaszumiony; uczymy wektora w stronę czystego obrazu – tworzymy mapę gradientów w przestrzeni obrazów.



Modele IMM

- Zhou, L., Ermon, S., & Song, J. (3/2025). [*Inductive Moment Matching*](#).
- Wyjaśnienie: [*Inductive Moment Matching \(3/2025\) – YouTube*](#)
- Używane przez Luma Photon, w połowie 2025 roku to najszybciej działający model. Podobnie jak modele dyfuzyjne, może startować od szumu, z mniejszą liczbą kroków.



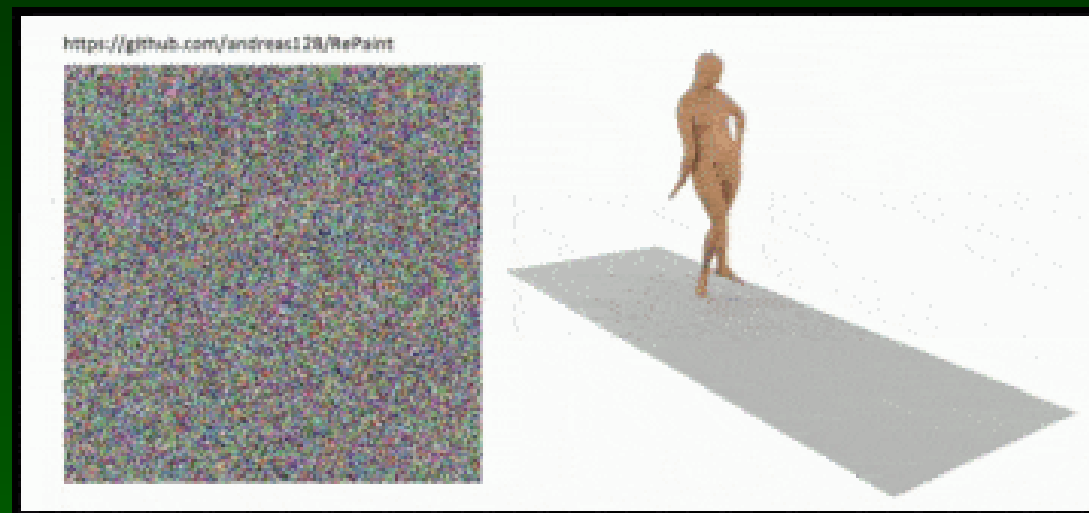
Superresolution – odszumianie

Od niskiej (LR) do wysokiej (HR) rozdzielczości. Idea: degradacja HR zależy od parametrów rozmycia k , szumu n , skalowania s , kompresji obrazu q które sieć próbuje przewidzieć.

Autoenkodery, modele dyfuzyjne, pozwalają na transformację obrazów LR => HR.



Wiele modeli tekst-obrazy, tekst-wideo, używa ukrytych modeli dyfuzyjnych:
Stability AI Stable Diffusion, OpenAI DALL·E 2, Google Imagen, GLIDE, Midjourney.
RectifiedHR (3/2025) tworzy obrazy w 1/10 sekundy!



Modele dyfuzyjne mogą tworzyć całe zdania jednocześnie, zamiast tworzyć kolejne słowa.

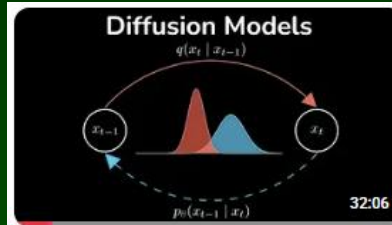
Nie, S. et al. (2025). Large Language Diffusion Models. [arXiv:2502.09992](https://arxiv.org/abs/2502.09992)

LLaDA (Large Language Diffusion with mAsking)

Generative AI Animated

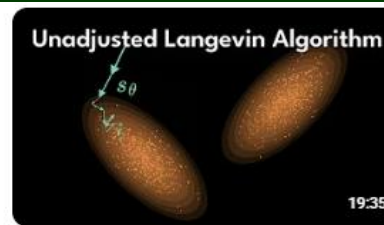
Animacja algorytmów Deepia

- [Latent Space Visualisation: PCA, t-SNE, UMAP](#)
- [Autoencoders](#)
- [Denoising Autoencoders](#)
- [Diffusion Models: DDPM](#)
- [Variational Autoencoders](#)
- [MIT Sloan course and tools](#)



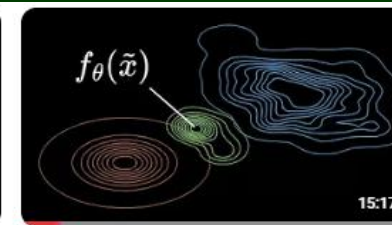
Diffusion Models: DDPM | Generative AI Animated

2K views • 13 hours ago



Unadjusted Langevin Algorithm | Generative AI Animated

19K views • 5 months ago



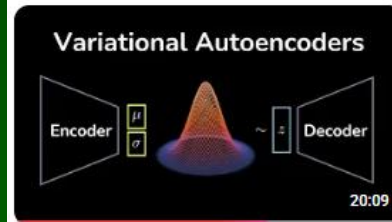
Denoising Autoencoders | Deep Learning Animated

27K views • 5 months ago



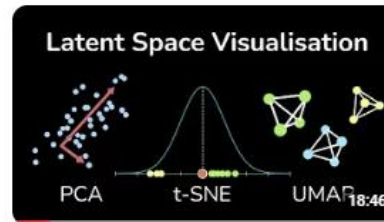
Contrastive Learning with SimCLR | Deep Learning Animated

15K views • 7 months ago



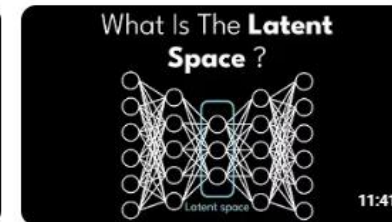
Variational Autoencoders | Generative AI Animated

100K views • 8 months ago



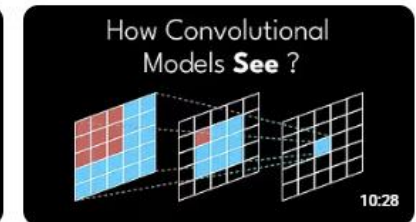
Latent Space Visualisation: PCA, t-SNE, UMAP | Deep Learning Animated

83K views • 9 months ago



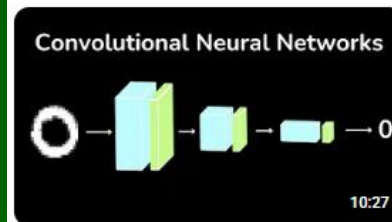
Autoencoders | Deep Learning Animated

56K views • 11 months ago



CNN Receptive Field | Deep Learning Animated

18K views • 11 months ago

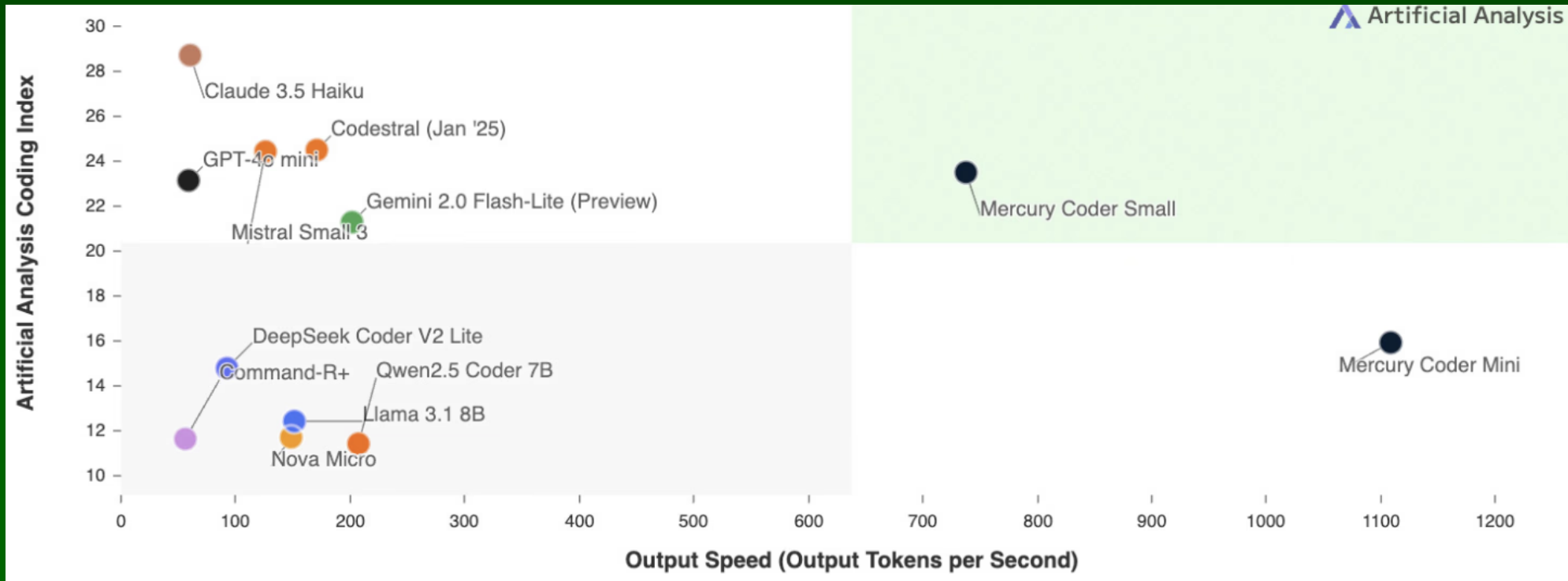


Convolutional Neural Networks | Deep Learning Animated

33K views • 1 year ago

Dyfuzyjne LLMs

- Inception Labs Mercury, pierwszy dyfuzyjny model językowy na skalę komercyjną.
- Rodzina dyfuzyjnych dużych modeli językowych (dLLM) Mercury, nowa generacja LLM, która przesuwa granicę szybkiego generowania wysokiej jakości tekstu. Generuje ponad 1000 tokenów/s na komputerach NVIDIA H100.
- Model generowania kodu, Mercury Coder, jest dostępny do testowania.
- Takie modele umożliwią głębokie rozumowanie i odpowiedzi bez żadnych opóźnień.



Tutorials on ML

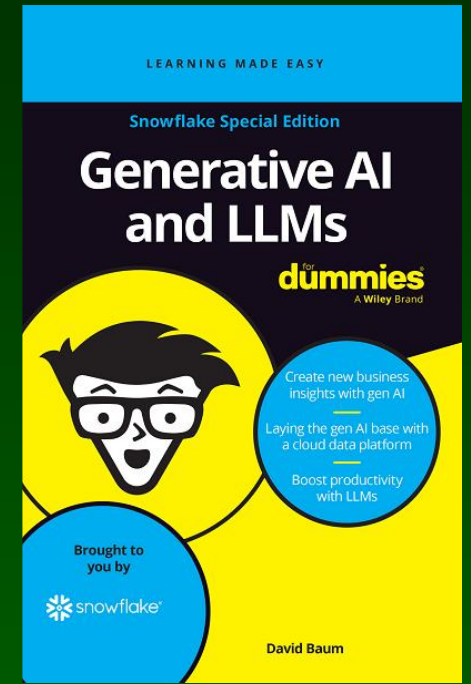
- Autocoder explanations

<https://www.youtube.com/watch?v=qJeaCHQ1k2w>

- More ML tutorial videos
- Deep Learning automated

<https://www.youtube.com/@Deepia-ls2fo>

Generative AI and LLMs for Dummies



Pytania

- Model Isinga
- Maszyny Boltzmana
- Sieci konwolucyjne
- Jak działają modele dyfuzyjne.