Нормализующие потоки в TTS

Свищев Алексей
ЦРТ, ведущий научный сотрудник

Нормализующие потоки
вывод

Пусть:

$$X = x_i, i = 1, 2, ..., N \qquad (1)$$

наблюдаемые данные
многомерного случайный вектора с неизвестной плотнос $p(X)$

Хотим найти максимально правдоподобные парам θ ры модели для аппроксимации данных, порожденных неизвестным распределением

$$L(X|\theta) = \log(p(X|\theta)) = \sum_{i=1}^{N} \log p(x_i|\theta) \qquad (2)$$

Максимизация логарифмической функции правдоподобия позволяет найти решение задачи

$$p(X|\theta) = \prod_{i=1}^{N} p(x_i|\theta)$$

Равенство (2) возможно в силу предположения о независимости наблюдаемых данн е.

http://akosiorek.github.io/ml/2018/04/03/norm_flows.html

Нормализующие потоки
вывод

Исходное истинное распределение вероятн $p(X)$ а аппроксимируем

некоторым параметризованным распределе $p_X(X|\theta)$ которое задается нейросетью

Один из способов аппроксимации - применить к нормальному распределению $p_Z(z) = \mathcal{N}(z; 0, 1)$

несколько нелинейных преобразований, сохраняя полную

$$\int p_Z(z)dz = 1 \qquad (3)$$

$p_X(X|\theta)$ $x = x(z)$

хотим упростить, и $\frac{dx(z)}{dz} \neq 0 \forall z$ тему координат с помощью преобразования

$$\int p_X(x)dx = \int p_X(x(z))\frac{dx(z)}{dz}dz = \int p_Z(z)\frac{dx(z)}{dz}dz \qquad (4)$$

$$\int_a^b f(x)\,dx = \int_\alpha^\beta f[\varphi(t)]\,\varphi'(t)\,dt.$$

Нормализующие потоки вывод

Пусть

$$z = z(x) = f(x) \qquad (5)$$

и существует

$$g(z) = f^{-1}(z(x)) = x \qquad (6)$$

и

$$\frac{dg(z)}{dz} \neq 0, \forall z \frac{df(x)}{dx} \neq 0, \forall x \qquad (7)$$

тогда (4) можно переписать

$$\int p_X(x)dx = \int p_Z(z)|\tfrac{dg(z)}{dz}|dz = \int p_Z(z)|(\tfrac{dz(x)}{dx})^{-1}|dz = \int p_Z(z)|(\tfrac{df(x)}{dx})^{-1}|dz \qquad (8)$$

Нормализующие потоки вывод

Трансформация одного распределения с сохранением нормировки и есть простейший поток (*flow*).

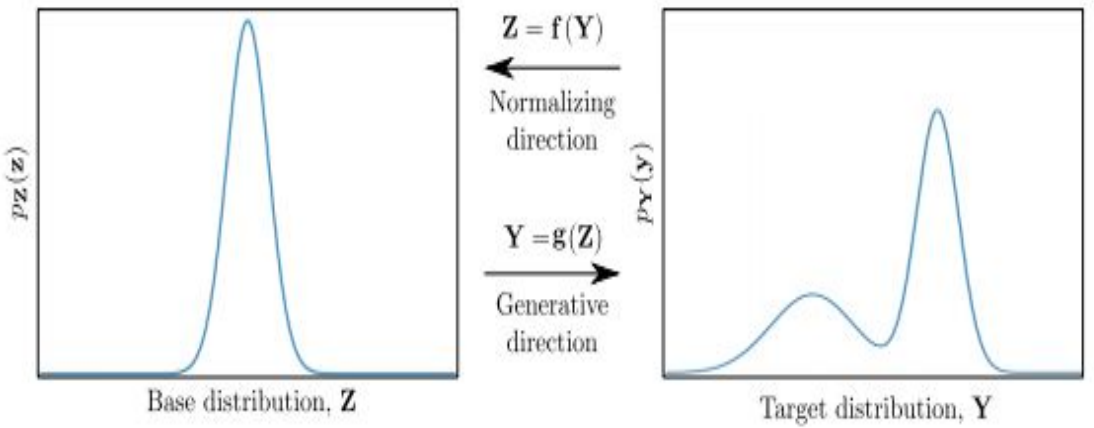
Совокупность потоков строится как последовательное применение функций преобразования (композиции функций):

$$f : f = f_1 \circ f_2 \circ \ldots \circ f_M \qquad (9)$$

$f(x)$ нормализующий поток (normalizing flow)

$g(z)$ генеративный поток (generative flow)

$$g = g_M \circ g_{M-1} \circ \ldots \circ g_1 \qquad (10)$$



Идея метода

на обучении нормализующий поток

используя обратимость $f(x)$

на инференсе генеративный

Нормализующие потоки
функция потерь

Потребуем

$$\frac{\partial f_{ij}(x)}{\partial x} = \frac{\partial z_i}{\partial x_j} \neq 0, \forall i, j = 1, ..., n \qquad \textbf{(11)}$$

также пусть существует
и ее производные не равны нулю

$$g_{ij}(z) = f_{ij}^{-1}(z(x)) \frac{\partial g_{ij}}{\partial z} \neq 0 \qquad (12)$$

Тогда можно составить матрицу Якоби
которая будет обратимой

$$J_{f(x)} = \begin{pmatrix} \frac{\partial z_1}{\partial x_1} & \cdots & \frac{\partial z_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_n}{\partial x_1} & \cdots & \frac{\partial z_n}{\partial x_n} \end{pmatrix} = \begin{pmatrix} \frac{\partial f_1(x)}{\partial x_1} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(x)}{\partial x_1} & \cdots & \frac{\partial f_n(x)}{\partial x_n} \end{pmatrix} \qquad (13)$$

Замена переменных в многомерном случае

$$\int_{-\infty}^{\infty} p_X(x_1, ..., x_n) dx_1 ... dx_n = \int_{-\infty}^{\infty} p_Z(z_1 ... z_n) |\det J_{f(z)}|^{-1} dz_1 ... dz_n \qquad (14)$$

Нормализующие потоки
функция потерь

Рассматривая M последовательно применяемых потоков

$$f : f = f_1 \circ f_2 \circ ... \circ f_M \tag{15}$$

Согласно правилу дифференцирования сложной функции детерминант Якобиана такой цепочки

$$\det J_{f(z)} = \prod_{m=1}^{M} \det J_{f_m(z)} \tag{16}$$

$$\det J_{g(x)} = \prod_{m=1}^{M} \det J_{g_m(x)} \tag{17}$$

$$p_X(x|\theta) = p_Z(f(x))|\det J_g(x)(x)| = p_Z(f(x))|\prod_{m=1}^{M} \det J_{g_m(x)}| \tag{18}$$

Результат применения совокупности нормали $p_Z(z)$ х потоков к исходному простому

Нормализующие потоки функция потерь

В левой части (18) аппроксимирует $p(x_i|\theta)$
из формулы (2)

Подставим (16) и (18) в (2)

$$L(X|\theta) = \sum_{i=1}^{N} \log p_X(x_i|\theta) = \sum_{i=1}^{N} \log \left(p_Z(z) \middle| \prod_{m=1}^{M} \det J_{g_m(x,\theta)} \middle| \right) \tag{19}$$

$$\sum_{i=1}^{N} \log \left(p_Z(z) \middle| \prod_{m=1}^{M} \det J_{g_m(x,\theta)} \middle| \right) = \sum_{i=1}^{N} \log \left(p_Z(z)\right) - \sum_{i=1}^{N} \sum_{m=1}^{M} \log \left|\det J_{f_m(z,\theta)}\right|$$

$$\det J_{f_m(x,\theta)} = \left|\det J_{g_m(z,\theta)}\right|^{-1}$$

Нормализующие потоки
Flowtron

В качестве исходного распределения в модели предлагается использовать нормальное

$$z \sim N(z; 0, 1) \tag{20}$$

Преобразование исходного распределени помощью потоков выглядит так

$$p_X(x) \to p_Z(z) : z = f_1 \circ f_2 \circ ... \circ f_M(x) \tag{21}$$

$$p_Z(z) \to p_X(x) : x = f_M^{-1} \circ f_{M-1}^{-1} \circ ... \circ f_1^{-1}(z) = g_M \circ g_{M-1} \circ ... \circ g_1(z)$$

В качестве функции потока был выбран аффинный связывающий слой

$$x_t^{m-1} = f_m(x_t^m) = s_t^k \circ x_t^m + b_t^m \tag{22}$$

$$(\log s_t^m, b_t^m) = NN_m(x_{1:t-1}^m, speaker, text) \tag{23}$$

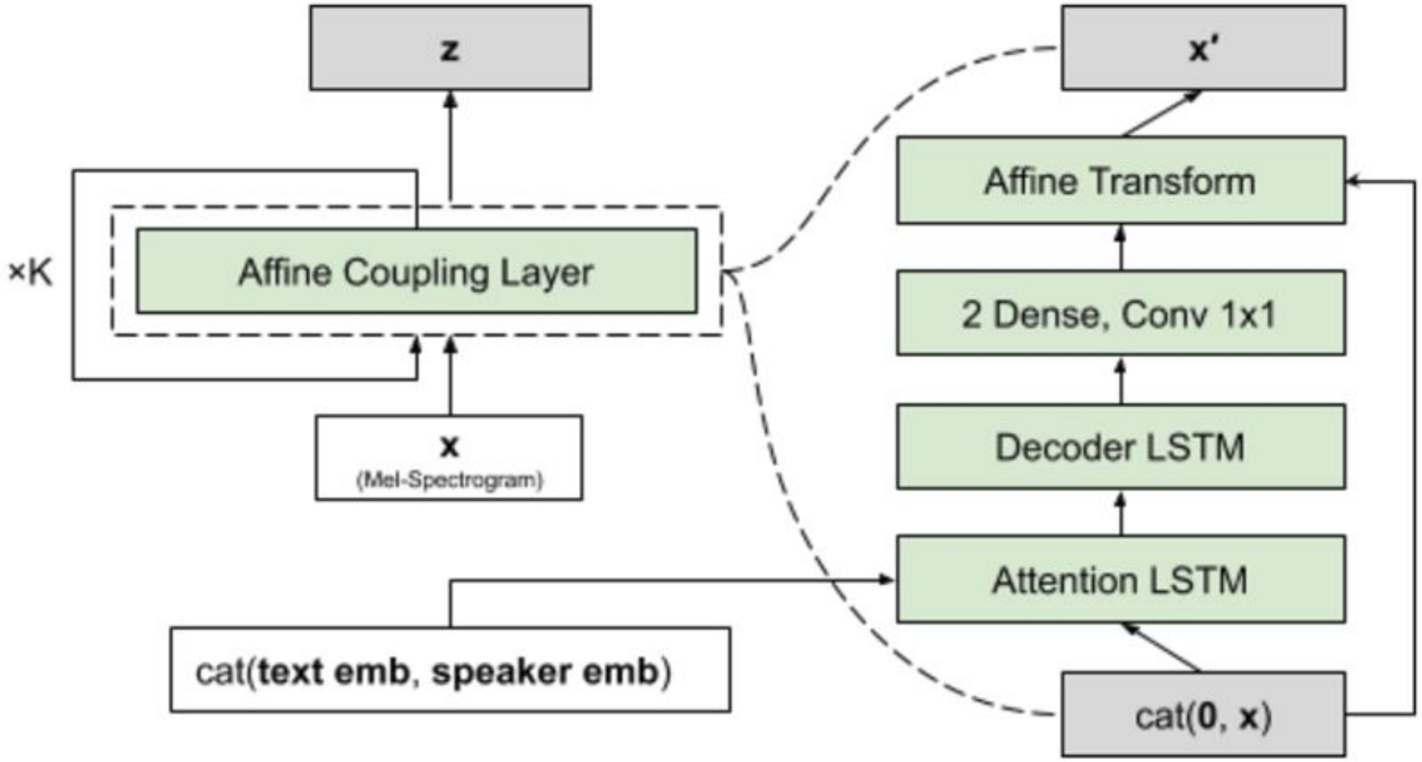
https://arxiv.org/pdf/2005.05957.pdf

Нормализующие потоки
Flowtron

В обратном направлении первым входом в поток служит Mel-спектрограмма известной фразы (*text*), сказанной диктором (*speaker*).

Нейронная сеть:

- выстраивает attention
- вычисляет scale и bias



https://github.com/NVIDIA/flowtron

Нормализующие потоки
Flowtron

Если хотим генерировать речь на основе текста и диктора то:

$$x_t^m = g_k(x_t^{m-1}) = \frac{x_t^{m-1} - b_t^m}{s_t^m} \tag{24}$$

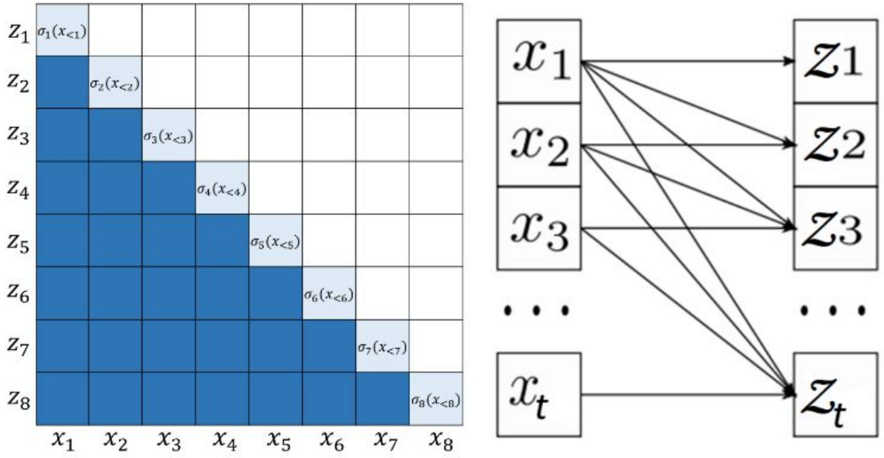
$$(\log s_t^m, b_t^m) = NN_m(x_{1:t-1}^{m-1}, speaker, text) \tag{25}$$

Коэффициенты предсказания потока зависят от результатов предыдущего предсказания

$$x_t^{m-1} = f_m(x_{1:t-1}^m) \tag{26}$$

Из этого следует что матрица зависимостей является ниже-треугольной, а значит

$$\det J_{f_m(x)} = \prod_{t=1}^{D} \frac{\partial f_m(x_t)}{\partial x_t} \tag{27}$$



Нормализующие потоки Flowtron

Принимая во внимание (22) производная

$$f_m(x_t) = s_t^m \circ x_t^m + b_t^m \Rightarrow \frac{\partial f_m(x_t)}{\partial x_t} = s_t^m \quad (28)$$

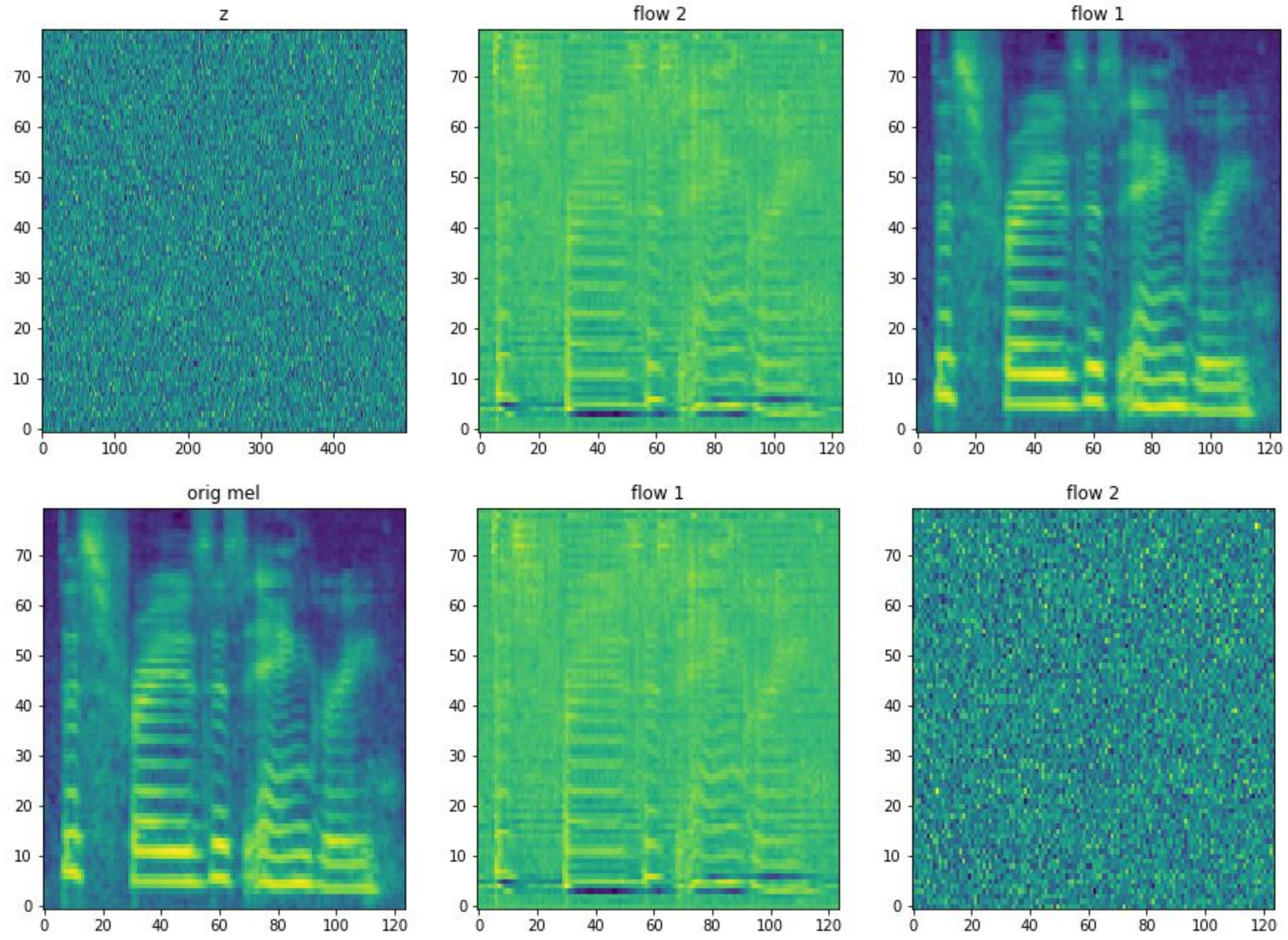
Подставляя в (19) выражения (27) и (28) получим

$$L(X|\theta) = \sum_{i=1}^{N} \log p_Z(z) - \sum_{i=1}^{N} \sum_{m=0}^{M} \log |\det J_{f_m(x)}| = \sum_{i=1}^{N} \log p_Z(z) - \sum_{i=1}^{N} \sum_{m=0}^{M} \sum_{t=0}^{D} \log |\det s_t^m| \quad \mathbf{(29)}$$

Нормализующие потоки
Flowtron

Направление генерации

Направление нормализации



Нормализующие потоки
некоторые функции потоков

Linear Flows (A - обратимая

$$\mathbf{g}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$$

Planar and Radial Flows

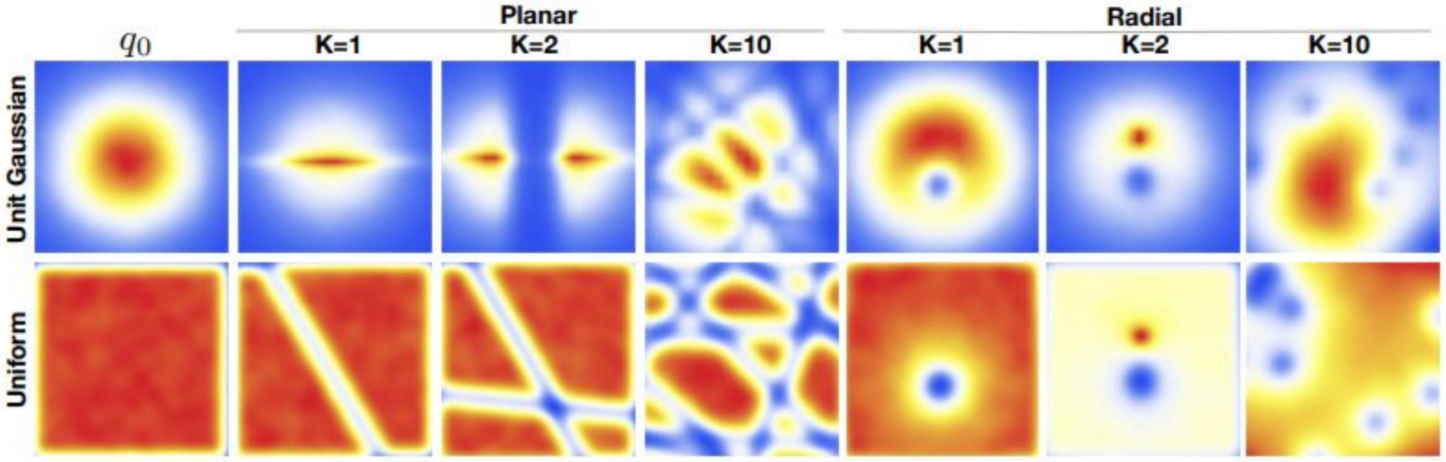
$$\mathbf{g}(\mathbf{x}) = \mathbf{x} + \mathbf{u}h(\mathbf{w}^T\mathbf{x} + b).$$

$$\det\left(\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right) = \det(\mathbb{1}_D + \mathbf{u}h'(\mathbf{w}^T\mathbf{x} + b)\mathbf{w}^T) = 1 + h'(\mathbf{w}^T\mathbf{x} + b)\mathbf{u}^T\mathbf{w},$$

$$\mathbf{g}(\mathbf{x}) = \mathbf{x} + \frac{\beta}{\alpha + \|\mathbf{x} - \mathbf{x}_0\|}(\mathbf{x} - \mathbf{x}_0)$$

Функции потоков определяют выразительность потока

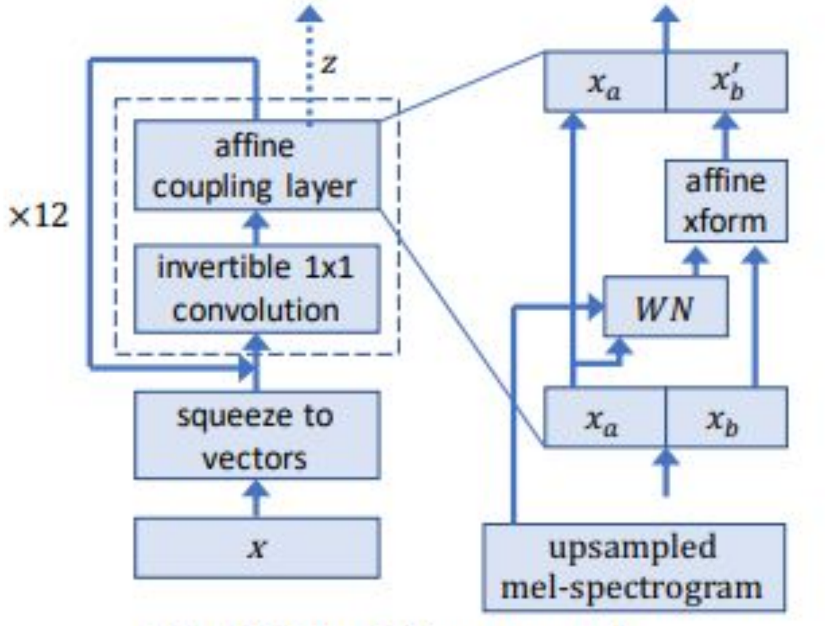
От функции зависит вычисление Якобиана



https://arxiv.org/pdf/1908.09257.pdf - хороший обзор по потокам
https://arxiv.org/pdf/1505.05770.pdf

Нормализующие потоки
WaveGlow

- Архитектура схожа с Wavenet
- Параллельный инференс (быстро)
- Последовательное обучение (долго)
- Требует шумоподавления



$$\log p_\theta(\boldsymbol{x}) = -\frac{\boldsymbol{z}(\boldsymbol{x})^T \boldsymbol{z}(\boldsymbol{x})}{2\sigma^2} + \sum_{j=0}^{\#coupling} \log \boldsymbol{s}_j(\boldsymbol{x}, \textit{mel-spectrogram}) + \sum_{k=0}^{\#conv} \log \det |\boldsymbol{W}_k|$$

$$\boldsymbol{x}_a, \boldsymbol{x}_b = split(\boldsymbol{x})$$
$$(\log \boldsymbol{s}, \boldsymbol{t}) = WN(\boldsymbol{x}_a, \textit{mel-spectrogram})$$
$$\boldsymbol{x}_b\prime = \boldsymbol{s} \odot \boldsymbol{x}_b + \boldsymbol{t}$$
$$\boldsymbol{f}^{-1}_{coupling}(\boldsymbol{x}) = concat(\boldsymbol{x}_a, \boldsymbol{x}_b\prime)$$

https://arxiv.org/pdf/1811.00002.pdf