Language Embeddings

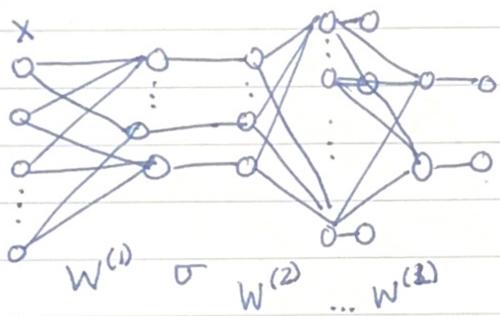
Plan

Review
Autoencoders
Contrastive Learning
PCA

Logistics

Games tonight!
Check in form!
Scribed notes!
Zoom!

Review



1. Model $f: \mathbb{R}^d \to \mathbb{R}^k$

2. Loss $\mathcal{L}(w)$

3. Optimizer

$$w^{(t+1)} = w^{(t)} - \alpha \nabla \mathcal{L}(w^{(t)})$$

momentum $v^{(t+1)} = (1-\beta)\, v^{(t)} + \beta \nabla_w \mathcal{L}(w^{(t)})$

adaptivity $s^{(t+1)} = (1-\beta)\, s^{(t)} + \beta [\nabla_w \mathcal{L}(w^{(t)})]^2$

Backprop:

Forward:
for i in $\{1, \dots, N\}$:
compute v_i from Parents(v_i)

Backward:
for i in $\{N, \dots, 1\}$:
compute $\frac{\partial \mathcal{L}}{\partial v_i} = \sum_{j \in \text{Children}(v_i)} \frac{\partial \mathcal{L}}{\partial v_j} \frac{\partial v_j}{\partial v_i}$

ASIDE

$$\begin{tikzcd} & v = v(u) \arrow[dr] & \\ u \arrow[ur] \arrow[dr] & & z = z(v,w) \\ & w = w(u) \arrow[ur] & \end{tikzcd}$$

$$\frac{\partial z}{\partial u} = \frac{\partial z}{\partial v} \cdot \frac{\partial v}{\partial u} + \frac{\partial z}{\partial w} \frac{\partial w}{\partial u}$$

$v(u+\delta) \approx v(u) + \delta \frac{\partial v}{\partial u}(u)$ **

$w(u+\delta) \approx w(u) + \delta \frac{\partial w}{\partial u}(u)$

* w/o δ^2 or higher

$z(u+\delta) = z(u) + \delta \frac{\partial v}{\partial u} \frac{\partial z}{\partial v} + \delta \frac{\partial w}{\partial u} \frac{\partial z}{\partial w}$

$$\frac{\partial z}{\partial u} = \lim_{\delta \to 0} \frac{z(v(u+\delta), w(u+\delta)) - z(v,w)}{\delta} \approx \frac{\partial z}{\partial v}\frac{\partial v}{\partial u} + \frac{\partial z}{\partial w}\frac{\partial w}{\partial u}$$

** $= \lim_{\delta \to 0} \frac{z(v + \delta \frac{\partial v}{\partial u}, w + \delta \frac{\partial w}{\partial u}) - z(v,w)}{\delta}$

* $z(v+\delta_1, w+\delta_2) \approx z(v,w) + \delta_1 \frac{\partial z}{\partial v}(v,w) + \delta_2 \frac{\partial z}{\partial w}(v,w)$

Language Embeddings

Our models take vectors... how can we convert words to vectors?

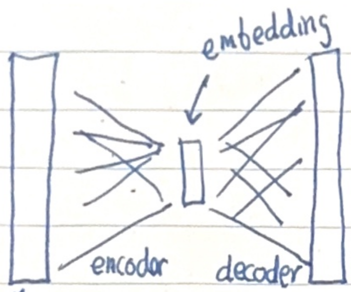
Approach #1: One-hot encodings

⊖ not meaningful

⊖ large

Why train decoder?

Approach #2: Autoencoders



one hot encoding
vocab size x1

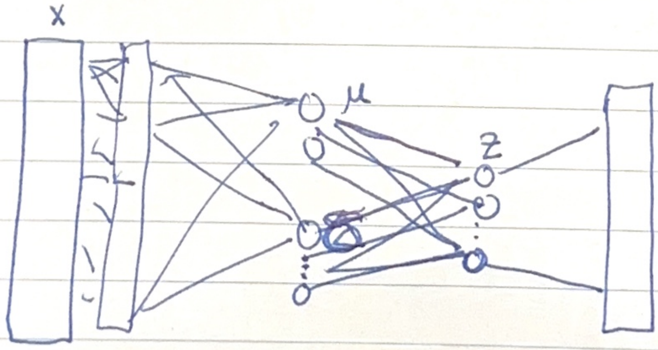
Questions for Activity:

- What loss?
- How to make similar words close?

Approach #3: Variational Autoencoders

What if we want embeddings to be nicely distributed in latent space?

$z \sim N(0, I)$ $z \in \mathbb{R}^r$ $z_i = \mu_i + \sigma_i \epsilon_i$ for $N(0,1)$



$$\mathcal{L}(w) = \| f(x) - x \|_2^2 + \lambda \, KL(N(0,I), N(\mu,\sigma))$$

distance between distributions

$$KL(p\|q) = H(p,q) - H(p)$$

Contrastive Learning

We're working on <u>unsupervised</u> task i.e., no labels

positive (word, next_word) hopefully close

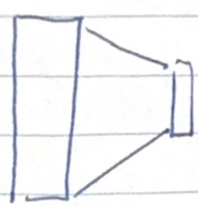
↳ (x, x^+) $f(x)^T f(x^+)$ large

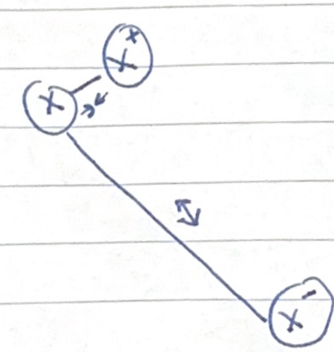
negative (word, unrelated_word) probably far

↳ (x, x^-) $|f(x)^T f(x^-)|$ small

$$\mathcal{L}(w) = \sum_{x,x^+} f(x)^T f(x^+) - \sum_{x,x^-} \left[f(x)^T f(x^-)\right]^2$$

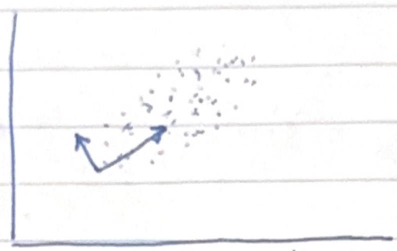
$f: \mathbb{R}^d \to \mathbb{R}^n$





Principal Component Analysis

Motivation: visualize points in $\mathbb{R}^d$ meaningfully



$X \in \mathbb{R}^{n \times d}$ $\begin{bmatrix} \\ \\ \\ \end{bmatrix}$

rotation = linear operation

Find $v \in \mathbb{R}^d$: $\underset{n \times d}{X} \underset{d \times 1}{v}$ is meaningful ie., captures variation of data!

$\iff \|Xv\|_2^2$ is large $\iff v^T X^T X v$ is large.

$$\max_{v: \|v\|_2^2 = 1} v^T X^T X v \iff \text{largest eigenvalue of } X^T X$$

$$\max_{v: \|v\|_2^2 = 1,\ v \perp v^{(1)}} v^T X^T X v \iff \text{2nd largest eigenvalue}$$

$$X^T X = \sum_{i=1}^{r \,\swarrow \text{rank}} \lambda_i \underset{d \times 1}{v^{(i)}} \underset{1 \times d}{v^{(i)T}} \quad \text{where } \|v^{(i)}\|_2^2 = 1 \text{ and } v^{(i)} \cdot v^{(j)} = 0$$

maximize w/ eigenvalue corresponding to largest λ_i

sanity check: $X^T X v^{(j)} = \sum_{i=1}^{r} \lambda_i v^{(i)} v^{(i)T} v^{(j)} = \lambda_j v^{(j)} v^{(j)} \cdot v^{(j)} = \lambda_j v_j$