

Computer-Linguistische Anwendungen

CLA | B.Sc. | LMU



Skipgram Negative Sampling (SGNG)

Ziel

10''



Skipgram Negative Sampling (SGNG)

$$\operatorname{argmax}_{\theta} \left[\sum_{(w,c) \in D} (\vec{v}_w \cdot \vec{v}_c) + \beta \sum_{(w,c) \in V \times V} (-\vec{v}_w \cdot \vec{v}_c) \right] \quad \text{fast komplett!}$$

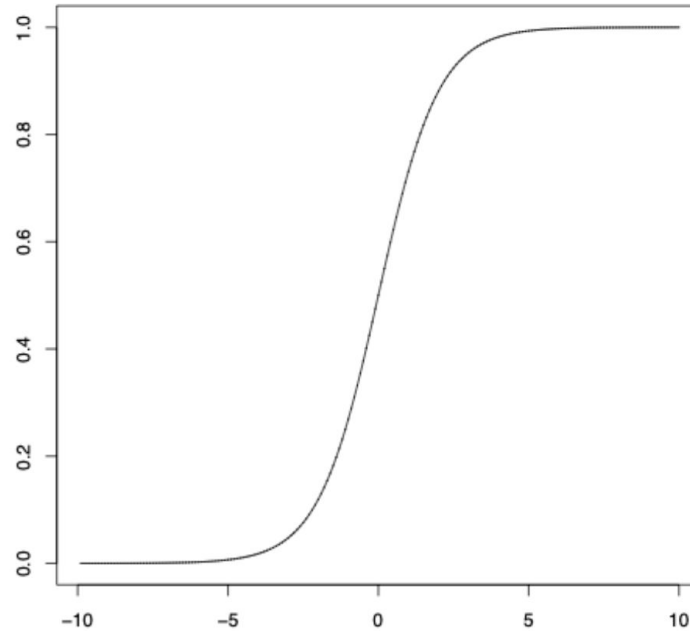
- Training Daten D : Wort-Kontext-Wort-Paare (w, c)
- Wir lernen ein **embedding** \vec{v}_w für jedes w , wir lernen ein **embedding** \vec{v}_c für jedes c
- Jedes Wort hat 2 embeddings: **input embedding** und **context embedding**
- β : ein Gewichtungsfaktor um den Einfluss des zweiten Terms der Funktion zu reduzieren (weil wir wesentlich mehr FALSCH Paare haben)
- Generell verwenden wir nur die **input embedding** für weitere Berechnungen, e.g. Ähnlichkeitsberechnung
- Lies: Mache das Dot-Product der “true” (GOOD) Paare so groß wie möglich
- Lies: Mache das Dot-Product der “false” (BAD) Paare so klein wie möglich (davon gibt es viele)

Skipgram Negative Sampling (SGNG)

$$\operatorname{argmax}_{\theta} \left[\sum_{(w,c) \in D} \log \sigma(\vec{v}_w \cdot \vec{v}_c) + \beta \sum_{(w,c) \in V \times V} \log \sigma(-\vec{v}_w \cdot \vec{v}_c) \right]$$

- $\sigma(x) = 1 / (1 + e^{-x})$: Sigmoid Funktion welche uns Werte zwischen 0 and 1 zurückgibt, also eine binäre Wahrscheinlichkeit
- $\sigma(\vec{v}_w \cdot \vec{v}_c)$: $P(GOOD|w, c)$
- $\sigma(-\vec{v}_w \cdot \vec{v}_c)$: $1 - P(GOOD|w, c) = P(BAD|w, c)$
- Lies: mach den Sigmoid des dot-product der “true” (GOOD) Paare so groß wie möglich
- Lies: mach den Sigmoid des dot-product der “false” (BAD) Paare so klein wie möglich

σ : Logistic = Sigmoid



X-Axis: $\vec{v}_w \cdot \vec{v}_c$

Y-Axis: $\sigma(\vec{v}_w \cdot \vec{v}_c) = P(GOOD|w, c)$