

Δ pycne croxkubating: $u_1 u_2 \dots u_{n-1} w$

$$P(w | u_1 u_2 \dots u_{n-1}) = \frac{c(u_1 \dots u_{n-1} w)}{c(u_1 \dots u_{n-1})}$$

n-gramme

tex
Likelyhood

Morph

$$\begin{aligned} 3_{rp}: \quad \text{wörter} &\xrightarrow{\text{§ unkl}} -0 \\ 2_{rp} &\xrightarrow{\text{§ unk}} -10 \\ 1_{rp}: \quad \text{morph} &\xrightarrow{-20} \end{aligned}$$

Unterphrasen und croxkubating

$$P(w | u_1 u_2 \dots u_{n-1}) = \lambda_{h1} \frac{c(u_1 \dots u_{n-1} w)}{c(u_1 \dots u_{n-1})} + \lambda_{h2} \frac{c(u_2 \dots u_{n-1} w)}{c(u_2 \dots u_{n-1})} + \dots + \lambda_1 \frac{c(u_{n-1} w)}{c(u_{n-1})} + \lambda_0 \frac{c(w)}{N}$$

grüne Körnchen

λ reale wogdunz, $\sum \lambda_i = 1$

Ausg. für wörter

$$P(w | u) = \lambda_1 \frac{c(uw)}{c(u)} + \lambda_0 \frac{c(w)}{N}$$

$$\sum_{w \in V} P(w | u) = 1$$

$$1 = \sum_w \left(\lambda_1 \frac{c(uw)}{c(u)} + \lambda_0 \frac{c(w)}{N} \right) = \lambda_1 \sum_w \frac{c(uw)}{c(u)} + \lambda_0 \sum_w \frac{c(w)}{N} = \lambda_1 + \lambda_0$$

$$\lambda_1 + \lambda_0 = 1$$

$$0.5 \quad 0.5$$

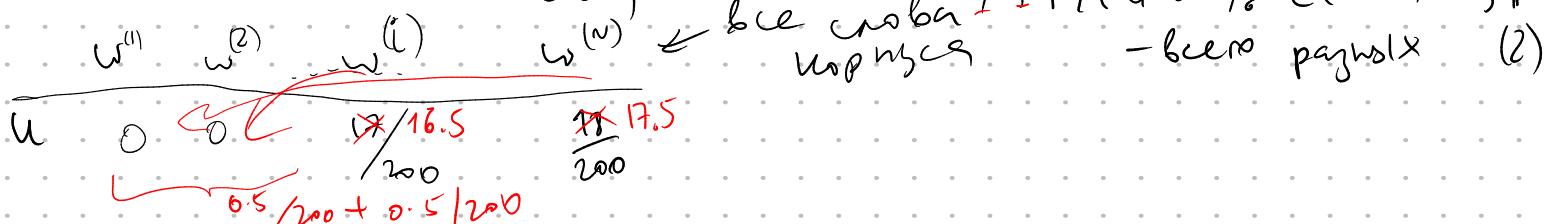
$$0.8 \quad 0.2$$

$$0.2 \quad 0.8$$

wogdop
grüne Körnchen
metab.

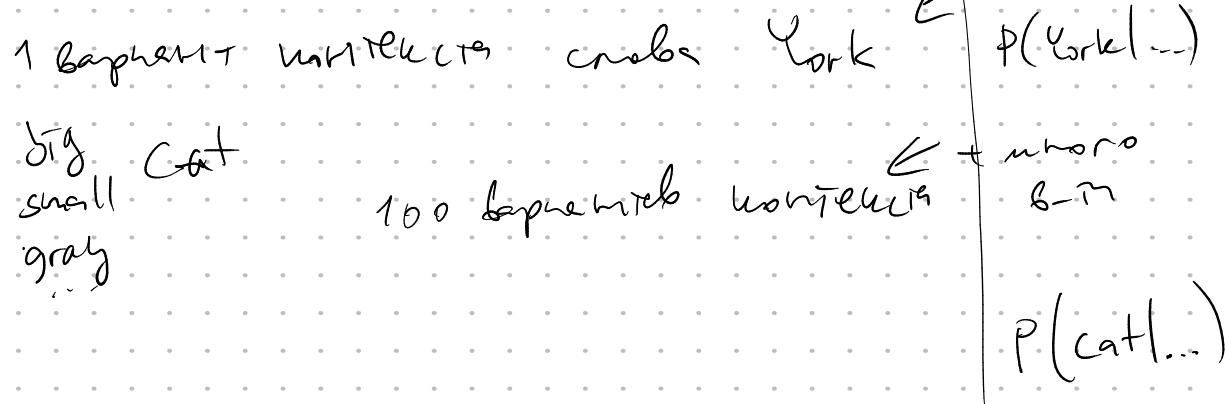
Nyumenti Lapshant: Kheysen-Ney.

$$P(w | u) = \frac{c(uw) - \delta}{c(u)} + \frac{\delta}{\text{zurück evn } c(uw) > 0} \quad (1)$$



Больше всего мы можем организовать один класс, например
Черепахи лучше в наборах контекстов.
(и которых познаний нету).

Пример: New York



(1) - non-B0 параметров, которые встречаются в w

(2) - non-B0 всех crab, которые больше соответствуют контекстам в контексте.

$$\lambda - \text{недостаток}, \text{так что } \sum_w p(w|_i) = 1.$$

Слева недостаток синхронизации в машинном обучении

Хотим близкого значения моделей: ~~1. MLE ($p = \frac{c(w)}{C(w)}$)~~ - 0.

синхронизация: metag / λ_i

$$2. \text{Lambda} \frac{c(w)}{C(w) + \lambda_i}$$

λ - размер

3. инвертировать,
модели λ_i

4. Kheyser-Ney $\delta = ?$

Компьютер		
85%	10%	10%
training set	validation set	test set
85%	10%	10%

Как решить:

all_sentences = [, ,]

shuffle - \rightarrow из многих
строк выбрать как

hacne shuffle → all-sentences [$: 0.8 * \text{len}$]
 → all-sentences [$0.8 * \text{len} : 0.9 * \text{len}$]
 → all-sentences [$0.9 * \text{len} :$]

N passen mögeln.

Bspn 1: 2e ^{grazende} \checkmark Kneip \rightarrow parameter (Naissance, $\lambda=1$)
 obigen Ha training-set, obenbares Varecbo \rightarrow Validations

Bspn 2: 2e ^{grazende} (Naissance, $\lambda=2$)

obigen Ha training-set obenbares Varecbo \rightarrow Validations

Bspn 3: 2e ^{grazende} (Kneser, Neg, $\delta = 0.1$)

obigen Ha training-set obenbares Varecbo \rightarrow Validations

$\hookrightarrow T \cdot g$

Modellgalt JETZT, y Voro Varecbo \rightarrow validation-set machen MUßO.

Other: modellkunst no gern + Varecbo \rightarrow test-set.

Obenbares Varecbo müssen ke Kopnyce (test-set)

P{Tenutz} ↑ Sonne - Regen

$$\begin{aligned} P\{\text{Tenutz}\} &= P\{\text{wegen 1}\} \cdot P\{\text{wegen 2}\} \cdots \cdot P\{\text{wegen } N\} = \\ &= \underbrace{P(w_1^{(1)} | \text{cs}) \cdot P(w_2^{(1)} | w_1^{(1)}) \cdots P(\text{cs} | w_{n_1}^{(1)})}_{\text{P } (w_1^{(2)} | \text{cs}) \cdot P(w_2^{(2)} | w_1^{(2)}) \cdots P(\text{cs} | w_{n_2}^{(2)})} \cdot \underbrace{\dots}_{\text{underflow}} \end{aligned}$$

$$\log P\{\text{Tenutz}\} = \log P(+) + \log P(-) + \dots + \log P(-)$$

- hat underflow

Grundidee rech - beobachtet möglich.

harry seyt he p texts, a cegtho
bepostahtas h-param.

$$\overbrace{P(\text{text})}^N$$

$$\frac{1}{N} (\log P(\text{text}))$$

measur

N-param

$$E = -\frac{\log P(w_1^{(1)} | s) + \log P(1) + \dots + \log P(s_i | w_n^{(n)})}{N} > 0$$

N = bese n-param.

$$\log P(.) < 0$$

Perplexity = 2^E /ceghs Bepostaht ogho' n-param.

perplexity = 20 \Rightarrow ceghs bepostaht = $1/20$

perplexity = 2000 \Rightarrow ceghs bepostaht $1/2000$ ↑

perplexity goetw dits keu mothe mense.