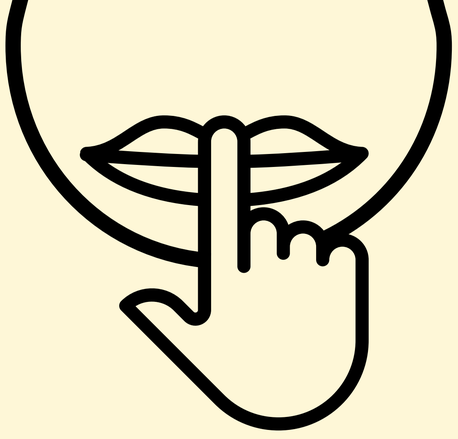
Tokenization and the Noiseless Channel



Which tokenization?	|V|
1) rapac ··i ··ous zeal ··ous aardvark ··s	16k
2) r ··ap ··ac ··i ··ous z ··eal ··ou ··s aar ··dvark ··s	32k
3) rapacious zealous aardvarks	1M
4) rapac ··ious zeal ··ous aardv ··arks	16k

vzouhar@ethz.ch

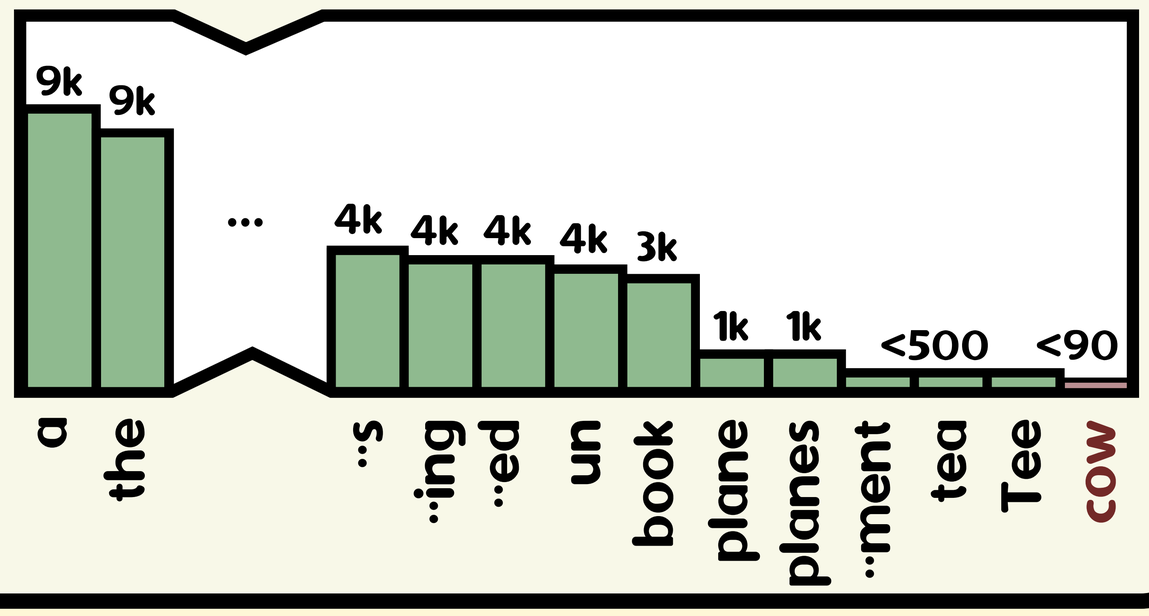
$$t^* = \underset{\text{tokenization} \in \mathscr{T}}{\text{argmax}}\ \text{performance}(\ \text{model}(\ \text{tokenization}(\ \text{data}\ )\ )\ )$$

- BLEU/COMET: 1 GPUhr
- Transformer: 7 GPUday
- BPE/Unigram: 1 CPUhr

A) Choose BPE with |V|=32k **pro:** covers >90% cases **con: hmm**

B) Look at token distribution

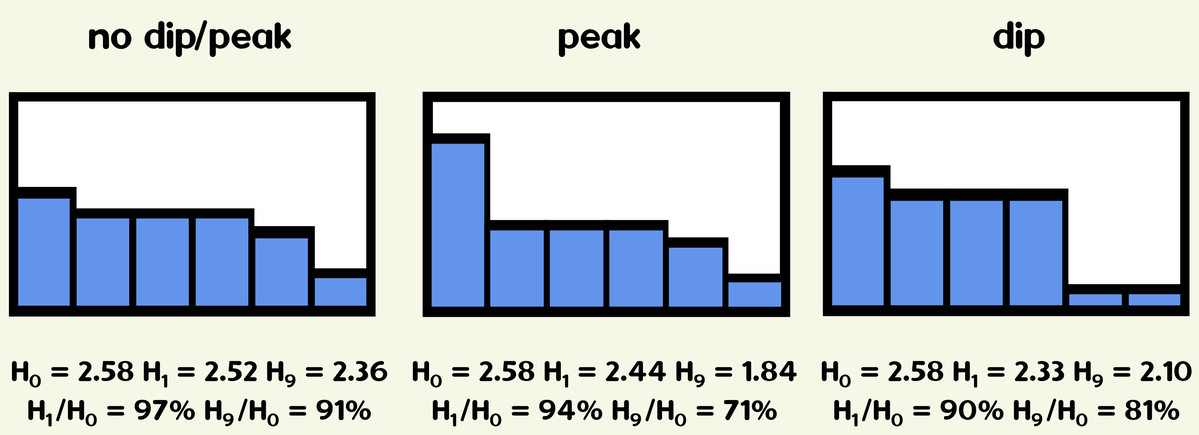
Stop when new token freq <90
pro: good heuristics
con: arbitrary, not a metric



C) Quantify balanced distributions using entropy (uniformity) (no very high or very low frequency tokens)

Entropy
penalizes low freq. tokens
$H(p) = -\sum \log p(x)$
$\text{eff}(p) = H(p) / \log |V|$

Rényi entropy
disproportionately penalizes low/high freq. tokens
$H_\alpha(p) = 1/(1-\alpha) \log \sum p(x)^\alpha$
$\text{eff}_\alpha(p) = H_\alpha(p) / \log |V|$




```
pip3 install tokenization-scorer

tokenization-scorer -i en-de.tok_unigramlm.{en,de}
> 0.4826

tokenization-scorer -i en-de.tok_wordpiece.{en,de}
> 0.5047
```

Vilém Zouhar, Clara Meister, Juan Luis Gastaldi, Li du, Mrinmaya Sachan, Ryan Cotterell