CS/INFO 6742: NLP and Social Interaction, Fall 2021

Nov. 11, 2021: Lecture 20: continued example of language-model development: latent information; distances between language models

---

1 Reminder: Motivating example: modeling small-talk vs. non-small talk

1.1 Sample data

Written "vertically" instead of "horizontally" to leave room to write.

Two sequences (in this case, monologue documents):

st hi
nst i
NST agree
ST thanks
ST bye

I want to see the states! ⇒ label the data. (training)

p(hi | st)

still: w/ new data, it won't have those labels.

P(hi hi hi).

labeling the test data:

hi
sell
hi [some stock ticker symbol]
now
thanks

1.2 A skeleton generative story

1. Pick a sentence length ℓ.
2. Pick a sequence of ℓ *states*: where the two possible state types are st for small talk, nst for not small-talk
3. For each state, pick a word according to that state's distribution over single words.

1.3 Ideas for instantiation (these are informal "priors")

1. (from last lecture) st might have a higher probability of *being in longer sentences than in shorter sentences*.
2. (motivation for step 2 and 3 of the generative story) st might have a higher probability of including the word "hi" than nst.
3. (new) st might have a higher probability of starting or ending the sentence than nst.

1.3.4 "Quiz": What is the probability of our first sample-data sequence?

Assume we pick specific lengths (not length "buckets" like "short" vs. "long") Note how the formulation below avoids talking about the state sequences' probs.

- P(a length-5 sequence (with respect to all possible lengths)) × P(st nst nst st st) × P(hi | st) P(i | nst) P(agree | nst) P(thanks | st) P(bye | st)
- P(a length-5 sequence) × $\sum_{\text{state sequences } \sigma_1\sigma_2\sigma_3\sigma_4\sigma_5}$ P(hi | σ_1) P(i | σ_2) P(agree | σ_3) P(thanks | σ_4) P(bye | σ_5)
- P(a length-5 sequence) × $\sum_{\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5}$ P($\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5$) P(hi | σ_1) P(i | σ_2) P(agree | σ_3) P(thanks | σ_4) P(bye | σ_5)
- Something else

where to get p(hi | σ)?

p(hi being word 1) = P(hi being 1st word | σ_1 = st) P(σ_1 = st) + P(hi being 1st word | σ_1 = nst) P(σ_1 = nst)

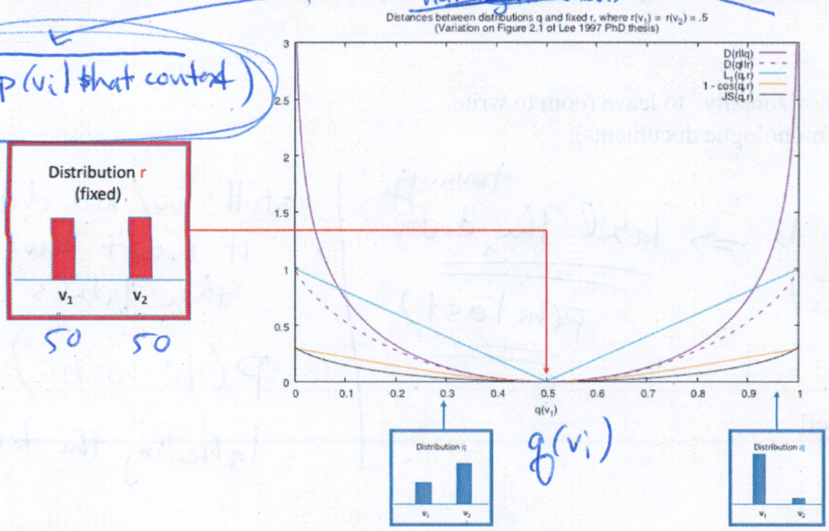
2 Measuring the difference between two "single-word" distributions

We restrict attention to proper distributions $q(\cdot)$ and $r(\cdot)$ over finite "vocabulary" $V = \{v_i\}$. We write q_i and r_i for $q(v_i)$ and $r(v_i)$.

• But LMs give probs to an unbounded number of strings? One can take V to be single words (or whatever), and for a given language model $p(\cdot)$, set p_i to $p(v_i|$some context of interest$)$. ~~p(the cat)~~

normalize

$\sum_i p(v_i | \text{that context})$



The *surprisal*[1]:

$$-\log(r_i) = \log \frac{1}{r_i} \tag{1}$$

can be thought of as how *surprised* we should be from the perspective of using r as a model to see v_i, or r's *surprisedness* or *surprisingness* for v_i. The base of the log is customarily taken to be 2, which makes this surprisingness number interpretable as a number of bits of information.[2]

2.1 Cross-entropy

If we considered the "reference" distribution to be q, then the *cross-entropy*

$$H(q||r) = \sum_i q_i \log \frac{1}{r_i} \tag{2}$$

is the expected surprisedness for r with respect to reference distribution q.[3]

2.2 KL-Divergence

$$D(q||r) = \sum_i q_i \log \frac{q_i}{r_i} \tag{4}$$

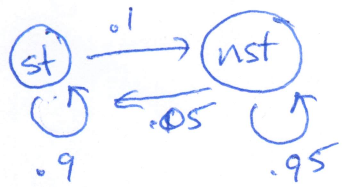
[1] According to Wikipedia, the term was coined in Tribus, 1961, *Thermostatics and Thermodynamics*.

[2] Indeed, a much more common interpretation of equation 1 is as a number of bits needed to encode v_i assuming the distribution r over V.

[3] *How you often see this in papers:* If the "reference" distribution is taken to be the one induced from the empirical counts from a sample $S = w_1 w_2 \ldots$, where each $w_k \in V$ and the length of the sample is L, then this can be refactored as:

$$\hat{H}_S(r) = \frac{1}{L} \sum_{k=1}^{L} \log \frac{1}{r(w_k)} \tag{3}$$

conventional idea:
use state transitions (somehow)



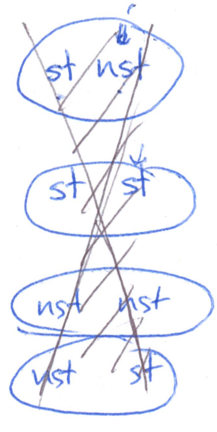
"idealish"

initial state probs: .99 in st
.01 in nst

conditioned on length choice.

problem ∅ — or, by construction this model ~~ing~~ instantiation is memoryless.

we need some ~~position~~/memory in the states



st nst nst nst st

or

(st nst st nst nst

~~smaller prob than above?!~~

$\approx 2^l$ states

~~$\approx (2^l$ parameters$)$~~

~~$\approx 2^l \times 2^l$ parameters~~

~~$\approx$ [illegible] parameters.~~

~~[illegible] transitions~~

too many state transitions, let's skip.

~~states~~ let's track:

p(st at word i)

~~2(length of sequence)~~

~~→ [illegible] parameters.~~

→ $\approx 2 \times l$ ~~parameters~~ s

→ $\approx 2 \times l$ states