Cross Validated

The effect of temperature in temperature sampling

Asked 2 years, 11 months ago Active 2 years, 11 months ago Viewed 3k times

4

3

I was reading this while I found:

> The high temperature sample displays greater linguistic variety, but the low temperature sample is more grammatically correct. Such is the world of temperature sampling - lowering the temperature allows you to focus on higher probability output sequences and smooth over deficiencies of the model.

How do you define temperature sampling?

In sigmoid, temperature is at the bottom of the exponent, so I know that as t--> infinity, sigmoid activation tends to 1. So, higher temperature corresponds to higher entropy.

Specifically, what is the explanation for behavior when τ->1 and τ→0? Intuitively, how do these limits modify probability to induce the kind of behavior mentioned above (smoothness, argmax)?

I am also seeing tempartureiin other places like in the temperature sampling above. Is it some general thing?

machine-learning probability mathematical-statistics sampling deep-learning

edited Jan 10 '17 at 6:24
Glen_b -Reinstate Monica
229k 24 457 810

asked Jan 9 '17 at 2:52
Rafael
797 8 23

My guess is that it is based on the simulated annealing metaphor, and considering the probability distribution to be the "objective function"? – GeoMatt22 Jan 9 '17 at 3:06

1 Right near the top of the page you link to it defines how the temperature parameter comes into it and what effect it has, right down to giving a formula. $\tilde{p}_i = f_\tau(p)_i = \frac{p_i^{\frac{1}{\tau}}}{\sum_j p_j^{\frac{1}{\tau}}}$ Can you give more of an indication of what is unclear? – Glen_b -Reinstate Monica Jan 9 '17 at 3:09

That formula looks like the output vector will have a reduced contrast relative to the input, for large "temperature" (e.g. similar to gamma correction in image processing). So high temperature would correspond to a more uniform output, i.e *higher* entropy, as you had expected? (Note: I did not read the link, so may have to update this again if @Glen_b reads more of it!) – GeoMatt22 Jan 9 '17 at 3:20

@Glen_b my understanding is that p is transformed using f and then we sample from f. Sampling from a language model means saying like give me 100 words and it will draw new words each conditioned on all of the previous ones. I don't understand why for tau = 1, the freezing function is just the identity function.

For τ→0, the freezing function turns sampling into the argmax function, returning the most likely output word. – Rafael Jan 9 '17 at 3:24

Note that $\sum p_j = 1$ so when $\tau = 1$ you have $f(p)_i = p_i$ which is indeed the identity -- I don't see where the difficulty is. – Glen_b -Reinstate Monica Jan 9 '17 at 3:43

1 Answer

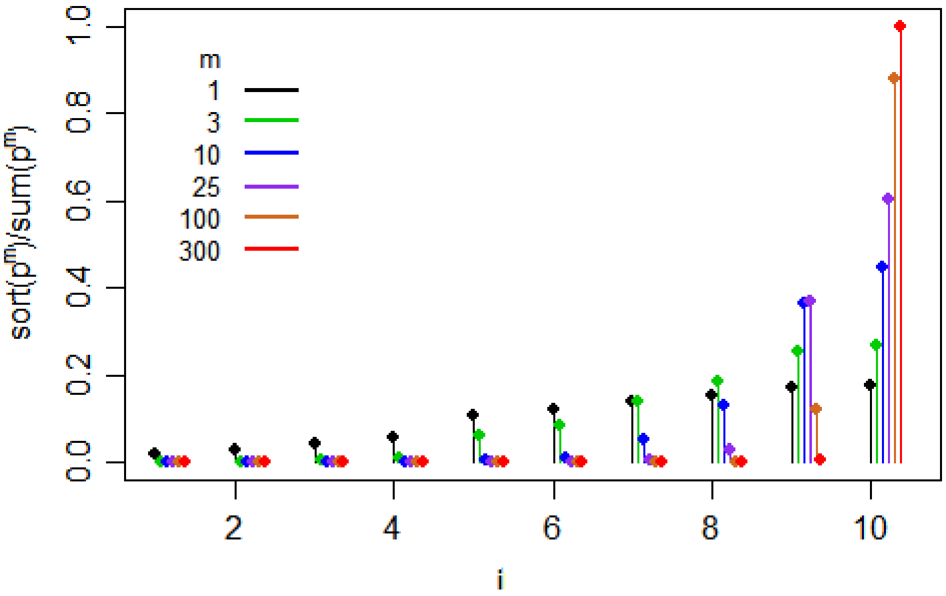
5

Note that we start with a set of probabilities which sum to 1. We define a function ($f(p)$ where the ith probability component $f_\tau(p)_i = \frac{p_i^{1/\tau}}{\sum_j p_j^{1/\tau}}$) in order to modify those probabilities as a function of temperature (for which the original probabilities have temperature $\tau = 1$). If we increase τ from 1, the transformed probabilities would become more nearly equal and if we decrease τ toward 0, the transformed probabilities become "shifted" toward the larger ones, away from the smaller ones.

For $\tau = 1$: $\sum p_j = 1$ so when $\tau = 1$ you have $f(p)_i = p_i$ which is indeed the identity.

For $\tau \to 0$ note that if you have two values of p, say $p_2 = kp_1$ (where $k < 1$) then $(p_2/p_1)^m = k^m$. Now let $m \to \infty$. We see that the ratio of a smaller p_i^m to a larger one will go to 0. Consequently, if you have a set of p's, then as m increases $p_i^m/p_{\text{largest}}^m$ will all vanish, apart from the p that is the largest (which is 1). Now if you replace p_{largest}^m on the denominator with the sum of the p_j^m you just make the denominator slightly larger (you're just adding terms that all go to 0). As a result, the scaled $f(p)_i = \frac{p_i^m}{\sum_j p_j^m}$'s will go to 0 on everything but the largest, which will go to 1. So if you select among the i's using those set of f values as probabilities, as $m \to \infty$, you'll select the largest. Now let $m = 1/\tau$ and let $\tau \to 0$ and you get $m \to \infty$ and it corresponds to selecting the argmax.

It's easy to see numerically. Here are 10 p_i values -- they're generated as uniform random values sorted into order and normalized to sum to 1 (shown in black below). Note that the second and third largest values are quite close to the largest (the second largest is really close to the largest in value). Then we increase the power in f progressively. The smaller terms rapidly decrease to a zero-share, while the largest term increases to 1. The ones close to the largest in size initially increase their share (they have k close to 1 in the above discussion, so their share initially stays close to the largest p, but the increasing power soon blows the largest one up much bigger than all the other terms)



On this particular example, by the time we get to $m = 300$ (i.e. $1/\tau = 300$), the probability of selecting the largest term is very close to 1. As τ goes closer to 0, $m = 1/\tau$ increases without limit, leaving only the argmax with any chance of being selected.

edited Jan 10 '17 at 23:11

answered Jan 10 '17 at 4:38

Glen_b -Reinstate Monica

229k 24 457 810