# Block 6: How do transformers do it?

*In the previous session, we explored evidence that neural networks represent belief state geometry of HMM processes. Based on your interest, we will now split into two tracks, each addressing a question:*

1. ***Sensors.*** *How might the transformer develop representations of the belief state?*
2. ***Actuators.*** *How might the transformer use belief state representations to influence downstream predictions?*

*The Sensors question is marginally better understood and will centre around reading Piotrowski et al., which partially answers this question for single-layer transformers trained on the Mess3 process. The Actuators question is relatively less explored, and is more conceptual.*

## Sensors

**Reading suggestion (45 minutes):**

- **Skim.** Sections 1 & 2.
- **Read.** Sections 3–5.
- **Evaluation.** Explain Figure 2.

**Overview.** Given the architectural constraints of a transformer block, how might belief states be constructed out of raw token embeddings? For the case of single-layer transformers trained on the Mess3 process, Piotrowski et al. identify an algorithm implemented by the attention mechanism that transforms input sequences into *constrained beliefs* (Equ. 6), which are then transformed into the corresponding belief state by the MLP layer.

The key ansatz is that the attention block outputs constrained beliefs. This observation is then used to anticipate structure in various components relevant to the attention mechanism:

- Attention pattern (see Equ. 13)
- OV-vectors (see Equ. 14)
- Embeddings (see Equ. 15)
- Number of attention heads (see Sec. 4.4.1)

These predictions are then empirically verified across four different Mess3 parameterisations.

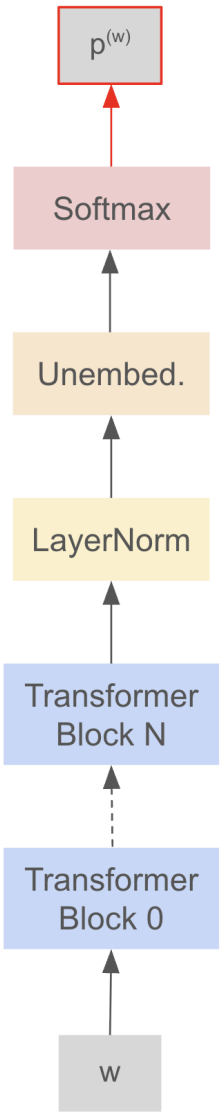
**Discussion (15 minutes).** Form groups of 3-4 and discuss aspects of the readings you found interesting/confusing. Here are some discussion prompts to consider:

- How is the constrained belief geometry related to the full belief geometry?

- What is the role of the MLP in transforming the constrained belief geometry into the full belief geometry?
- Do you expect the constrained belief state update formula to apply to:
    - Processes beyond Mess3?
    - Attention blocks with more than two heads?
    - Transformers that are deeper than a single layer?

(*Feel free to ignore these if you/your group is excited to explore other aspects of the reading.*)

# Actuators




**Background.** Suppose the transformer represents the belief state corresponding to a given input sequence at some residual stream position. Given the architectural constraints of the transformer consider the questions:

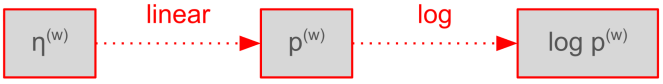
- How might the belief state be used to compute the corresponding next-token probability distribution?
- Which layer is the belief state most likely to be used by the transformer?

Recall that there always exists a linear map from the belief state to the next-token probabilities vector:


But given the transformer architecture, the next-token probabilities are produced by softmaxing logits (activations post-unembedding), which are related to the probabilities by

$$\text{logits}(w) = \log p^{(w)} + c_w \mathbf{1}$$

where $c_w$ is an arbitrary w-dependent constant and $\mathbf{1}$ is the all-ones vector of appropriate size. This observation motivates considering three types of features: *beliefs* ($\eta^{(w)}$), *next-token probabilities* ($p^{(w)}$) & *next-token log-probabilities* ($\log p^{(w)}$); and considering how they relate to each other. Without assumptions on the HMM process, they are related as follows:



**Exercises.**

1. Annotate the transformer diagram with where you would expect the trained models to represent: beliefs, next-token probabilities & next-token log-probabilities (if at all) under the following assumptions:
    a. No assumptions.
    b. There exists a linear map from next-token probabilities to next-token log-probabilities.
    c. There exists a linear map from next-token probabilities to beliefs.

    (*Keep in mind the expressiveness of the transformer components as you pass between beliefs / next-token probabilities / log next-token probabilities.*)
2. Design intervention experiments to gain evidence for your hypotheses.