World models and LLMs

VL Lab meeting

26/03/2025

Where is this coming from?

Imagine you need to decipher an unknown language. All you have is a (monolingual) dictionary.

The argument from grounding:

You can't just rely on language-to-language relationships to learn meaning.

The only reason cryptologists of ancient languages and secret codes seem to be able to successfully accomplish something very like this is that their efforts are grounded in a first language and in real world experience and knowledge

(Harnad 1990)

But grounding isn’t everything

It seems a bit extreme to argue that meaning representations are *always* grounded in external, non-linguistic data.

Here are some recent arguments:

Causal theories of reference

What do you know about Godel? (Or Peano, or Socrates…)

You can know what an expression refers to even if you don’t have direct experience of the referent. LLMs are exposed to linguistic forms, but also to their histories.

(e.g. Mandelkern and Linzen, 2024)

Sources of meaning beyond communication

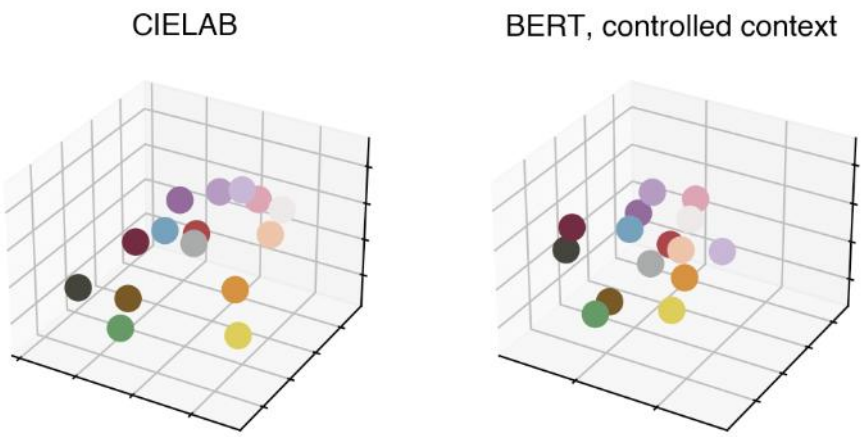
Symbols aren’t just defined by their use in communication (so not just what we intend them to mean). Meaning is also due to internal computation.

(e.g. Pavlick, 2023)

Today's paper

- Part of a recent(ish) trend to explore whether LLMs acquire some kind of "internal" representation of (aspects of) the (non-linguistic) world, based on their exposure to linguistic data.

- Many other examples, e.g. Abdou et al 2021



Main question in the current paper

- Does a LLM trained on symbol data using next-token prediction recover a world model corresponding to the sequence data?
- And is this world model an accurate one?

Case study: NYC taxi rides

→Sequences that capture trips in a specific, known environment

→Training data: turn-by-turn sequence dataset of taxi trips.

→Models (trained on up to 4.7B tokens): predict next turn

Data example

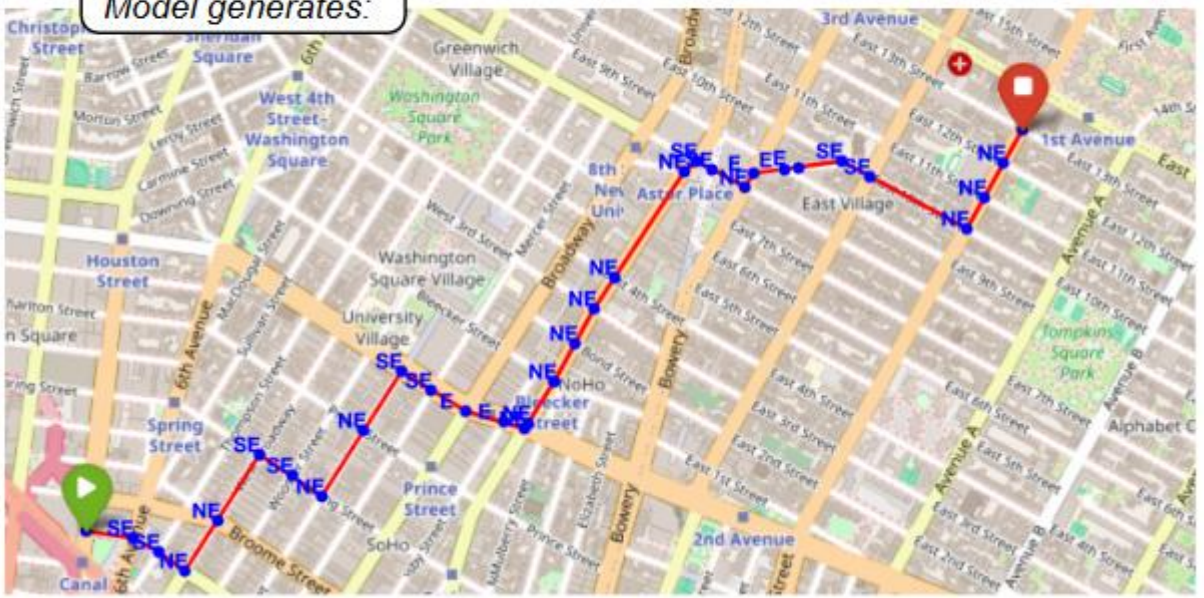
Training sequences

820 210 N E E E SE W W N SE N end
193 450 E E E N E S E end
301 592 N N N SE S S S S S end
…

Contexts for evaluation

110 244
850 820
592 301
…

Model generates:



How to evaluate accuracy of a world model?

Accuracy of prediction

Does the model accurately predict the sequence to get from A to B?

This is easy…

Accuracy of underlying model

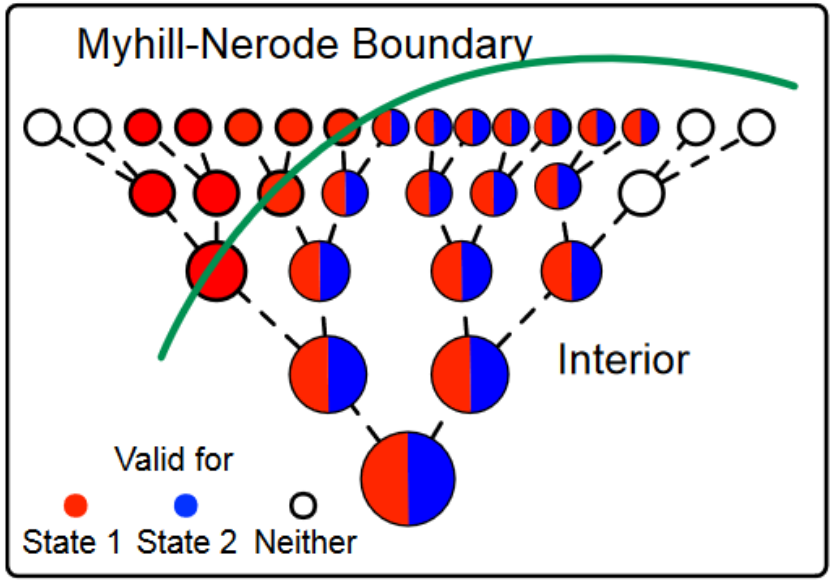
Much harder!

Possible for the model to make the right predictions, but have the wrong, internalised model.

The world model as DFSA

For domains like taxi rides, we can model the world as a deterministic finite state automaton.

- Assume a regular language *L*
- Let *x, y* be strings in *L*.
- Let *z* be a string s.t. either *xz* is in *L,* or *yz* is in *L*, but not both
  - (Intuition: *z* is a legal extension of only one of those strings)
- Conversely, if *xz* and *yz* are both in L, or both not in L, then x and y are indistinguishable in L.
- So this gives us an equivalence relation over strings in *L*



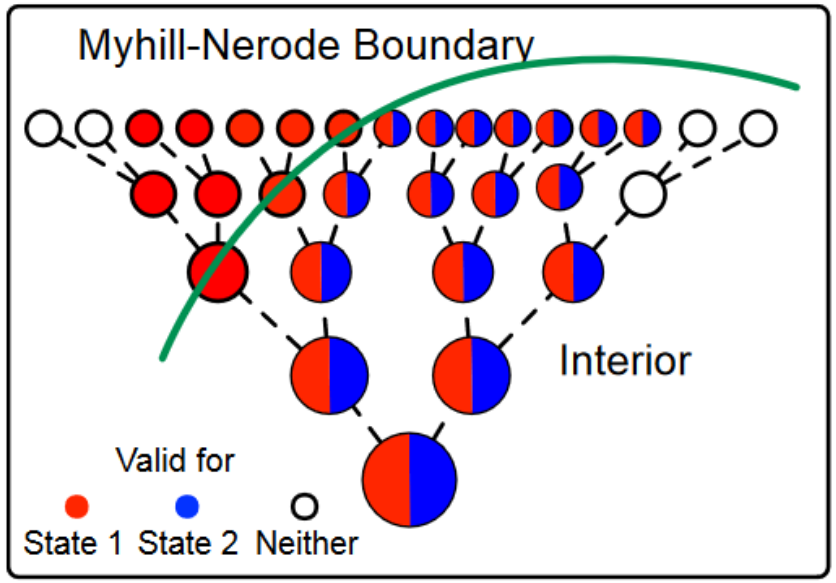
Myhill-Nerode Theorem:

Every pair of distinct states can be distinguished by some sequence which is admitted by one state but not the other.

L is regular if and only if the equivalence relation partitions *L* into finitely many disjoint sets.

For a minimal DFA, if we start at two distinct states, then the sequences accepted by the DFA are distinct. **However, the sequences can overlap.**

The MN interior and the MN boundary



$$\mathrm{MNI}^W(q_1, q_2) = \{s \in \Sigma^* \text{ s.t. } s \in L^W(q_1) \cap L^W(q_2)\}.$$

$$\mathrm{MNB}^W(q_1, q_2) = \{s = a_1 a_2 .. a_k \mid s \in L^W(q_1) \setminus L^W(q_2) \text{ and } \forall j < k : a_1 .. a_j \in \mathrm{MNI}^W(q_1, q_2)\}.$$

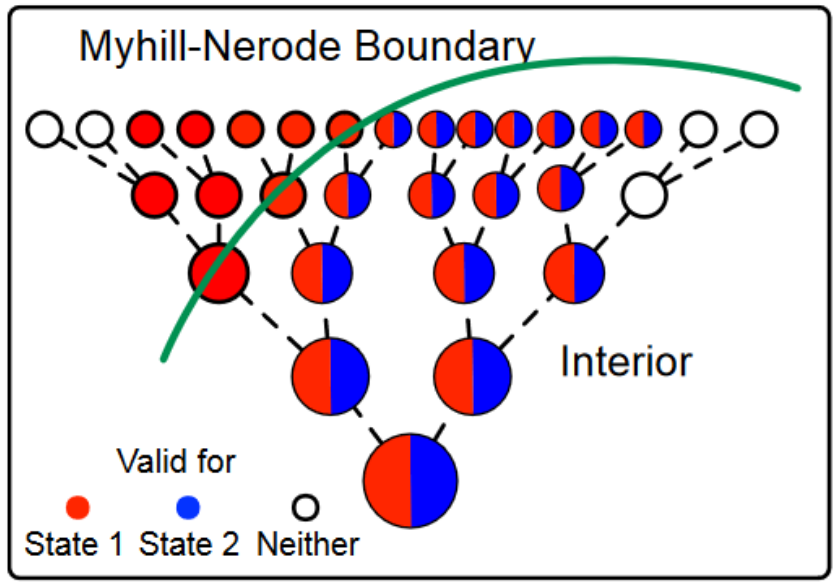
The boundary is defined by states which extend sequence q1, but not q2. But below the boundary there is overlap between q1 and q2 (which is the interior).

Boundary recall/precision for a LLM

A model *m()* trained on sequences of *L* has an implicit MNB between any two sequences s1, s2

The set of minimal sequences accepted by the model conditioned on s1, but not s2.

MNB Recall and precision are obtained by comparing the model's implied MNB^m to the actual one MNB^w.



$$\frac{|\mathrm{MNB}^W(q_1,q_2) \cap (m(s_1) \setminus m(s_2))|}{|\mathrm{MNB}^W(q_1,q_2)|}$$

$$\frac{|\mathrm{MNB}^m(s_1,s_2) \cap (L^W(q_1) \setminus L^W(q_2))|}{|\mathrm{MNB}^m(s_1,s_2)|}$$

Metrics

Sequence compression

- Observe that a DFA provides multiple ways to arrive at the same state.
- Let q1 = q2 be two equal states.
- Here there is no true MN-boundary.
- Sample many such state pairs and ask: does a model recognise that the two sequences correspond to the same state?

Sequence distinction

- Sample distinct state pairs, q1 =/= q2
- Here there is a true MN-boundary
- Does the model recover it?

Back to the taxi rides

Data:

12.6m taxi rides

Each sequence is a string in the DFA representing NYC:

- Nodes are intersections
- Edges are streets, weighted by distance
- Edge labels are directions

Models:

- Predict a traversal: given origin and destination, predict the directions
- All models trained from scratch.

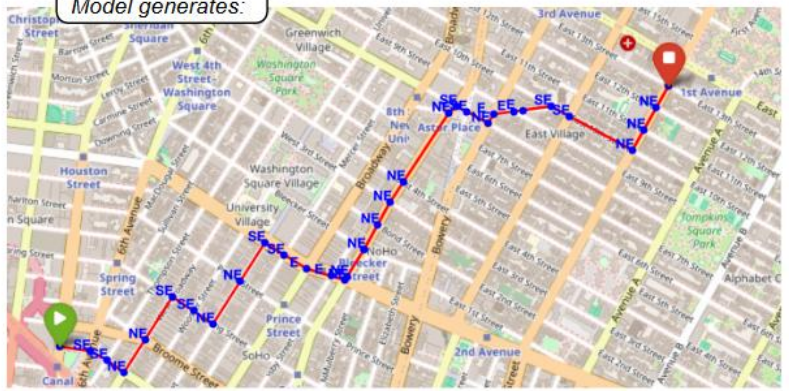
Training sequences

820 210 N E E E SE W W N SE N end
193 450 E E E N E S E end
301 592 N N N SE S S S S S end
…

Contexts for evaluation

110 244
850 820
592 301
…

Model generates:



- Shortest paths: find the shortest path between two nodes
- Noisy shortest path: find the shortest path, but perturbing the edge weights of the underlying graph (to mimic traffic conditions, e.g.)
- Random traversals.

Results

	Existing metrics		Proposed metrics		
	Next-token test	Current state probe	Compression precision	Distinction precision	Distinction recall
Untrained transformer	0.03 (0.00)	0.10 (0.00)	0.00 (0.00)	0.00 (0.00)	0.01 (0.00)
Shortest paths	1.00 (0.00)	0.91 (0.00)	0.19 (0.01)	0.36 (0.01)	0.26 (0.01)
Noisy shortest paths	1.00 (0.00)	0.90 (0.00)	0.07 (0.01)	0.36 (0.01)	0.25 (0.01)
Random walks	1.00 (0.00)	0.99 (0.00)	0.68 (0.02)	0.99 (0.00)	1.00 (0.00)
True world model	1.00	—	1.00	1.00	1.00

Does the model predict the correct next turn?

Standard probe: check if the transformer representation predicts the current intersection

Model prediction vs actual world

(a) World model **(b)** World model with noise **(c)** Transformer

Figure 3: Reconstructed maps of Manhattan from sequences produced by three models: the true world model (left), the true world model corrupted with noise (middle), and a transformer trained on random walks (right). Edges exit nodes in their specified cardinal direction. In the zoomed-in images, edges belonging to the true graph are black and false edges added by the reconstruction algorithm are red. We host interactive reconstructed maps from transformers at the following links: shortest paths, noisy shortest paths, and random walks.

Some takeaways and discussion points

- Models may make correct predictions, but still learn an incorrect underlying model of the “world”.

- What does this imply for benchmarks?
  - E.g. is it possible for a model to perform well on a VL benchmark, but still have incorrect representations?

- What other domains could this type of work be applied to?
  - Here, relatively simple world model (DFA) with well-understood formal properties.