Masked Hard Attention Transformers Recognize Exactly the Star-Free Languages

Andy Yang (University of Notre Dame, USA)
David Chiang (University of Notre Dame, USA)
Dana Angluin (Yale University, USA)

11 Nov 2024

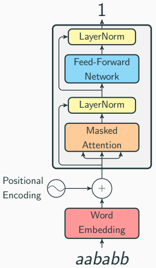
Background

Over inputs of unbounded length, what problems can (and can't) transformers solve?

Over inputs of unbounded length, what problems can (and can't) transformers solve?

and how can we prove it?

Expressivity: Transformer Encoders and Formal Models



$\longleftrightarrow$

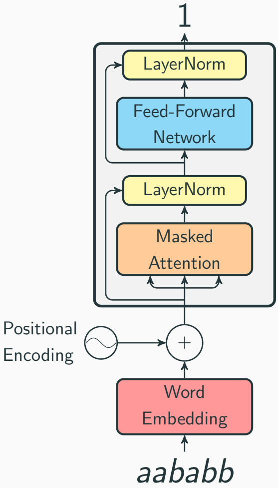
$$\forall i.Q_a(i)$$
$$\forall i.(Q_a(i) \to \exists j.(i < j \land Q_b(j)))$$
etc.

What formal languages are recognized by transformer encoders?

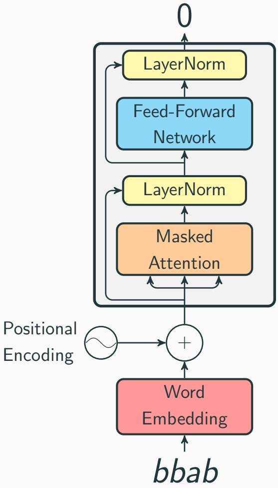
What formal languages are defined by logical formulas?

Masked Hard Attention Transformers

Transformer Encoders

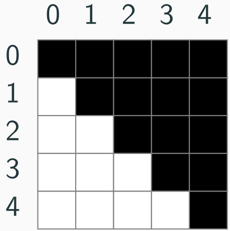


Transformer Encoders



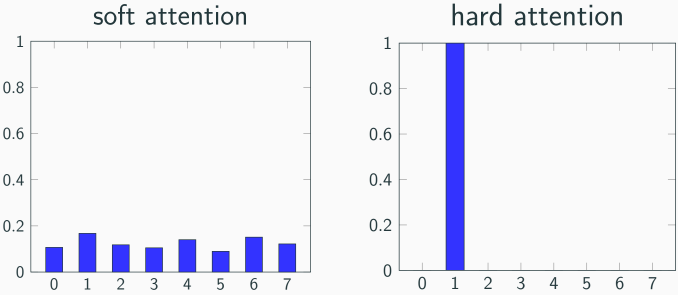
Strict Future Masking

Each position can only attend to positions strictly to the left



Leftmost/Rightmost Unique Hard Attention

Focus all attention on a single position - find maximum score and break ties to the left/right



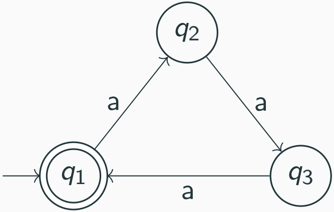
Main Result

Masked Hard-attention Transformers $\longleftrightarrow$ **B-RASP** $\longleftrightarrow$ LTL $\longleftrightarrow$ Star-Free Languages

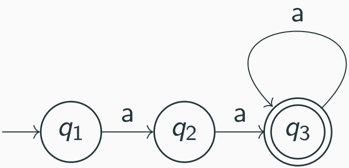
Star-Free Languages

What are Star-Free Languages?

Periodic (regular)



Aperiodic (star-free)



Examples of Star-Free Languages

Dyck-1 of Depth 2	(matched parentheses 2 deep)
$(ab)^*$	(repeated ab's)
$\Sigma^* aa \Sigma^*$	(strings that contain substring aa)
$\Sigma^* ab\, (\Sigma \setminus \{a\})^*\, ab$	(building block of induction heads)

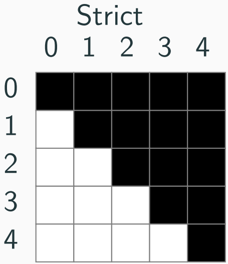
Main Result

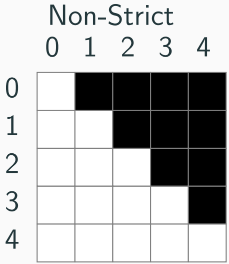
Masked Hard-attention Transformers $\longleftrightarrow$ **B-RASP** $\longleftrightarrow$ LTL $\longleftrightarrow$ Star-Free Languages

Corollaries

How does using strict vs non-strict masking affect expressive power?

Non-Strictness





Strict masking is more expressive

Stutter-Invariant Star-Free Languages

Theorem

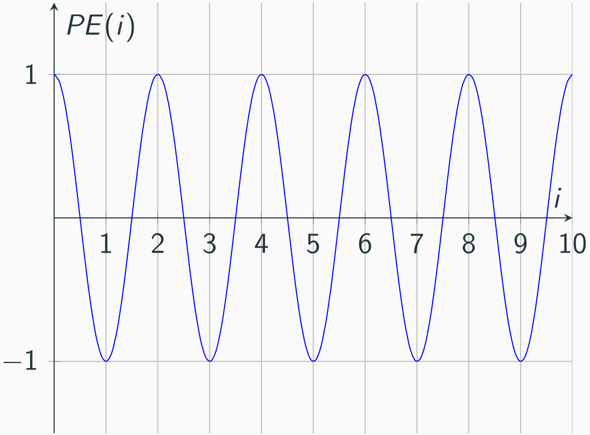
Masked hard-attention transformers with only non-strict masking recognize exactly the stutter-invariant star-free languages.

For instance

- $(ab)^*$ is not stutter invariant
- $(a^* b^*)^*$ is stutter invariant

How do positional embeddings affect expressive power?

Positional Embeddings



Using sinusoidal position embeddings is more expressive

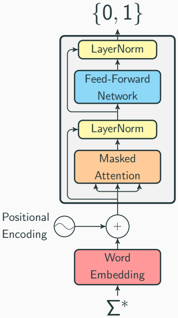
Transformers and Regular Languages in AC^0

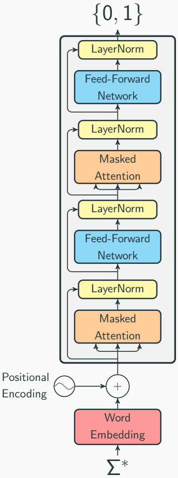
Theorem

Masked hard-attention transformers with rational sinusoidal positional embeddings recognize exactly the regular languages in AC^0

How does adding more layers affect expressive power?

Transformer Depth





Adding more layers is more expressive

Depth Hierarchy

Theorem

Masked hard-attention transformers with $k+1$ layers are strictly more expressive than masked hard-attention transformers with k layers

It requires $k+1$ layers to recognize the language STAIR_{k+1}

Parting Notes

Limitations

- Hard attention results may not apply to softmax attention
- We don't consider autoregressive language modeling
- No claims on empirical learnability

Takeaway

Formal language theory can quite effectively explain the computational behavior of masked-hard attention transformers