Probability Distributions Computed by Autoregressive Transformers

Andy Yang[1] Anej Svete[2] Jiaoda Li[2] Anthony Widjaja Lin[3,4] Jonathan Rawski[5] Ryan Cotterell[2] David Chiang[1]

[1]University of Notre Dame [2]ETH Zürich [3]Max-Planck Institute for Software Systems [4]University of Kaiserslautern-Landau [5]San Jose State University

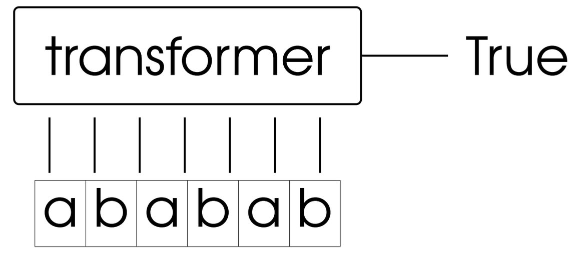
Main Contribution

Previous work: characterize what functions transformers can compute as language *classifiers*

Our work: characterize what transformers compute as *autoregressive language models* – and prove these settings differ

Previously: Language Classification

Previous work considered transformers as classifiers: accept or reject a string in one forward pass



Hard-attention transformers can express functions definable in *linear temporal logic*

UHATs = TL[$\mathbf{S}$] Yang et al. 2025

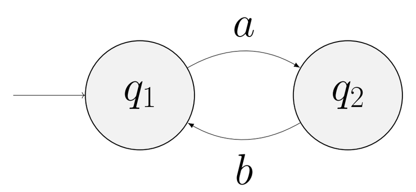
Finite-precision soft-attention transformers can express functions definable in a restriction of linear temporal logic

SMATs = TL[$\mathbf{P}$] Li & Cotterell 2025

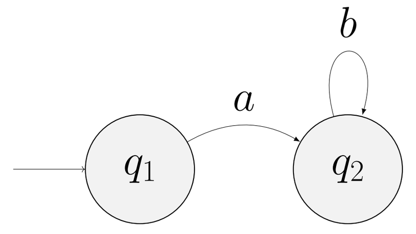
But language models are *autoregressors* and not *classifiers*! How do these results carry over?

Logic and Automata

We connect classifiers and autoregressors using logic and automata



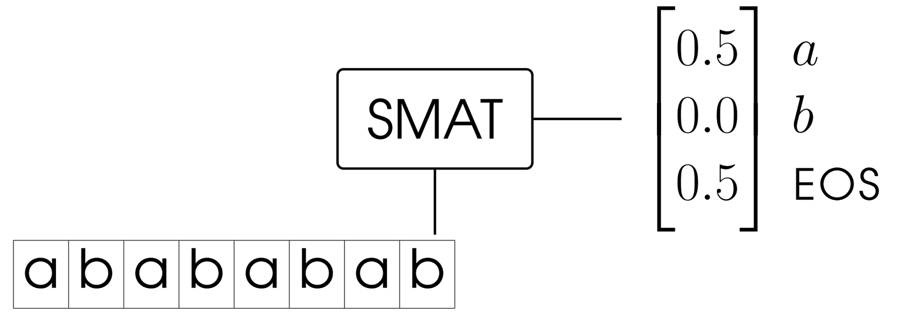
TL[$\mathbf{S}$] formulas are equivalent to counter-free automata; no modular counting



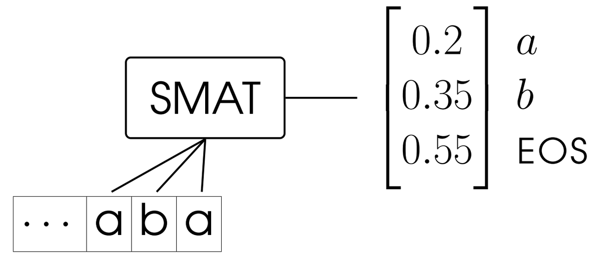
TL[$\mathbf{P}$] formulas are equivalent to partially-ordered automata; only self-loops

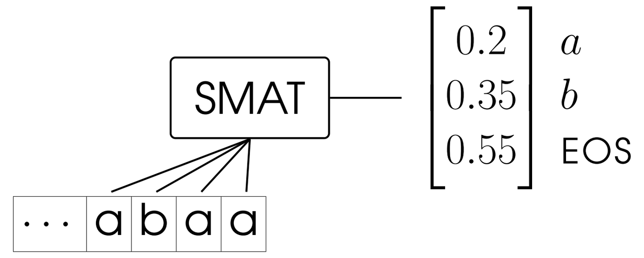
We Tease Apart Classifiers and Autoregressors

A SMAT classifier cannot recognize $(ab)^*$, but a SMAT autoregressor can! Simple construction: if a, predict b. if b, predict a or EOS.

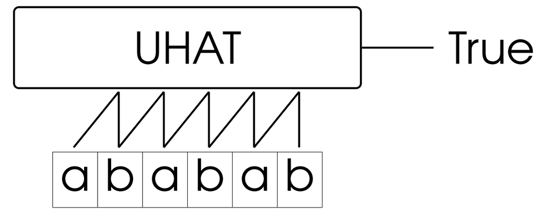


However, we can prove every SMAT autoregressor cannot recognize $(aab)^*$.





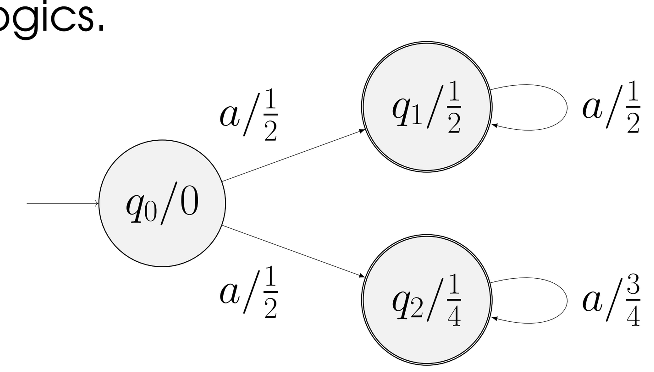
Intuition: on sufficiently long strings, the output activations will converge for any string of the form $\ldots aba$ and $\ldots abaa$. In the Boolean-weighted case, UHAT autoregressors and UHAT classifiers are actually equivalent.



The UHAT attends to each predecessor position, and simulates the next-token predictions using a prefix construction. Then, the UHAT checks at every position. that each such prediction matches the current symbol

We Characterize the Real-Weighted Case

For the first time, we can exactly characterize the *probability distributions* definable by transformer LMs in terms of weighted automata and logics.



UHAT LMs = Weighted cfDFAs

SMAT LMs = Weighted TL[$\mathbf{P}$] autoregressors

Boolean-Weighted Separations

Classifiers		Autoregressors
UHAT	$=$	UHAT
$\subsetneq$		$\subsetneq$
SMAT	$\subsetneq$	SMAT

Autoregression...

($=$) *cannot increase* Boolean-weighted UHAT expressivity

($\uparrow$) *increases* Boolean-weighted SMAT expressivity

Real-Weighted Separations

Classifiers		Autoregressors
UHAT	$\neq$	UHAT
$\subsetneq$		$\subsetneq$
SMAT	$\neq$	SMAT

Autoregression...

($\neq$) *can increase or decrease* real-weighted UHAT expressivity

($\neq$) *can increase or decrease* real-weighted SMAT expressivity

Future Work

Our framework enables us to apply our understanding of transformer classifiers onto autoregressive transformer language models. Future work should:

- Extend to log-precision transformers: hard attention/finite precision are restrictive
- Handle CoT and positional encodings: these are important practical settings
- Use our framework to explain empirical findings: sometimes transformers as language models behave differently than the literature on classifiers may predict, and our results say why