Language Modeling by Language Models with Genesys

Junyan Cheng*, Peter Clark, Kyle Richardson

Allen Institute for AI, Dartmouth College*

Motivation

Autonomous scientific discovery

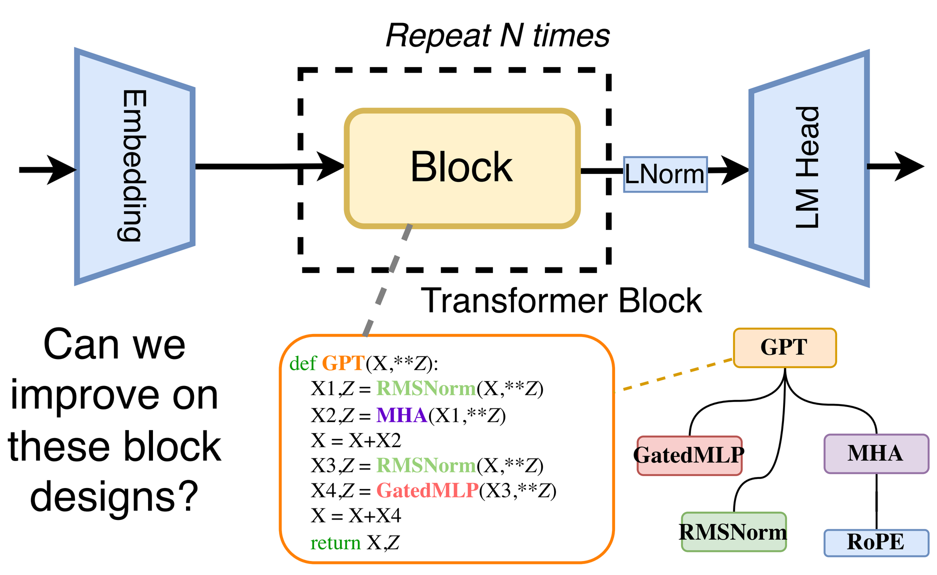
- Much recent excitement, though unclear goal**s** and lack of standard discovery tasks to hill-climb on.
- Little understanding on how to effectively design and build large-scale efficient discovery systems.

This work

- Look at challenging discovery problems in language model research, **architecture design.**
- Propose **algorithmic framework** for efficient discovery, proof-of-concept system called **Genesys.**

Language Model Architecture Discovery: What?

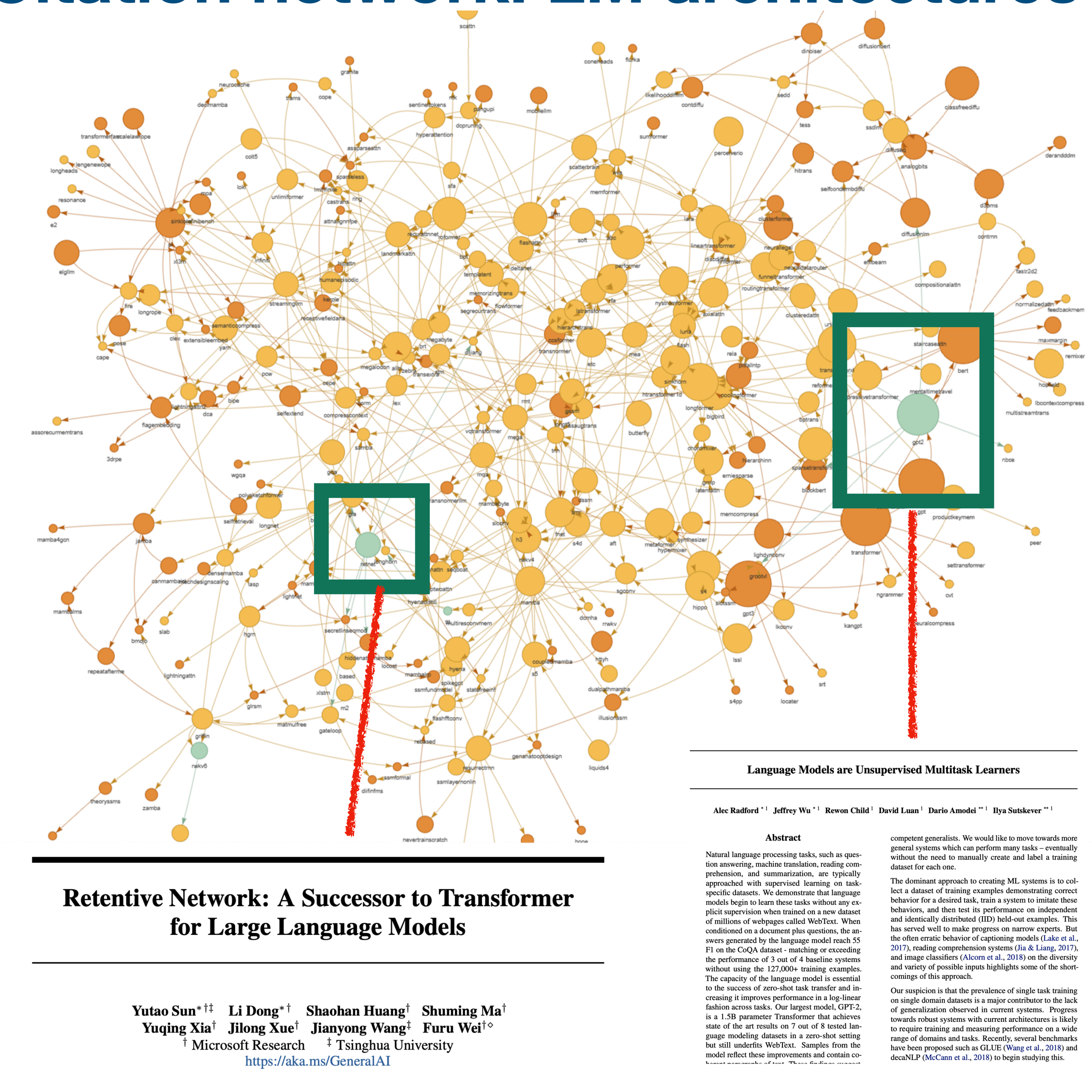
Autoregressive models



➢ **Block designs** dictate how information flows in neural architectures.

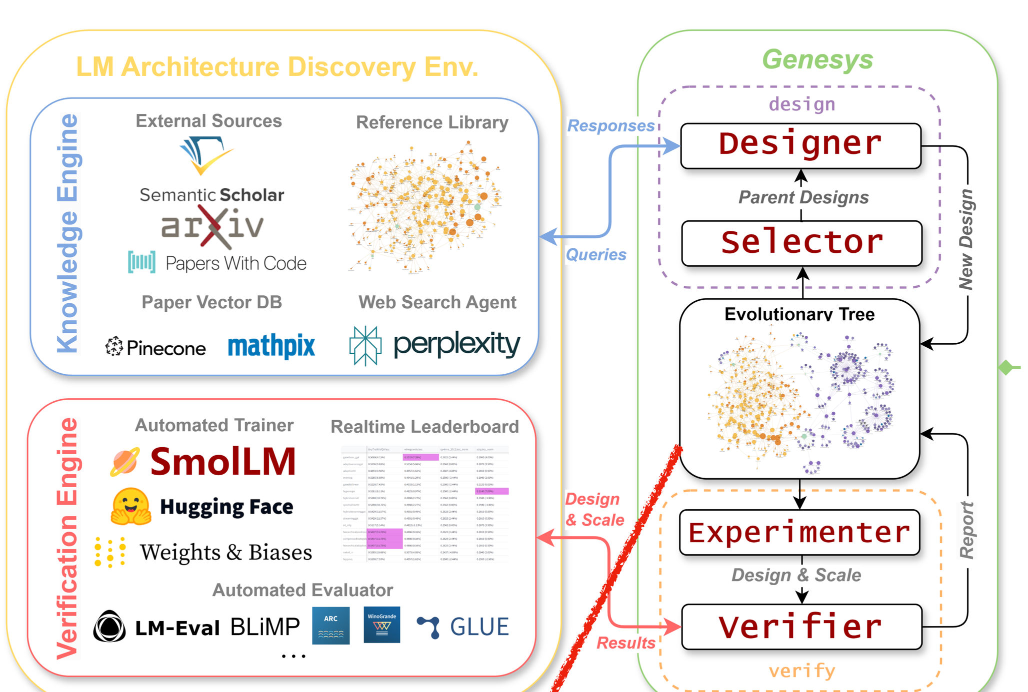
➢ While the transformer architecture remains the de-facto standard, large literature on alternative architectures. **Can we automatically discover new designs building from this literature?**

Citation network: LM architectures



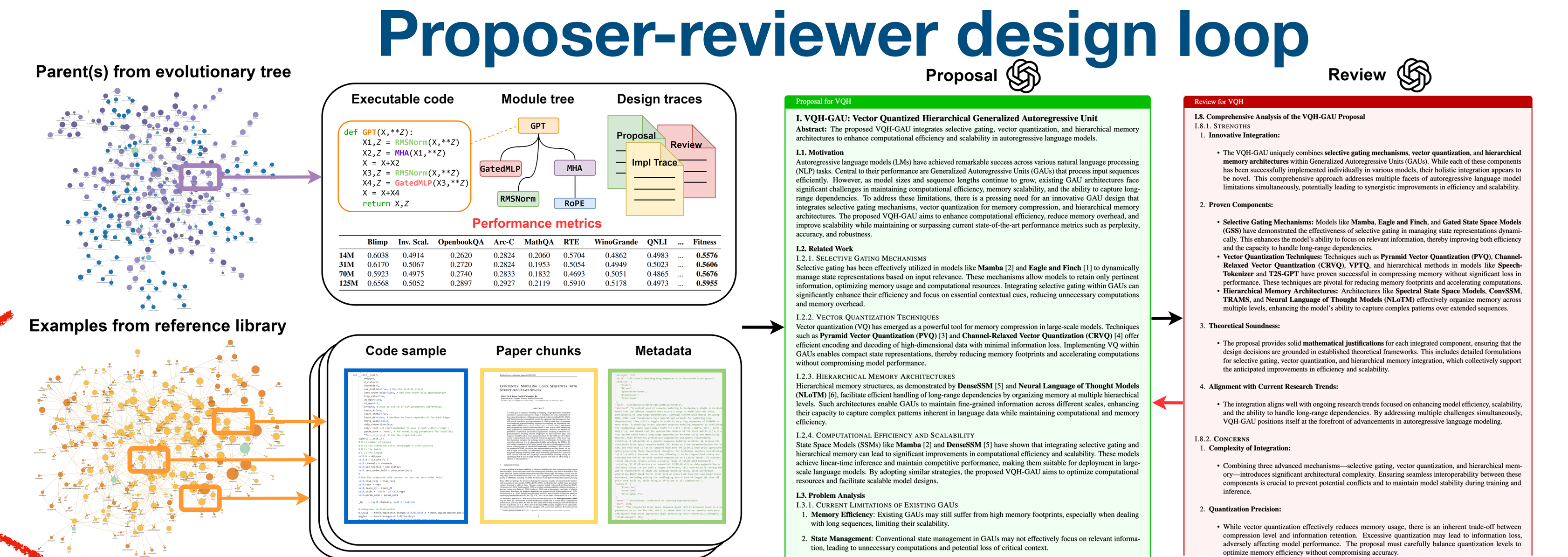
The Genesys system for architecture discovery

Core Genesys System

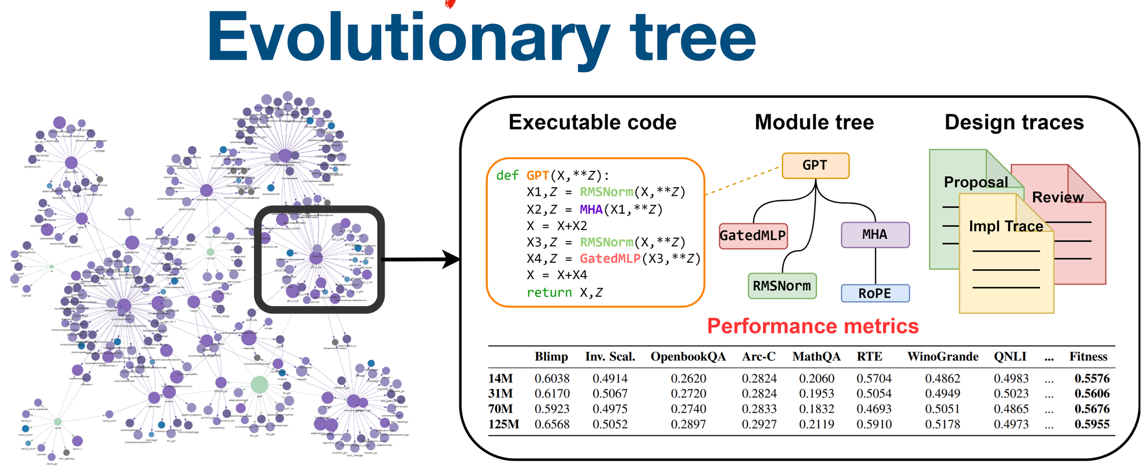



```
async def genesys_evolve(EvoTree, KE, VE):
    """Run distributed discovery loop"""
    while budget:
        async design(EvoTree,KE)
        async verify(EvoTree,VE)

async def design(EvoTree, KE):
    """Select parent designs to improve,
    produce new design and add to tree"""
    parents, refs = Selector(EvoTree)
    design = Designer(parents,refs,KE)
    async EvoTree.update(design)

async def verify(EvoTree, VE):
    """Select a design and scale to verify,
    run experiments and produce a report"""
    design, scale = Experimenter(EvoTree)
    report = Verifier(design,scale,VE)
    async EvoTree.update(design.report)
```

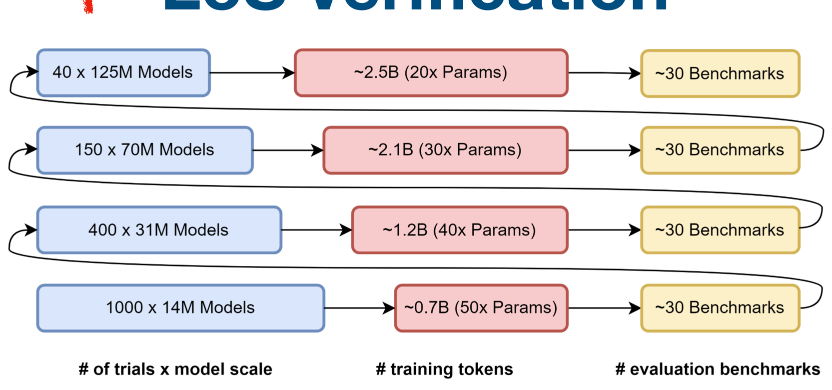
Proposer-reviewer design loop



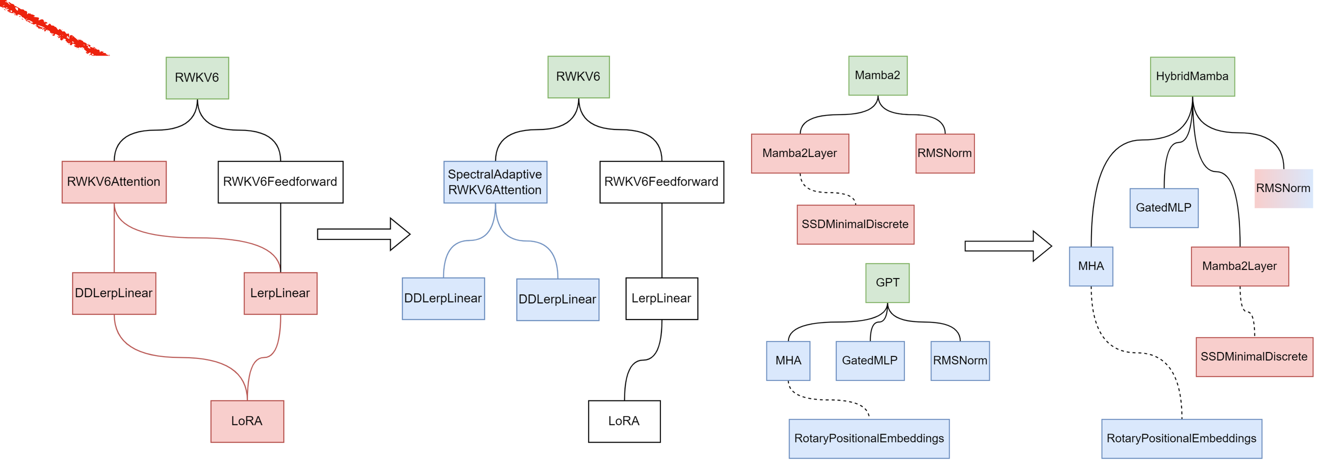
Evolutionary tree



LoS verification



Unit-based design implementation



(mutation) (crossover)

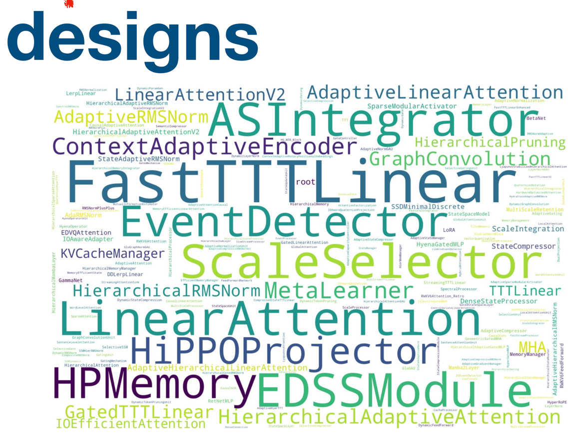
Have we made any discoveries yet?

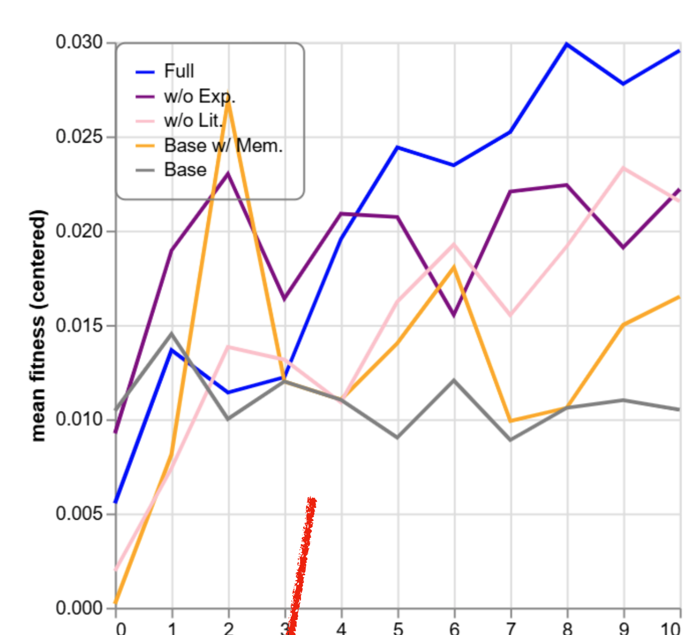
	Blimp	Wnli	RTE	WG	CoLA	SST2	WSC	IS	Mrpc	avg.
Random	69.75	43.66	52.71	48.78	50.00	49.08	49.82	50.03	31.62	49.49
GPT	92.70	60.56	62.80	52.17	*53.24*	54.13	56.76	55.31	68.38	61.78
Mamba2	83.22	63.38	**63.88**	51.22	55.94	56.58	57.12	53.85	67.89	61.45
RWKV7	88.76	61.97	60.21	*49.80*	54.25	55.32	54.57	**57.00**	68.38	61.14
RetNet	85.16	61.97	61.35	50.51	**56.29**	55.43	56.03	54.95	*56.37*	59.78
TTT	86.13	63.38	*55.23*	50.75	55.55	56.35	54.93	55.31	59.80	59.71
VQH	**94.37**	59.15	59.91	50.28	54.25	53.56	*53.83*	*49.45*	56.62	59.05
HMamba	83.74	64.79	61.35	**53.59**	54.69	**57.04**	56.40	54.58	59.31	60.61
Geogate	90.95	59.15	61.35	52.72	54.25	55.32	**58.96**	54.95	68.63	**61.81**
Hippovq	87.96	*50.70*	59.91	50.28	54.25	55.73	*53.83*	55.68	**69.88**	59.80
SRN	*80.83*	**65.52**	59.55	50.75	54.45	*52.98*	56.03	54.95	61.03	*59.57*

Table 3: Performance of human designs and discovered models on various Benchmarks (350M Parameters, 50B Tokens). Metrics indicate accuracy percentages. Bold and underlined denotes the top and second best, italics denoting worst.

new designs

➢ Our system produces highly innovative new block designs that are competitive with state-of-the-art human architectures designs, **showing the feasibility of automated discovery in this domain.**





system stability

	Valid	Attempts	Costs	LFC
Full	92%	2.6 (±1.1)	15.0 (±18.5)	181 (±44)
No FF	73%	3.0 (±1.7)	7.9 (±7.1)	75 (±29)
No Pl.	91%	2.6 (±1.1)	16.0 (±20.8)	218 (±69)
No Ob.	89%	2.6 (±1.1)	12.1 (±20.1)	211 (±67)
No SC	30%	2.4 (±1.0)	2.9 (±4.7)	167 (±33)
Simple	6%	1.1 (±0.2)	0.3 (±0.3)	49 (±15)
Library	-	-	-	220 (±136)

Table 3. Agent benchmark results. Bold and underlined denotes the top and second best. "Library" stands for our reference library with 180 designs providing core block code.

successful code generation rates

➢ System design decisions can be justified both empirically *(e.g., **improved system stability, effective code generation**)* as well as algorithmically ***(exponentially improved bounds on rates of generating correct code via Viterbi-style search)***.