The linguistic power of LLMs

András Kornai

Algebra Department, Budapest University of Technology and Economics
and
SZTAKI Institute of Computer Science

November 20 2025

Plan of the talk

- Formal language theory: what is it and why it matters
- Classic results on human languages and formal grammars
- Modern results on formal grammars and LLMs
- The humanities and the sciences: together again?

Formal grammar refresher

Formal languages

Let Σ be a finite set of symbols (node labels). We divide this in two parts, $N \subset \Sigma$ is called the *nonterminal vocabulary*, which includes a distinguished *start symbol* S, and $\Sigma \setminus N$ is called the *terminal vocabulary*. With the concatenation operation, the free monoid over Σ is denoted Σ^*, we call the elements 'strings' or 'words' or 'sentential forms' depending on the phase of the moon. There is also a distinguished *empty string* (unit element) denoted λ. Sets $L \subset \Sigma^*$ are called *formal languages*

We can specify formal languages by several methods:

- By properties
- By acceptors
- By generators
- By operations on already given languages

Operations on languages

- Since languages are sets, we have the Boolean operations
- We can also concatenate languages L and M producing $LM = \{lm : l \in L \wedge m \in M\}$
- Kleene $^+$ and *
- Homomorphism, inverse homomorphism, projections
- More tricky operations: reversal, shuffle, Parikh mapping

Automata, grammars

- **Automata:** The concept of an automaton. Finite automata. Sequential behavior of automata. Nondeterministic automata. Homomorphism, isomorphism. Congruences of automata. Characteristic semigroup. Automaton mappings.
- **Language Recognition in Automata:** Languages recognizable by automata without output symbols. Semigroup-theoretic characterization. Syntactic semigroup. Equivalence of recognizer automata. Languages recognizable by nondeterministic automata. Closure properties.
- **Generative Grammars:** Chomsky language classes. Standard grammars. Closure properties. Context-free grammars.
- **Context-Free Languages:** Chomsky normal form. Bar-Hillel lemma. Reduced grammars. Leftmost derivations. Recursive variables. Greibach normal form. Regular context-free languages. Homomorphic characterization. Context-free expressions. Parikh functions.

The (extended) Chomsky hierarchy

- $\mathcal{L}_{-1}$: any language (not necessarily recursively enumerable). Over a binary alphabet these correspond to non-computable numbers: we know (e.g. by a cardinality argument) that there are many, but we can't pin them down
- $\mathcal{L}_0$: recursively enumerable. This implies a one-sided decision (e.g. by listing by a TM). Analogous to computable numbers
- $\mathcal{L}_{0.5}$: recursive (iff both L and $\Sigma^* \setminus L$ are r.e.)
- $\mathcal{L}_1$: context-sensitive. Can be decided by TM limited to the tape the size of the input. Can be generated by CSGs
- $\mathcal{L}_{1.5}$: mildly context-sensitive. Can be decided in polynomial time, see Joshi, Vijay-Shanker, and Weir, 1990

The (extended) Chomsky hierarchy, cont'd

- $\mathcal{L}_2$: context free. Can be generated by CFGs. Can be accepted by pushdown automata. Analogous to algebraic numbers
- $\mathcal{L}_3$: finite state (regular). Can be accepted by finite automata. Can be generated by regular expressions. Has a finite syntactic monoid. Analogous to rational numbers.
- $\mathcal{L}_4$: counter-free (non-counting). Has an aperiodic syntactic monoid. Star-free regexps. See McNaughton and Papert, 1971; Kornai, 1985
- $\mathcal{L}_{4.5}$: Subregular. See Heinz, 2018
- $\mathcal{L}_5$: Finite

Early generative work (1956–1973)

- Emphasis on syntax (traditionally hugely underresearched)
- The key technical tool was the *transformation* (tree-to-tree mapping)
- Phonology was handled by CSGs Chomsky and Halle, 1968
- Everything else by Transformational Grammars (TG) Chomsky, 1965
- Morphology (structure of words) and discourse (structure beyond the sentence) pretty much ignored
- Pretend-meaning *Logical Form* by tree structure

Persistent issues with formal language theory

- Strings are likely not the most interesting/revealing data structures to consider
- Phonology employs several, partially synchronized strings (autosegmental representations, see Goldsmith, 1990; Kornai, 1995)
- Computational linguists prefer *weighted languages* (weights from a semiring, typically the *tropical semiring* modeled on log probabilities)
- Trees may not be the best descriptor of structures (other descriptors added e.g. in Lexical-Functional Grammar, see Bresnan, 2001)
- This is all about form (phonology, syntax), but what about meaning?

The pivot from formal language theory

- Salomaa, 1971 and independently Peters and Ritchie, 1973 proves that TG is Turing-powered (every Type 0 lg has a TG)
- Chomsky (and most others) want a theory of grammar that is specific to linguistic phenomena. This requires making TG *less* powerful.
- Ross, 1967 starts to work in this direction, by adopting a theory where transformations manipulate meaing representations (LF trees) until they become syntactic trees. Chomsky prefers a different model where the initial (deep structure) trees are manipulated both towards meaning representation (LF) and towards syntactic representation, see Chomsky, 1970; Chomsky, 1971. Thus we had the *linguistic wars*, a story told from Chomsky's side by Harris, 1995 and from the other side by Goldsmith and Huck, 2013.

Less powerful transformations

- Having (in a sociological sense) won the linguistic wars, Chomsky, 1973 also starts working towards less powerful transformations
- There is a whole evolution of theories starting with Government-Binding (Chomsky, 1981) going through early minimalist theory (Chomsky, 1995) and leading to the current 'New Mininalism' Marcolli, Chomsky, and Berwick, 2023; Marcolli, Berwick, and Chomsky, 2023a; Marcolli, Berwick, and Chomsky, 2023b; Chomsky et al., 2023 using Hopf Algebras
- The architecture is based on a revised theory of signs, which have three parts: a meaning (some kind of trees), a syntax (other kinds of trees), and a form (phonology, not discussed in any detail in old or new MT)
- These units are what Harley, 2014 calls *roots*
- Besides roots, *features* are also used

Climbing up the Chomsky hierarchy

- Chomsky, 1956: NLs must be $\mathcal{L}_1$, but more complex (transformational) is preferred, because "A weaker, but perfectly sufficient demonstration of inadequacy would be to show that the theory can apply only clumsily; that is, to show that any grammar that can be constructed in terms of this theory will be extremely complex, *ad hoc*, and 'unrevealing', that certain very simple ways of describing grammatical sentences cannot be accommodated within the associated forms of grammar, and that certain fundamental formal properties of natural language cannot be utilized to simplify grammars"
- Pullum and Gazdar, 1982 Need not be more complex than $\mathcal{L}_2$
- Current consensus in syntax (Culy, 1985; Shieber, 1985; Joshi, Vijay-Shanker, and Weir, 1990): $\mathcal{L}_{1.5}$
- Current consensus in phonology (Kornai, 1995; Karttunen, 1998; Kornai, 2009; Karttunen and Beesley, 2003; Yli-Jyrä, 2018): $\mathcal{L}_4$

Where is morphology?

- Studying the structure of words was the mainstay of classical linguistics (Sanskrit, Greek, Latin, Arabic grammarians going back to 2,500 years)
- The structuralists were no less enthusiastic, because (i) segmentation into words (items that tolerate no pause in the middle) is readily performed by native speakers, even in languages where sentences are often just one word as in Hungarian *Meglátogatnálak* 'I would like to visit you'; (ii) Much of phonology is morphophonology (phonological changes driven by putting morphemes next to one another)
- *Morphotactics*, as analyzed by the structuralists (Nida, 1949; Harris, 1951) is easy to implement by FSA $\mathcal{L}_3$

Now how about semantics?

- Kornai, 1987 argued for $\mathcal{L}_3$ and virtually everything in the GOFAI tradition fits there. The exception is arithmetic, which requires $\mathcal{L}_1$
- Conjecture: FOL requires $\mathcal{L}_{1.5}$ (Marsh and Partee, 1984). Fact: even with full-blown variable binding, $\mathcal{L}_1$ is sufficient
- All purported evidence against $\mathcal{L}_{1.5} - \mathcal{L}_1$ comes from arithmetic (see e.g. Radzinski, 1991 on Chinese number names) and multiple variable binding (Georgian Suffixaufnahme, see Michaelis and Kracht, 1997; Kracht, 2002).

Learnability of formal languages

- In general, learnability is undecidable. This was established in the discrete case in Gold, 1967: for example, given arbitrarily large corpora, no $\mathcal{L}_1, \mathcal{L}_2$, or $\mathcal{L}_3$ grammar can be learned for the language these are samples from!
- In the continuous case this is just as hard: Suppose that $X_1, \ldots, X_n$ is an i.i.d. sample from an unknown distribution P, where P is a distribution on [0, 1] with finite support. Does there exist an algorithm that, with arbitrarily high probability, returns a finite subset of [0, 1] that has an arbitrarily high P-measure? (You may choose the sample size depending on how high the probability and the P-measure is required to be). According to Ben-David et al., 2019 this is independent of ZFC!

Research direction

- Has flipped from top-down to bottom-up (Kornai, 2021)
- This has happened precisely because in theory learning is hard, yet in practice learning (both human and machine) works well, this is The Bitter Lesson (Sutton, 2019)
- Powered by approximation theorems such as (Cybenko, 1989) neural nets are universal approximators
- Implementing Turing Machines in NNs (Siegelmann and Sontag, 1992; Siegelmann, 1996): recurrent NNs a la Elman, with sigmoid activation function, rational weights, and infinite precision can simulate a TM (in real time)
- This is better than the Shannon, 1941 GPAC model (can do e.g. Γ function) but really we are sub-Turing so the hypercomputing advantage clamed by Siegelmann, 1999 is illusory
- For careful analysis of where the tire meets the road, see Weiss, Goldberg, and Yahav, 2018 (the issue is with arbitrary precision required for embedding loops of arbitrary depth)

How well can LLMs learn formal languages?

- Depends on how you define ‘learn well’. Kleinberg and Mullainathan, 2024 define ‘generation in the limit’ as opposed to ‘identification in the limit’.
- First we assume standard 80-10-10 split
- Generate random train/dev/set (disjoint, taken i.i.d from some language L)
- Set hyperparameters based on dev performance
- Test of test (unseen, disjoint from dev or train), compute the usual figures of merit (prec, rec, F)
- We decide *in advance* what sort of merit we accept
- This need not be 100%! In fact, it shouldn’t be, since humans don’t perform flawlessly on such tests
- As is well known, LLMs make significant errors on multiplying two 3-digit numbers (Bai et al., 2025)

A concrete example

- ‘majority language’ has more 0s or 1s in the presence of noise given by 2s (Zsámboki et al., 2025)
- First we assume standard 80-10-10 split
- Generate random train/dev/set (disjoint, taken i.i.d from some language L)
- Set hyperparameters based on dev performance
- Test of test (unseen, disjoint from dev or tran), compute the usual figures of merit (prec, rec, F)
- We decide *in advance* what sort of merit we accept (typical is 95% to 99%)
- Using 99% threshold, and short (20-100 symbols) train/dev it is possible to learn the majority language so that performance on 200-300 long strings is still 99%
- In fact, perfect solutions are limited only by the precision of the vector arithmetic

Current state of play

- This a very active area in CL, hundreds of papers every year. A typical example is Wen et al., 2023 on bounded Dyck languages
- Linguistics is doing the same, a typical example is Wilcox, Futrell, and Levy, 2022 testing LLMs for filler-gap dependencies (incl. parasitic gaps) and island constraints. For a current summary see Futrell and Mahowald, 2025
- According to the latest and greatest definition of AGI (Hendrycks et al., 2025) systems improved from 28% (GPT 4) to-58% (GPT-5), and the biggest gaps are in long-term memory, not in abilities generally considered linguistic
- There seems to be real learning going on (Nanda et al., 2023; Lubana et al., 2024)
- We have rule compilers (Weiss, Goldberg, and Yahav, 2021) and decompilers (Friedman, Wettig, and Chen, 2023)
- Logic deduction and large-scale statistical learning are fundamentally related (Domingos, 2025)

Conclusions

- This is fun!
- LLMs are you friend
- Hungary needs to get its act together
- Maybe the HAS should follow Széchenyi's original will and the resulting 1827 XI. bill (törvénycikk) about "A hazai nyelv művelésére fölállitandó tudós társaságról vagy magyar akadémiáról" which defines the mission of HAS as *refining our language . . . in all arts and sciences* ("a hazai nyelvnek . . . a tudományok és művészetek minden nemében leendő kiművelése)
- To the extent HAS is limited in this by its minimal budget and the incessant attacks originating with FIDESZ minister of propaganda Antal Rogán, the government needs to get its act together!

Thank You!

Bai, Xiaoyan et al. (2025). *Why Can't Transformers Learn Multiplication? Reverse-Engineering Reveals Long-Range Dependency Pitfalls*. arXiv: 2510.00184 [cs.LG]. URL: https://arxiv.org/abs/2510.00184.

Ben-David, Shai et al. (2019). "Learnability can be undecidable". In: *Nature Machine Intelligence* 1, pp. 44–48. URL: https://doi.org/10.1038/s42256-018-0002-3.

Bresnan, Joan (2001). *Lexical-Functional Syntax*. Oxford, UK: Blackwell.

Chomsky, Noam (1956). "Three models for the description of language". In: *IRE Transactions on Information Theory* 2, pp. 113–124.

— (1965). *Aspects of the Theory of Syntax*. MIT Press. DOI: 10.21236/AD0616323.

— (1970). "Remarks on nominalization". In: *Readings in English Transformational Grammar*. Ed. by R. Jacobs and P. Rosenbaum. Waltham, MA: Blaisdell, pp. 184–221.

- Chomsky, Noam (1971). "Deep Structure, Surface Structure, and Semantic Interpretation". In: *Semantics: An Interdisciplinary Reader in Philosophy Linguistics and Psychology*. Ed. by Danny D. Steinberg and Leon A. Jakobovits. Cambridge University Press, pp. 183–216.
- — (1973). "Conditions on Transformations". In: *A festschrift for Morris Halle*. Ed. by S.R. Anderson and P. Kiparsky. New York: Holt, Rinehart and Winston.
- — (1981). *Lectures on Government and Binding*. Dordrecht: Foris.
- — (1995). *The Minimalist Program*. MIT Press.
- Chomsky, Noam and Morris Halle (1968). *The Sound Pattern of English*. New York: Harper and Row.
- Chomsky, Noam et al. (2023). *Merge and the Strong Minimalist Thesis*. Cambridge University Press. DOI: 10.1017/9781009343244.
- Culy, Christopher (1985). "The complexity of the vocabulary of Bambara". In: *Linguistics and Philosophy* 8.3, pp. 345–351.

Cybenko, George (1989). "Approximation by superpositions of a sigmoidal function". In: *Mathematics of control, signals and systems* 2.4, pp. 303–314.

Domingos, Pedro (2025). *Tensor Logic: The Language of AI*. arXiv: 2510.12269 [cs.AI]. URL: https://arxiv.org/abs/2510.12269.

Friedman, Dan, Alexander Wettig, and Danqi Chen (2023). *Learning Transformer Programs*. arXiv: 2306.01128 [cs.LG]. URL: https://arxiv.org/abs/2306.01128.

Futrell, Richard and Kyle Mahowald (2025). "How Linguistics Learned to Stop Worrying and Love the Language Models". In: *Behavioral and Brain Sciences*. DOI: 10.1017/S0140525X2510112X.

Gold, E. Mark (1967). "Language identification in the limit". In: *Information and Control* 10, pp. 447–474.

Goldsmith, John A. (1990). *Autosegmental and Metrical Phonology*. Cambridge, MA: Blackwell.

Goldsmith, John A. and Geoffrey J. Huck (2013). *Ideology and linguistic theory: Noam Chomsky and the deep structure debates*. Routledge.

Harley, Heidi (2014). "On the identity of roots". In: *Theoretical Linguistics* 40.3/4, pp. 225–276.

Harris, Randy Allen (1995). *The Linguistics Wars*. Oxford University Press.

Harris, Zellig (1951). *Methods in Structural Linguistics*. University of Chicago Press.

Heinz, Jeffrey (2018). "The computational nature of phonological generalizations". In: *Phonological Typology*. Ed. by Larry Hyman and Frans Plank. De Gruyter Mouton, pp. 126–195.

Hendrycks, Dan et al. (2025). *A Definition of AGI*. arXiv: 2510.18212 [cs.AI]. URL: https://arxiv.org/abs/2510.18212.

Joshi, Aravind K., K. Vijay-Shanker, and David J. Weir (1990). *The Convergence of Mildly Context-Sensitive Grammar Formalisms*. Tech. rep. University of Pennsylvania.

- Karttunen, L. and K.R. Beesley (2003). "Finite State Morphology". In: *CSLI Studies in Computational Linguistics. CSLI Publication, Stanford.*
- Karttunen, Lauri (1998). "The proper treatment of optimality in computational phonology: plenary talk". In: *Proceedings of the International Workshop on Finite State Methods in Natural Language Processing*. Association for Computational Linguistics, pp. 1–12.
- Kleinberg, Jon and Sendhil Mullainathan (2024). "Language Generation in the Limit". In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. URL: https://openreview.net/forum?id=FGTDe6EA0B.
- Kornai, András (1985). "Natural Languages and the Chomsky Hierarchy". In: *Proceedings of the 2nd European ACL Conference*. Ed. by Margaret King, pp. 1–7.
- — (1987). "Finite state semantics". In: *Computerlinguistik und philologische Datenverarbeitung*. Ed. by U. Klenk, P. Scherber, and M. Thaller. Hildesheim: Georg Olms, pp. 59–70.

- Kornai, András (1995). *Formal Phonology*. New York: Garland Publishing. ISBN: 0815317301.
- — (2009). "The complexity of phonology". In: *Linguistic Inquiry* 40.4, pp. 701–712.
- — (2021). *Kik termelik a tudást?* URL: https://hlt.bme.hu/media/pdf/kdfi21.pdf.
- Kracht, M. (2002). "Referent systems and relational grammar". In: *Journal of Logic, Language and Information* 11.2, pp. 251–286. ISSN: 0925-8531.
- Lubana, Ekdeep Singh et al. (2024). "The Concept Percolation Hypothesis: Analyzing the Emergence of Capabilities in Neural Networks Trained on Formal Grammars". In: *ICML 2024 Workshop on Mechanistic Interpretability*. URL: https://openreview.net/forum?id=kiNU4jwUoW.
- Marcolli, Matilde, Robert Berwick, and Noam Chomsky (2023a). *Old and New Minimalism: a Hopf algebra comparison*. URL: https://lingbuzz.net/lingbuzz/007373.

Marcolli, Matilde, Robert Berwick, and Noam Chomsky (2023b). *Syntax-semantics interface: an algebraic model*. URL: https://ling.auf.net/lingbuzz/007696.

Marcolli, Matilde, Noam Chomsky, and Robert Berwick (2023). *Mathematical Structure of Syntactic Merge*. arXiv: 2305.18278 [cs.CL].

Marsh, William and Barbara Partee (1984). "How non-context-free is variable binding?" In: *Proceedings of the West Coast Conference on Formal Linguistics III*. Ed. by M. Cobler, S. MacKaye, and M. Wescoat. Stanford, CA, pp. 179–190.

McNaughton, Robert and Seymour Papert (1971). *Counter-Free Automata*. MIT Press.

Michaelis, Jens and Marcus Kracht (1997). "Semilinearity as a syntactic invariant". In: *Logical Aspects of Computational Linguistics*. Ed. by Christian Retoré. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 329–345. ISBN: 978-3-540-69631-5.

Nanda, Neel et al. (2023). *Progress measures for grokking via mechanistic interpretability*. arXiv: 2301.05217 [cs.LG].

Nida, Eugene (1949). *Morphology: The Descriptive Analysis of Words*. Ann Arbor: University of Michigan.

Peters, Stanley and Robert W. Ritchie (1973). "On the generative power of transformational grammars". In: *Information Sciences* 6, pp. 49–83.

Pullum, Geoffrey K. and Gerald Gazdar (1982). "Natural languages and context free languages". In: *Linguistics and Philosophy* 4, pp. 471–504.

Radzinski, Daniel (1991). "Chinese number-names, Tree Aadjoining Languages, and Mild Context-Sensitivity". In: *Computational Linguistics* 17.3, pp. 277–299.

Ross, John Robert (1967). *Constraints on Variables in Syntax*. MIT.

Salomaa, A. (1971). "The generative capacity of transformational grammars of Ginsburg and Partee". In: *Information and control* 18.3, pp. 227–232. ISSN: 0019-9958.

Shannon, Claude E. (1941). "Mathematical Theory of the differential analyzer". In: *Journal of Mathematics and Physics of the MIT* 20, pp. 337–354.

Shieber, Stuart M. (1985). "Evidence against the context-freeness of natural language". In: *Linguistics and Philosophy* 8, pp. 333–343.

Siegelmann, Hava T. (1996). "Recurrent neural networks and finite automata". In: *Computational Intelligence* 12, pp. 567–574.

— (1999). *Neural Networks and Analog Computation: Beyond the Turing Limit*. Birkhäuser.

Siegelmann, Hava T. and Eduardo D. Sontag (1992). "On the computational power of neural nets". In: *Proc 5th ACM Conference on Computational Learning Theory, COLT 1992*, pp. 440–449.

Sutton, Richard (2019). *The bitter lesson*. URL: http://www.incompleteideas.net/IncIdeas/BitterLesson.html.

Weiss, Gail, Yoav Goldberg, and Eran Yahav (2018). "On the Practical Computational Power of Finite Precision RNNs for

Language Recognition". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by Iryna Gurevych and Yusuke Miyao. Melbourne, Australia: Association for Computational Linguistics, pp. 740–745. DOI: 10.18653/v1/P18-2117. URL: https://aclanthology.org/P18-2117.

Weiss, Gail, Yoav Goldberg, and Eran Yahav (2021). *Thinking Like Transformers*. arXiv: 2106.06981 [cs.LG]. URL: https://arxiv.org/abs/2106.06981.

Wen, K. et al. (2023). "Transformers are uninterpretable with myopic methods: a case study with bounded dyck grammars". In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh et al. Vol. 36. Curran Associates, Inc., pp. 38723–38766. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/79ba1b827d3fc58e129d1cbfc8ff69f2-Paper-Conference.pdf.

Wilcox, Ethan, Richard Futrell, and Roger Levy (2022). *Using Computational Models to Test Syntactic Learnability*. DOI: 10.1162/ling_a_00491.

Yli-Jyrä, Anssi (2018). "Optimal Kornai-Karttunen Codes for Restricted Autosegmental Representations". In: *Karttunen Festschrift*. Ed. by Cleo Condoravdi. Stanford: CSLI Publications.

Zsámboki, Pál et al. (2025). "Do attention heads compete or cooperate during counting?" In: arXiv: 2502.06923. URL: https://arxiv.org/abs/2502.06923.