Trade-Offs Between Hallucinations and Mode Collapse in Language Generation

Grigoris Velegkas

Based on joint works with Alkis Kalavasis and Anay Mehrotra

Yale

Early Days of CS + Language Learning [Shannon '51]

- **Shannon** introduced n-grams, tremendous impact on early text generators

- Text guessing game with his wife: reveal prefix of text, try to guess continuation!

| _

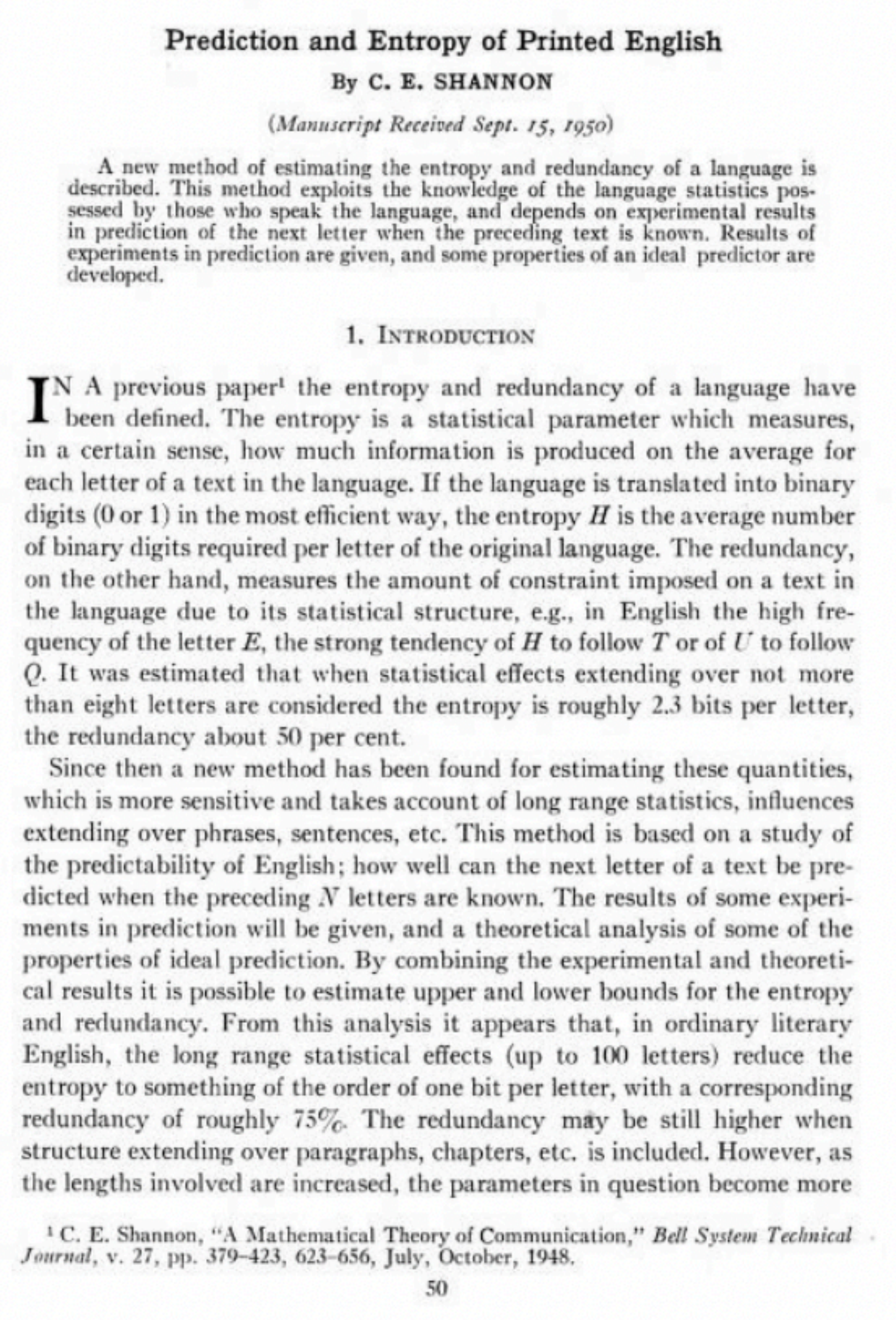
| *C* _

| *Ch* _

| *Chess* _

| *Chess is a board game for two player* _

- Related to LLM training!



Prediction and Entropy of Printed English

By C. E. SHANNON

(*Manuscript Received Sept. 15, 1950*)

A new method of estimating the entropy and redundancy of a language is described. This method exploits the knowledge of the language statistics possessed by those who speak the language, and depends on experimental results in prediction of the next letter when the preceding text is known. Results of experiments in prediction are given, and some properties of an ideal predictor are developed.

1. Introduction

In a previous paper[1] the entropy and redundancy of a language have been defined. The entropy is a statistical parameter which measures, in a certain sense, how much information is produced on the average for each letter of a text in the language. If the language is translated into binary digits (0 or 1) in the most efficient way, the entropy H is the average number of binary digits required per letter of the original language. The redundancy, on the other hand, measures the amount of constraint imposed on a text in the language due to its statistical structure, e.g., in English the high frequency of the letter E, the strong tendency of H to follow T or of U to follow Q. It was estimated that when statistical effects extending over not more than eight letters are considered the entropy is roughly 2.3 bits per letter, the redundancy about 50 per cent.

Since then a new method has been found for estimating these quantities, which is more sensitive and takes account of long range statistics, influences extending over phrases, sentences, etc. This method is based on a study of the predictability of English; how well can the next letter of a text be predicted when the preceding N letters are known. The results of some experiments in prediction will be given, and a theoretical analysis of some of the properties of ideal prediction. By combining the experimental and theoretical results it is possible to estimate upper and lower bounds for the entropy and redundancy. From this analysis it appears that, in ordinary literary English, the long range statistical effects (up to 100 letters) reduce the entropy to something of the order of one bit per letter, with a corresponding redundancy of roughly 75%. The redundancy may be still higher when structure extending over paragraphs, chapters, etc. is included. However, as the lengths involved are increased, the parameters in question become more

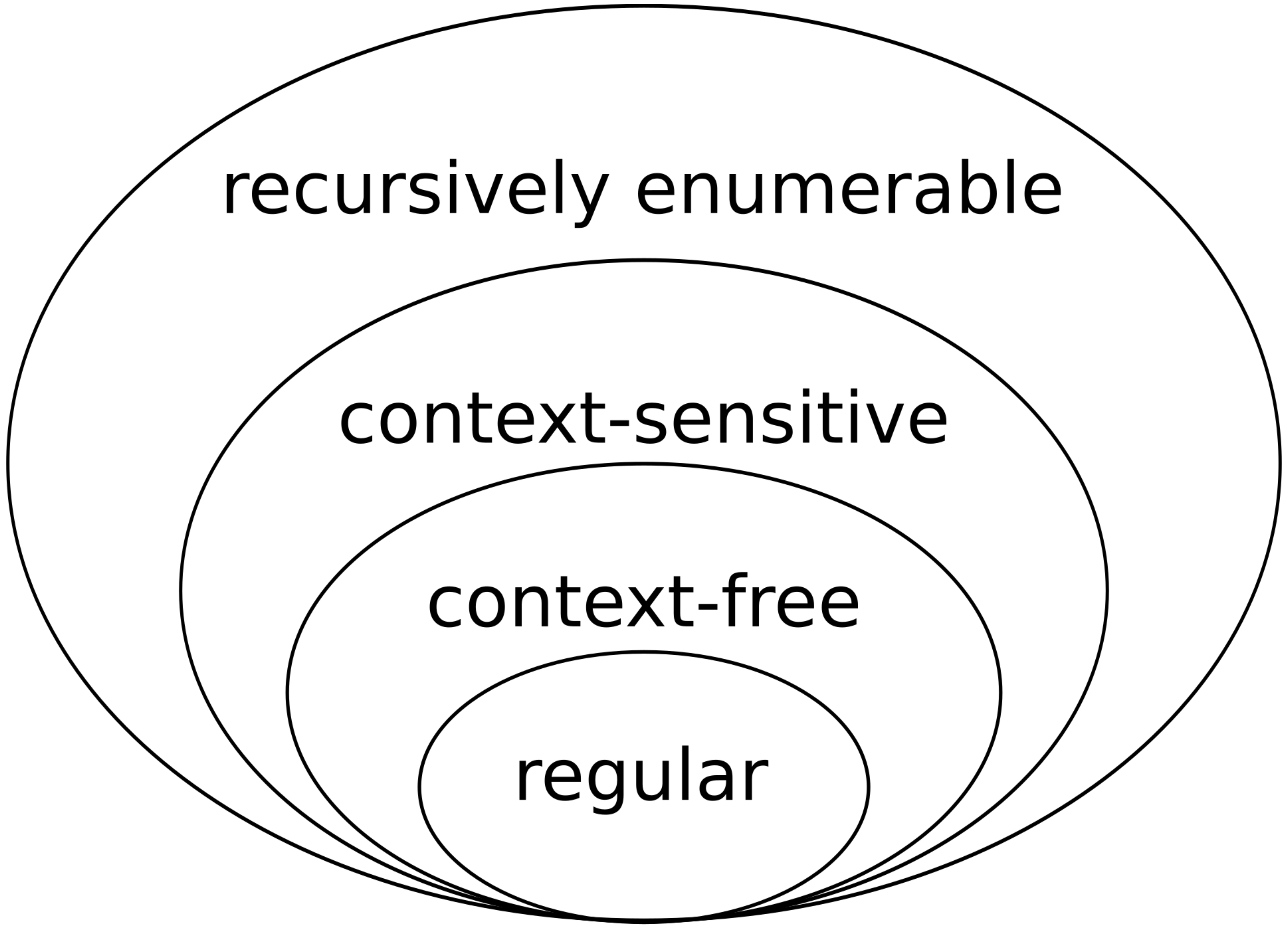
[1] C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, v. 27, pp. 379–423, 623–656, July, October, 1948.

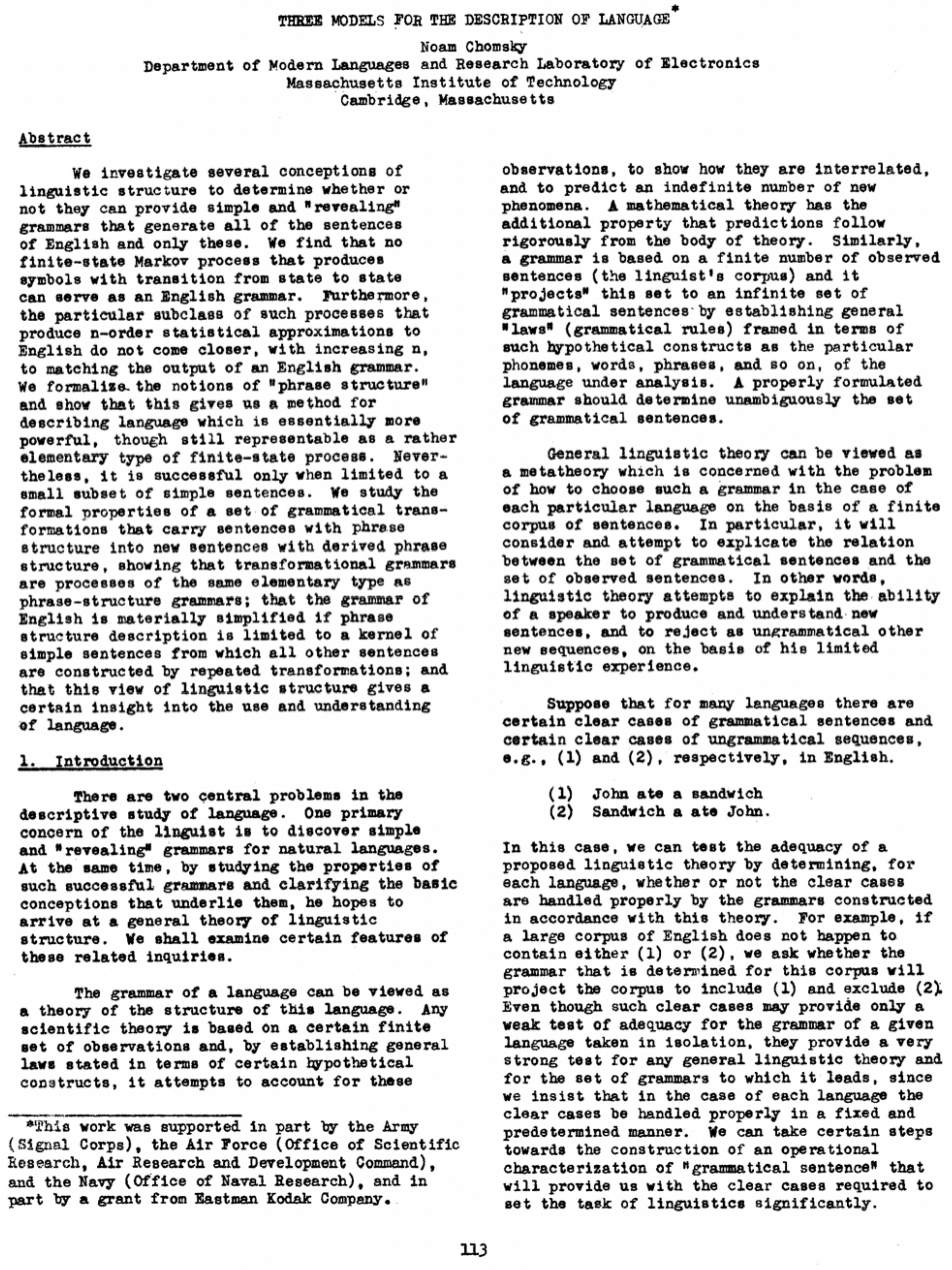
50

Yale

Early Days of CS + Language Learning [Chomsky '56]

- Chomsky hierarchy: a classification of formal languages based on their complexity





THREE MODELS FOR THE DESCRIPTION OF LANGUAGE*

Noam Chomsky
Department of Modern Languages and Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts

Abstract

We investigate several conceptions of linguistic structure to determine whether or not they can provide simple and "revealing" grammars that generate all of the sentences of English and only these. We find that no finite-state Markov process that produces symbols with transition from state to state can serve as an English grammar. Furthermore, the particular subclass of such processes that produce n-order statistical approximations to English do not come closer, with increasing n, to matching the output of an English grammar. We formalize the notions of "phrase structure" and show that this gives us a method for describing language which is essentially more powerful, though still representable as a rather elementary type of finite-state process. Nevertheless, it is successful only when limited to a small subset of simple sentences. We study the formal properties of a set of grammatical transformations that carry sentences with phrase structure into new sentences with derived phrase structure, showing that transformational grammars are processes of the same elementary type as phrase-structure grammars; that the grammar of English is materially simplified if phrase structure description is limited to a kernel of simple sentences from which all other sentences are constructed by repeated transformations; and that this view of linguistic structure gives a certain insight into the use and understanding of language.

1. Introduction

There are two central problems in the descriptive study of language. One primary concern of the linguist is to discover simple and "revealing" grammars for natural languages. At the same time, by studying the properties of such successful grammars and clarifying the basic conceptions that underlie them, he hopes to arrive at a general theory of linguistic structure. We shall examine certain features of these related inquiries.

The grammar of a language can be viewed as a theory of the structure of this language. Any scientific theory is based on a certain finite set of observations and, by establishing general laws stated in terms of certain hypothetical constructs, it attempts to account for these observations, to show how they are interrelated, and to predict an indefinite number of new phenomena. A mathematical theory has the additional property that predictions follow rigorously from the body of theory. Similarly, a grammar is based on a finite number of observed sentences (the linguist's corpus) and it "projects" this set to an infinite set of grammatical sentences by establishing general "laws" (grammatical rules) framed in terms of such hypothetical constructs as the particular phonemes, words, phrases, and so on, of the language under analysis. A properly formulated grammar should determine unambiguously the set of grammatical sentences.

General linguistic theory can be viewed as a metatheory which is concerned with the problem of how to choose such a grammar in the case of each particular language on the basis of a finite corpus of sentences. In particular, it will consider and attempt to explicate the relation between the set of grammatical sentences and the set of observed sentences. In other words, linguistic theory attempts to explain the ability of a speaker to produce and understand new sentences, and to reject as ungrammatical other new sequences, on the basis of his limited linguistic experience.

Suppose that for many languages there are certain clear cases of grammatical sentences and certain clear cases of ungrammatical sequences, e.g., (1) and (2), respectively, in English.

(1) John ate a sandwich
(2) Sandwich a ate John.

In this case, we can test the adequacy of a proposed linguistic theory by determining, for each language, whether or not the clear cases are handled properly by the grammars constructed in accordance with this theory. For example, if a large corpus of English does not happen to contain either (1) or (2), we ask whether the grammar that is determined for this corpus will project the corpus to include (1) and exclude (2). Even though such clear cases may provide only a weak test of adequacy for the grammar of a given language taken in isolation, they provide a very strong test for any general linguistic theory and for the set of grammars to which it leads, since we insist that in the case of each language the clear cases be handled properly in a fixed and predetermined manner. We can take certain steps towards the construction of an operational characterization of "grammatical sentence" that will provide us with the clear cases required to set the task of linguistics significantly.

*This work was supported in part by the Army (Signal Corps), the Air Force (Office of Scientific Research, Air Research and Development Command), and the Navy (Office of Naval Research), and in part by a grant from Eastman Kodak Company.

Yale

Early Days of CS + Language Learning [Gold '67]

"*I wish to construct a precise model for the intuitive notion "able to speak a language"* ***in order to be able to investigate theoretically how it can be achieved artificially.*** *Since we* ***cannot explicitly write down the rules of English*** *…* ***artificial intelligence which is designed to speak English will have to learn its rules from implicit information****….*"

- **Gold's** model is a predecessor to the celebrated PAC framework [Valiant 1984] (Turing Award 2010)
- Describes many pioneering ideas:
  - <u>Learning from examples</u>
  - <u>Hypothesis class</u>
  - <u>Two-player online adversarial game (predecessor to Littlestone's setting)</u>
  - <u>Active learning (!)</u>

INFORMATION AND CONTROL 10, 447–474 (1967)

Language Identification in the Limit

E Mark Gold*

The RAND Corporation

Language learnability has been investigated. This refers to the following situation: A class of possible languages is specified, together with a method of presenting information to the learner about an unknown language, which is to be chosen from the class. The question is now asked, "Is the information sufficient to determine which of the possible languages is the unknown language?" Many definitions of learnability are possible, but only the following is considered here: Time is quantized and has a finite starting time. At each time the learner receives a unit of information and is to make a guess as to the identity of the unknown language on the basis of the information received so far. This process continues forever. The class of languages will be considered *learnable* with respect to the specified method of information presentation if there is an algorithm that the learner can use to make his guesses, the algorithm having the following property: Given any language of the class, there is some finite time after which the guesses will all be the same and they will be correct.

In this preliminary investigation, a *language* is taken to be a set of strings on some finite alphabet. The alphabet is the same for all languages of the class. Several variations of each of the following two basic methods of information presentation are investigated: A *text* for a language generates the strings of the language in any order such that every string of the language occurs at least once. An *informant* for a language tells whether a string is in the language, and chooses the strings in some order such that every string occurs at least once.

It was found that the class of context-sensitive languages is learnable from an informant, but that not even the class of regular languages is learnable from a text.

1. MOTIVATION: TO SPEAK A LANGUAGE

The study of language identification described here derives its motivation from artificial intelligence. The results and the methods used also

* Present address: Institute for Formal Studies, 1720 Pontius Ave., Los Angeles, California 90025. Present sponsor: Air Force Office of Scientific Research. Contract F44620-67-C-0018.

447

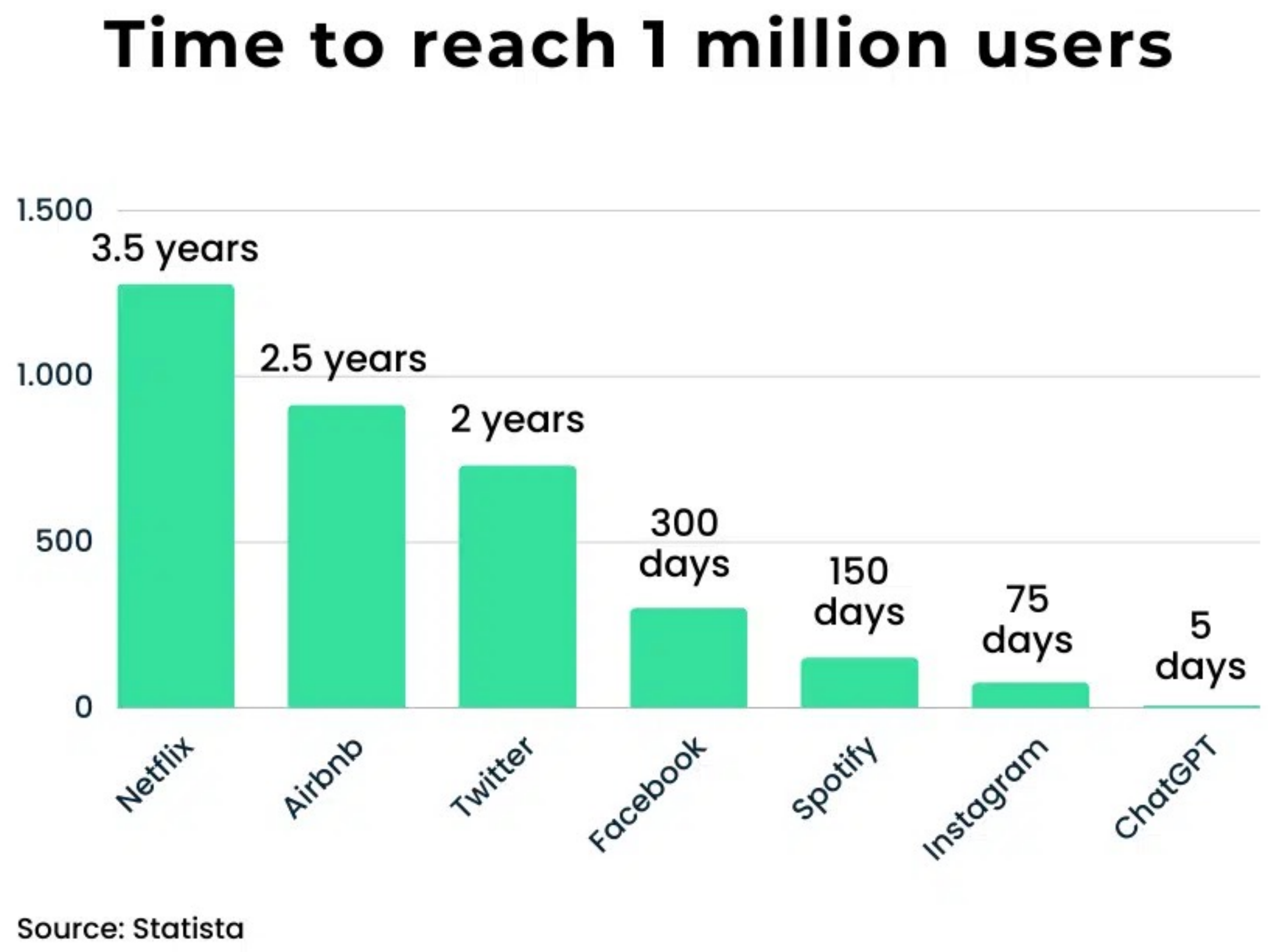
Modern Days of CS + Language Learning

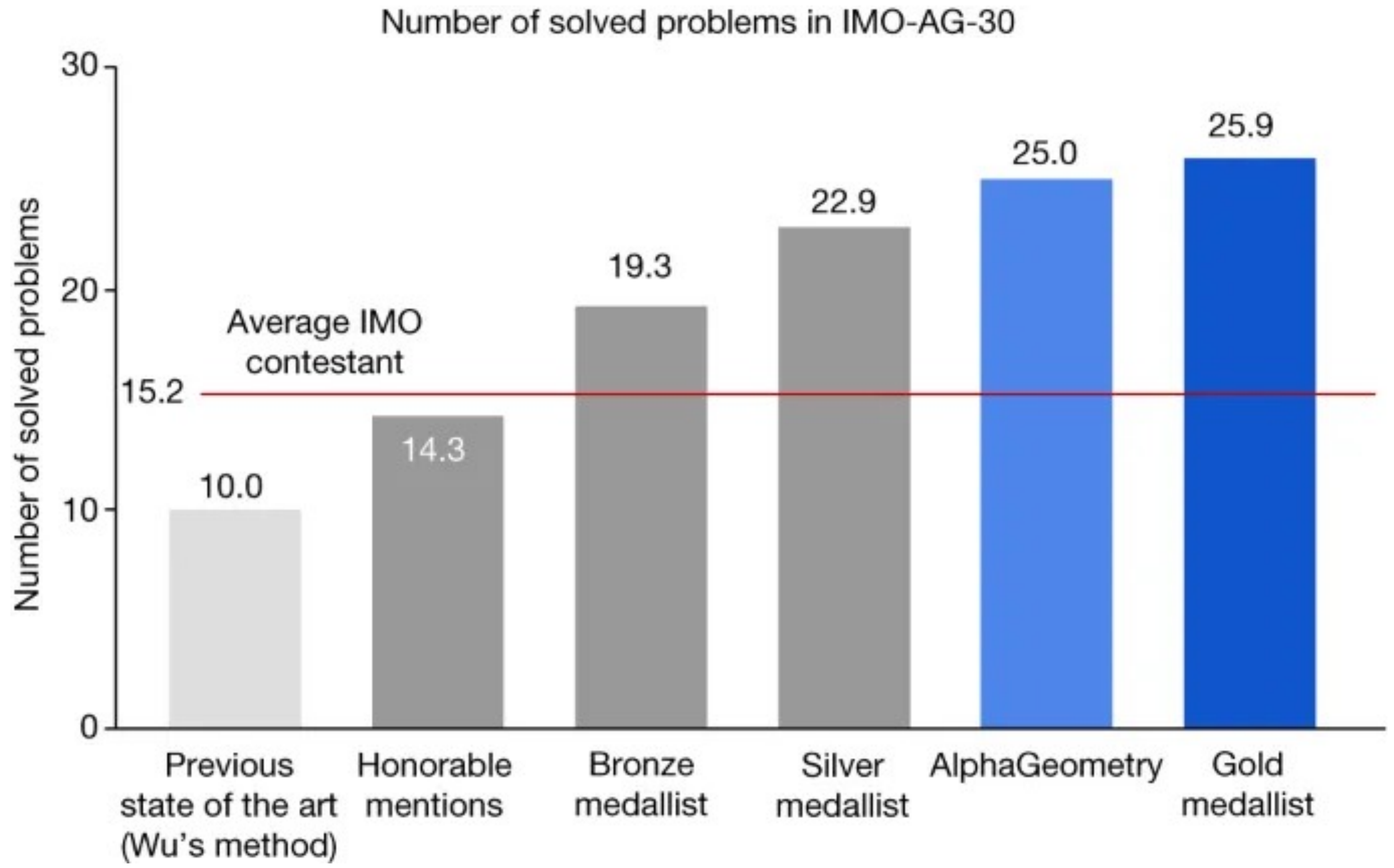
- Variety of techniques based on modern deep learning
  - Word-to-vector representation [Mikolov, Chen, Corrado, Dean’13]
  - Attention [Bahdanau, Cho, Bengio ’14]
  - Seq-2-seq [Sutskever, Vinyals, Le ’14]
  - Transformers [Vaswani et al. ’17]
  - GPT-2 [Radford et al. ’19]

ChatGPT

Gemini

Modern Days of CS + Language Learning



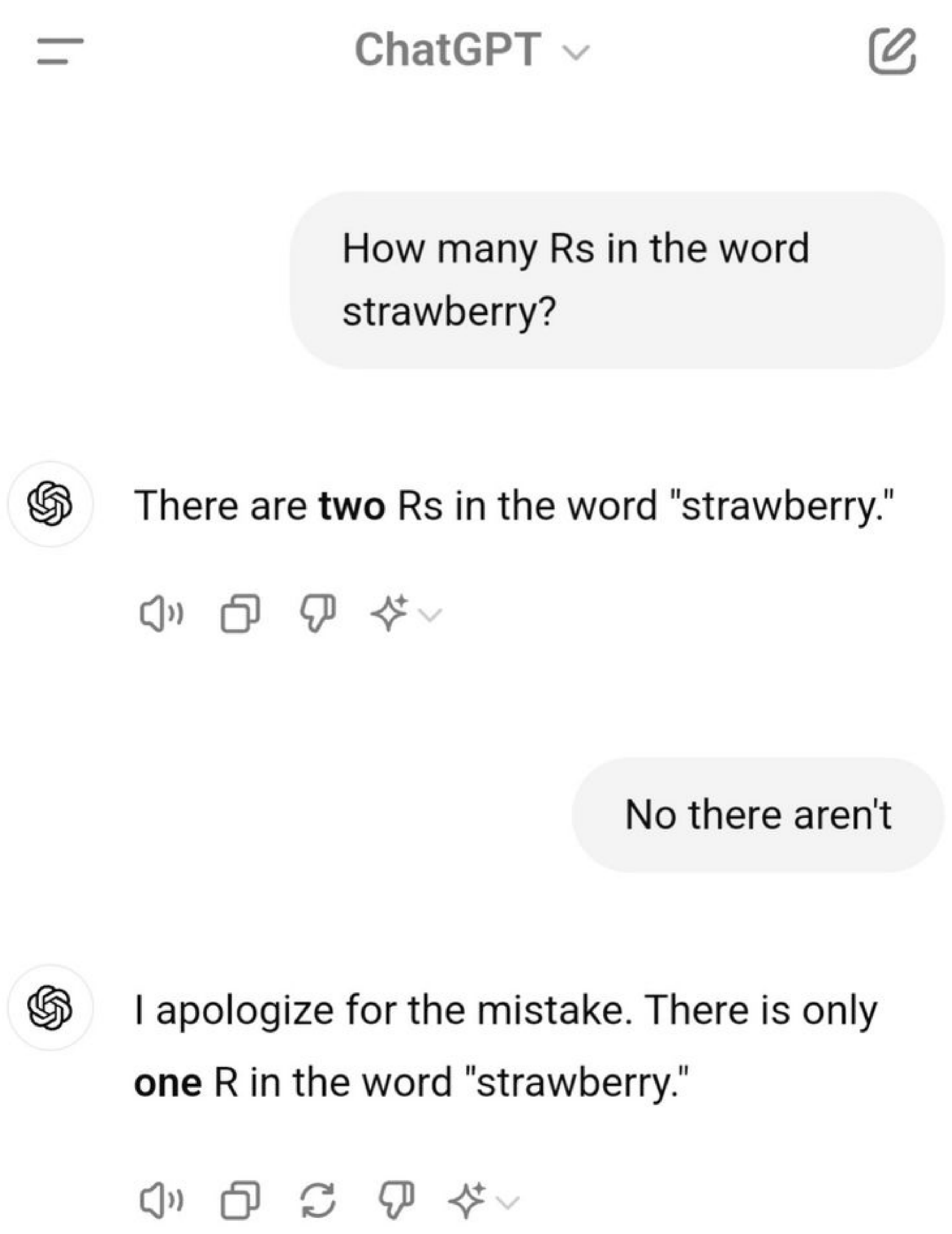


Hallucinations

I would like you to give me a succinct definition of hallucinations in LLMs

Hallucinations in large language models (LLMs) refer to instances where the model generates outputs that are factually incorrect, nonsensical, or not grounded in the input or reality, despite sounding plausible.

Hallucinations - Early Days



Hallucinations

BUSINESS > AEROSPACE & DEFENSE

What Air Canada Lost In 'Remarkable' Lying AI Chatbot Case

By Marisa Garcia, Senior Contributor. ⓘ Offering an insider's view of the bus

Follow Author

Feb 19, 2024 at 06:03am EST

Yale

Overarching Question

Can hallucinations be avoided with “better” models or are there inherent limitations?

- **This talk:** no computational constraints, no architecture-specific problems, abstract mathematical model to study this question

Outline of the Talk

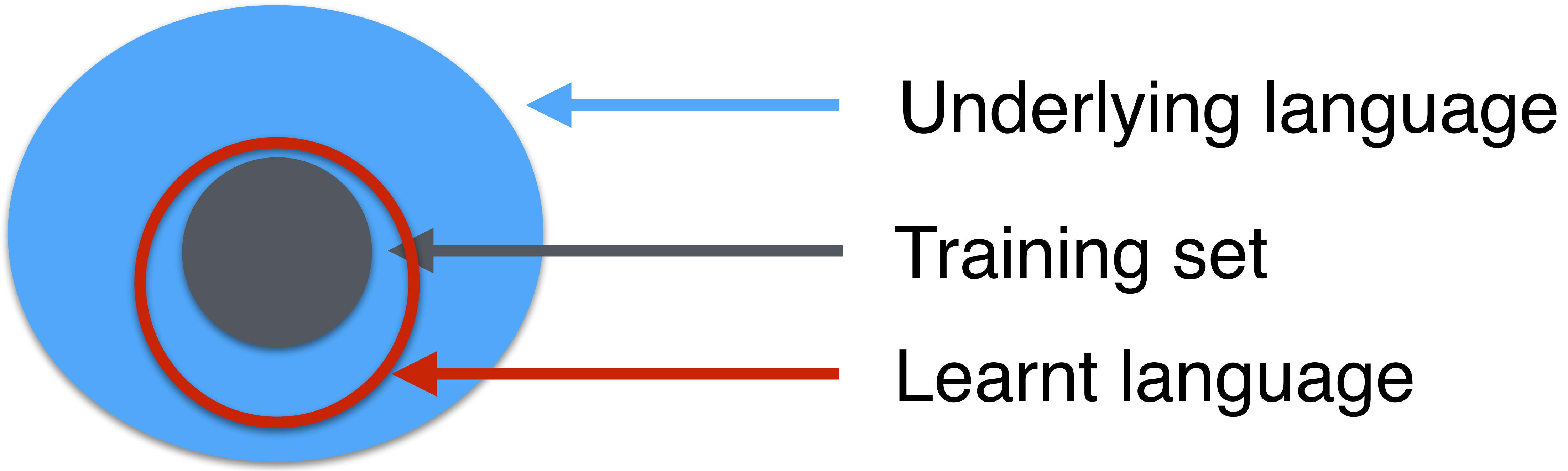
- Motivation: CS and Language
- Theoretical Model
- Overview of our Definitions and Results
- Overview of (some) Proofs

Outline of the Talk

- Motivation: CS and Language
- <u>Theoretical Model</u>
- Overview of our Definitions and Results
- Overview of (some) Proofs

What is the Essence of Language Generation?

Given text from an unknown language, learn to produce “valid” text that has not been seen before [Kleinberg and Mullainathan ’24]



- Simplications:
  - Remove the requirement to learn a distribution
  - Consider a promptless model (extension to prompted model can be achieved)
  - Do not necessarily need to learn the entirety of the target language

Yale

Mathematical Formulation

Classical work on language identification by Gold [Gol 67] and Angluin [Ang 79,80]

- Countable domain $\mathscr{X}$ (e.g., $\{0,1\}^*$, $\mathbb{N}$), countable collection of languages $\mathscr{L} = \{L_1, L_2, \ldots\}$
- Language identification is an infinite two-player game between the learner and the adversary:
  - The adversary picks a target language $K \in \mathscr{L}$
  - In every round $t = 1,2,3,\ldots$, the adversary presents some $x_t \in K$, the learner guesses $i_t \in \mathbb{N}$
  - The learner wins if there is some (finite) $t^* \in \mathbb{N}$ such that for all $t' \geq t^* : i_{t^*} = i_{t'}$ and $L_{i_{t'}} = K$
- The adversary presents a complete enumeration (for every $w \in K$ there is some t such that $x_t = w$)
- The learner can access $\mathscr{L}$ through a membership oracle ("is $w \in L_i$?") and subset oracle ("is $L_i \subseteq L_j$?")
- $\mathscr{L}$ is identifiable (in the limit) if there is a learner that wins for all $K \in \mathscr{L}$ and for all enumerations of K

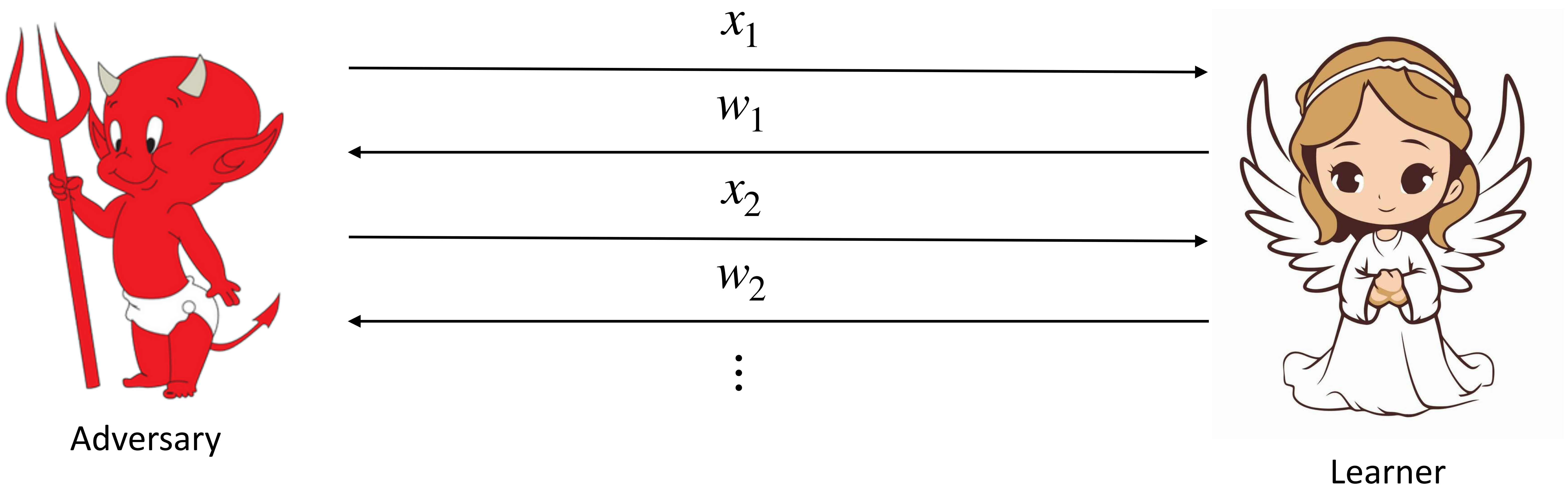
Mathematical Formulation (Cont.)

Variation of the model proposed by [KM 24]: generation in the limit

- Countable domain $\mathscr{X}$ (e.g., $\{0,1\}^*$), countable collection of infinite languages $\mathscr{L} = \{L_1, L_2, \ldots\}$
- Language generation is an infinite two-player game between the learner and the adversary:
  - The adversary picks a target language $K \in \mathscr{L}$
  - In every round $t = 1,2,3,\ldots$, the adversary presents some $x_t \in K$, the learner guesses $w_t \in \mathscr{X}$
  - The learner wins if there is some (finite) $t^* \in \mathbb{N}$ such that for all $t' \geq t^*$: $w_{t'} \in K$ and $w_{t'} \notin \{x_1, \ldots, x_{t'}\}$
- The adversary presents a complete enumeration (for every $w \in K$ there is some t such that $x_t = w$)
- The learner can access $\mathscr{L}$ through a membership oracle ("is $w \in L_i$?") and subset oracle ("is $L_i \subseteq L_j$?")
- $\mathscr{L}$ is generatable (in the limit) if there is a learner that wins for all $K \in \mathscr{L}$ and for all enumerations of K

Yale

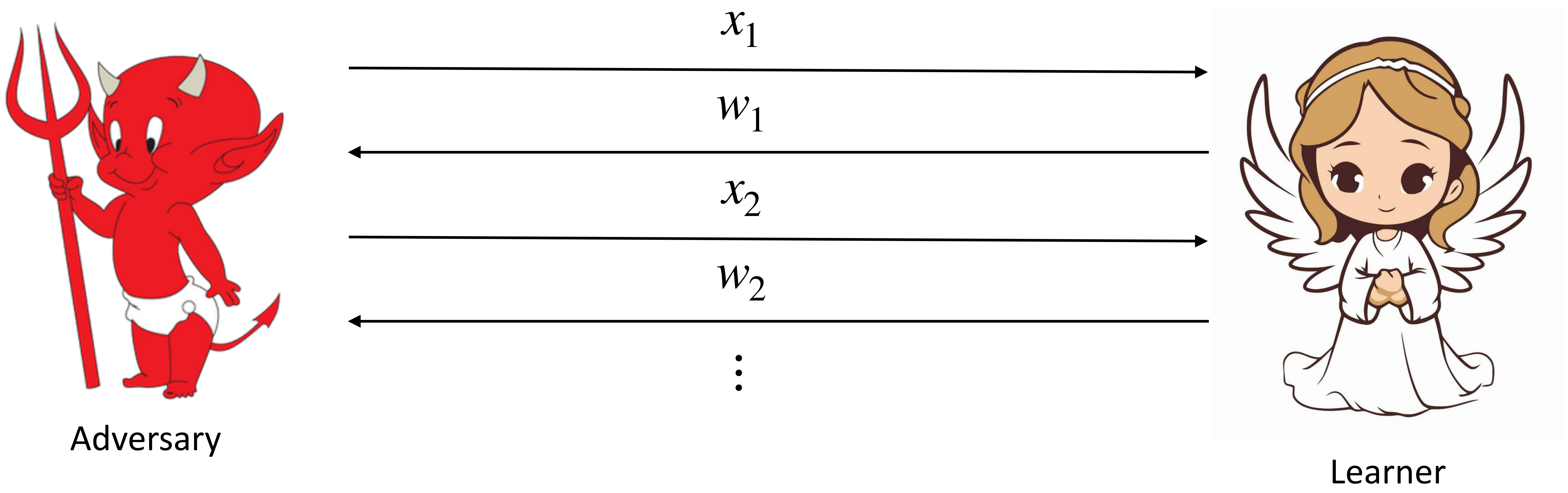
Remarks



The model, while abstract, captures many aspects of LLM training

- The learner does not receive any feedback
- The learner sees only “positive” examples
- The learner is trying to learn an unseen subset of the language
- Crucially, the learner cannot ask if $w \in K$

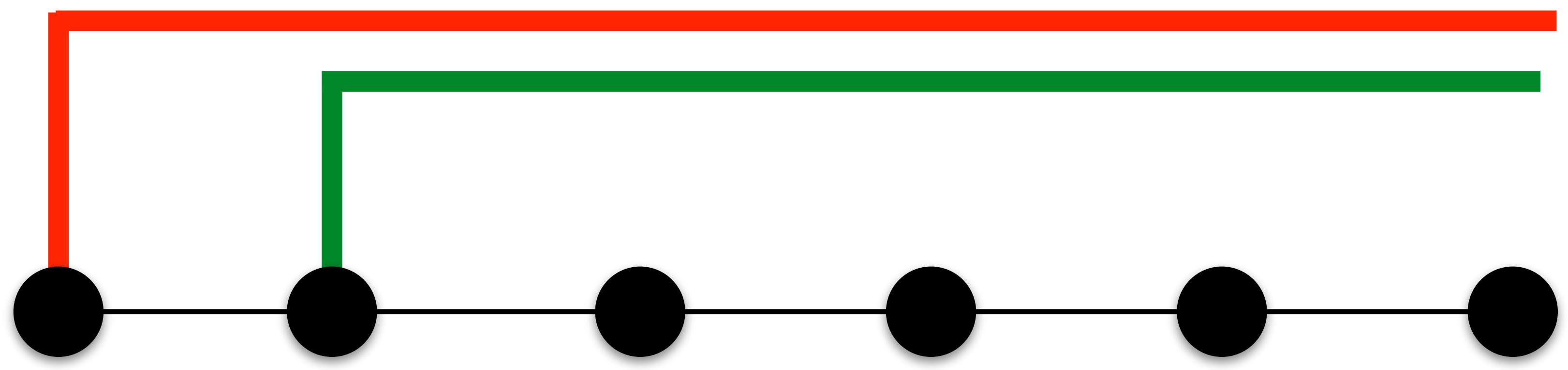
Remarks (Continued)



What makes the problem of identification (and generation) hard?

- Consider $L_i \neq L_j$ and assume that $K = L_j$
  - If $L_j \nsubseteq L_i$ then at some round t the learner will see some $x_t \in L_j, x_t \notin L_i$ so it knows $K \neq L_i$
  - If $L_j \subseteq L_i$ then L_i will always be consistent with the training set!
  - Seeing only positive examples in the training process does not allow the learner to distinguish such languages

Example



Consider the following setting [KM 24, CP 24]

- $\mathscr{X} = \mathbb{Z}$
- $L_i = \{-i, -i+1, -i+2, -i+3, \ldots\}$
- $\mathscr{L} = \{\mathbb{Z}, L_1, L_2, \ldots\}$
- Angluin's result [Ang 80] implies that $\mathscr{L}$ is not identifiable in the limit
- Is $\mathscr{L}$ generatable in the limit?
  - Yes, even with one sample! In every step output an unseen example from $\{x_1+1, x_1+2, \ldots\}$

Identification vs. Generation

[Gold 1967, Angluin 1979, 1980]

Theorem (informal):
Almost all interesting countable collections of languages are not identifiable in the limit.

This applies even to regular languages…

[Kleinberg and Mullainathan 2024]

Theorem (informal):
All countable collections of languages are generatable in the limit.

There exist algorithms that learn to generate new strings without hallucinating!

The Algorithm of Kleinberg and Mullainathan

[Kleinberg and Mullainathan 2024]

Theorem (informal):
All countable collections of languages are generatable in the limit.

Critical languages $C_1^{(t)}, C_2^{(t)}, \ldots$ at time t:

- Consistency: Every $C_i^{(t)}$ contains the training set S_t enumerated so far
- Inclusions: $C_1^{(t)} \supseteq C_2^{(t)} \supseteq \ldots$, where $C_1^{(t)}$ is the first consistent language

Key property: Target language K becomes critical after some finite time t and remains so!

Algorithm: Create chain of critical languages, output from the last one (whose index is at most t)

Validity vs. Breadth

- The learner of [KM 24] suffers from "mode-collapse": it keeps generating from a "decreasing" subset of K
- **Main open question of [KM 24]**:
  - Is there an inherent trade-off between between generating valid strings from K and generating from a "broad" subset of K ?
  - No formal notion of "breadth" was provided
- Series of follow-up works studying this (and related) problems
  - <u>[Kalavasis, Mehrotra, V, STOC'25], [Kalavasis, Mehrotra, V, '24], [Charikar, Pabbaraju, '24]</u>
    - Proposed and studied very similar notions of breadth
  - <u>[Peale, Raman, Reingold, ICML'25], [Kleinberg, Wei, '25]</u>
    - Studied different "fine-grained" notions of breadth
  - [Li, Raman, Tewari, '24]
    - Used a learning-theoretic lens
  - [Raman, Raman, ICML'25]
    - Studied a "noisy" variant

Outline of the Talk

- Motivation: CS and Language
- Theoretical Model
- <u>Overview of our Definitions and Results</u>
- Overview of (some) Proofs

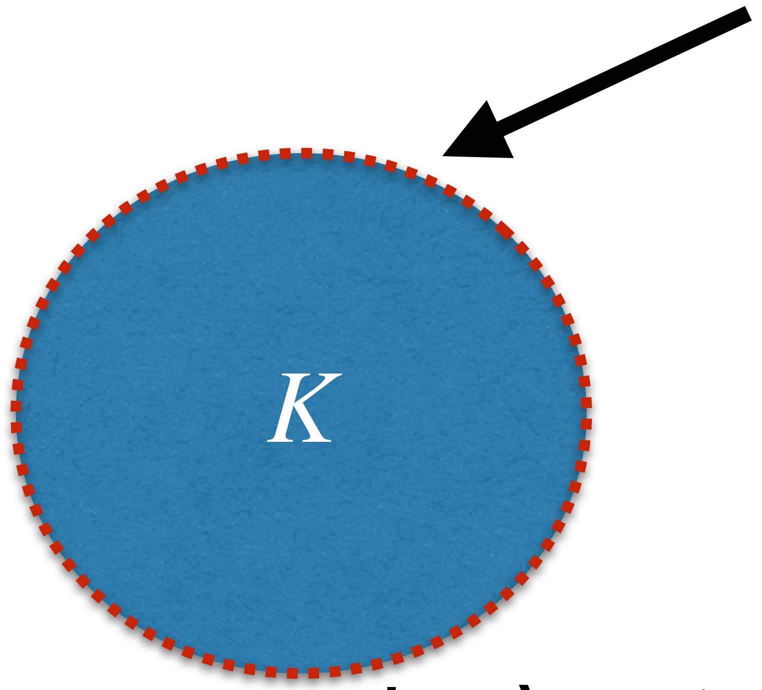
Generation with Breadth

- We view the learner G as a mapping from $S_t = \{x_1, \ldots, x_t\}$ to an (infinite) subset of $\mathscr{X}$

[Kalavasis, Mehrotra, V 2024a]

Definition (exact breadth):
We say that a learner achieves exact breadth in the generation game if for every target language K and for every enumeration of K there is some t^* such that for all $t \geq t^*$ it holds $G(S_t) = K$

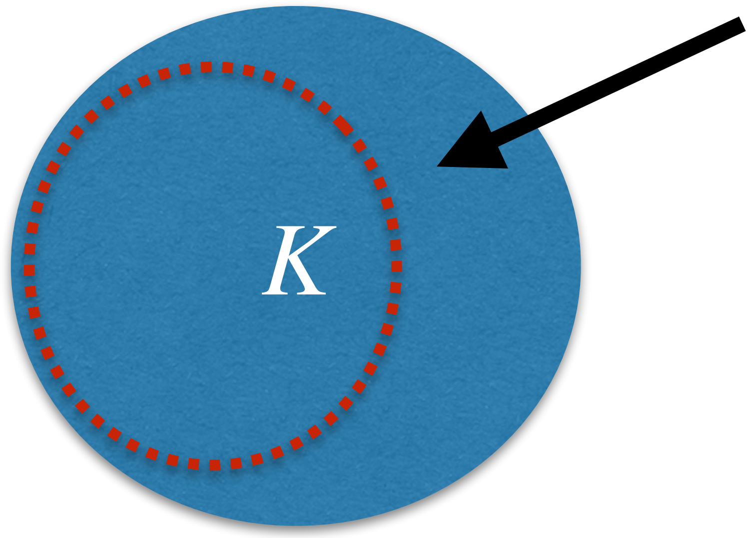
$K = G(S_t)$



[Kalavasis, Mehrotra, V 2024a]

Definition (approximate breadth):
We say that a learner achieves approximate breadth in the generation game if for every target language K and for every enumeration of K there is some t^* such that for all $t \geq t^*$ it holds $G(S_t) \subseteq K, |K \backslash G(S_t)| < \infty$

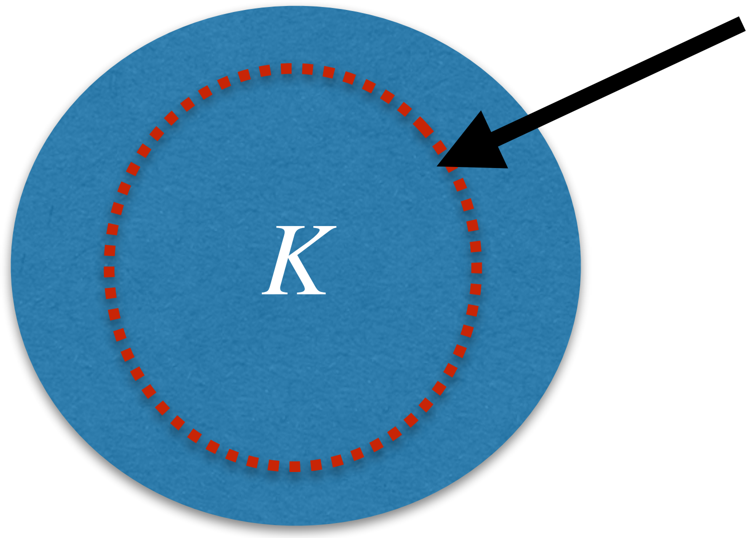
$|K \backslash G(S_t)| < \infty$



[Kalavasis, Mehrotra, V 2024a]

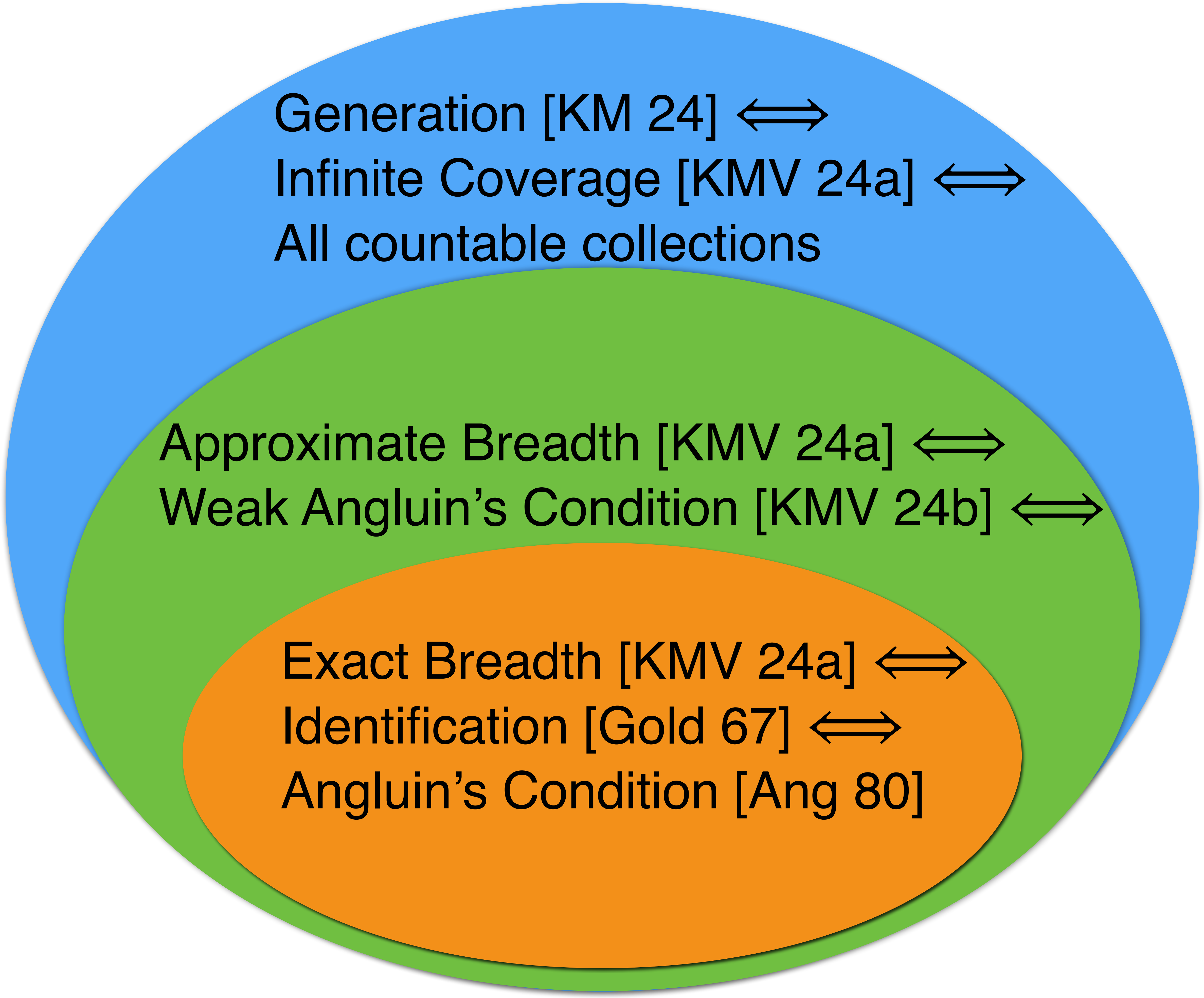
Definition (infinite coverage):
We say that a learner achieves infinite coverage in the generation game if for every target language K and for every enumeration of K there is some t^* such that for all $t \geq t^*$ it holds $G(S_t) \subseteq K, |G(S_t)| = \infty$

$|G(S_t)| = \infty$



Yale

Main Results I



[Kalavasis, Mehrotra, V 2024a, 2024b]

Main Takeaway

LLMs cannot avoid hallucinations while achieving any of these notions of breadth, for most collections of languages

Main Results II

	No Hallucinations $\lvert G(S_t)\backslash K \rvert = 0$	Finite Hallucinations $\lvert G(S_t)\backslash K \rvert < \infty$	Infinite Hallucinations $\lvert G(S_t)\backslash K \rvert = \infty$
Zero Missing Elements $\lvert K\backslash G(S_t) \rvert = 0$	Angluin's Condition [Ang 80] *(i.e., Exact Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections
Finite Missing Elements $\lvert K\backslash G(S_t) \rvert < \infty$	Weak Angluin's Condition [KMV 24b, CP 24] *(i.e., Approximate Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections
Infinite Present Elements $\lvert K \cap G(S_t) \rvert = \infty$	All Countable Collections	All Countable Collections	All Countable Collections

Yale

Stable Generation

- The algorithms from [KM 24] and our works change their outputs infinitely often during the game
  - Recall that Gold [Gol 67] required that the guesses of the algorithm stabilize
  - Moreover, if an algorithm “knows” it has learnt, then it can stabilize

[Kalavasis, Mehrotra, V 2024a]

Definition (stability):
We say that a learner achieves stability in the generation game if for every target language K and for every enumeration of K there is some t^* such that for all $t \geq t^*$ it holds $G(S_t) = G(S_{t^*})$

- Stability in identification comes for free:
  - A language is identifiable in the limit by any algorithm if and only if it is identifiable in the limit by a stable algorithm [KMV 24a, probably earlier works too…]

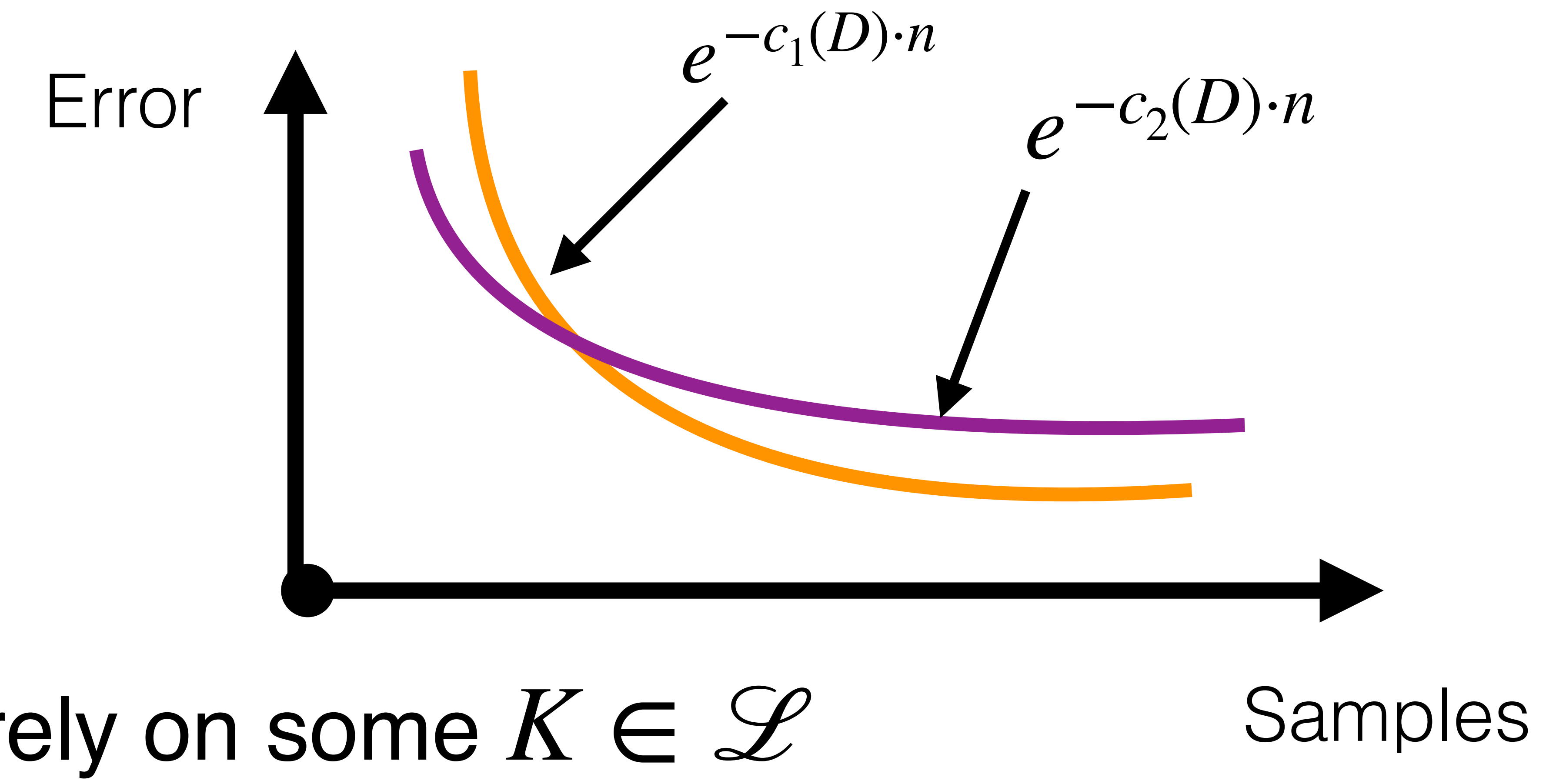
- How does the previous landscape change when we require stable generators?

Main Results III

Stable Generators	**No Hallucinations** $\lvert G(S_t) \setminus K \rvert = 0$	**Finite Hallucinations** $\lvert G(S_t) \setminus K \rvert < \infty$	**Infinite Hallucinations** $\lvert G(S_t) \setminus K \rvert = \infty$
Zero Missing Elements $\lvert K \setminus G(S_t) \rvert = 0$	Angluin's Condition [Ang 80] *(i.e., Exact Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections
Finite Missing Elements $\lvert K \setminus G(S_t) \rvert < \infty$	Angluin's Condition [Ang 80] *(i.e., Approximate Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections
Infinite Present Elements $\lvert K \cap G(S_t) \rvert = \infty$	Characterization ? *(Not all countable collections)*	Characterization ?	All Countable Collections

Learning Curves of Generation (with or without Breadth)

- Consider the following distributional setting
  - Countable domain $\mathscr{X}$
  - Countable collection of languages $\mathscr{L}$
  - Adversary picks a target distribution D supported entirely on some $K \in \mathscr{L}$
  - Learner gets as input n examples drawn i.i.d. from D and outputs some $G(S_n) \subseteq \mathscr{X}$
  - Error of the learner $\mathrm{er}(G(S_n) = 1\{G(S_n)$ does not satisfy the notion of breadth$\}$



- Main question: fixing some D and taking $n \to \infty$, how quickly does the error drop?

- [KMV 24a, 24b]: We provide a characterization of the shape of the learning curves for various notions of generation with breadth by establishing tight connections to the online setting

Yale

Outline of the Talk

- Motivation: CS and Language
- Theoretical Model
- Overview of our Definitions and Results
- <u>Overview of (some) Proofs</u>

Generation with Infinite Coverage and no Hallucinations

- Recall the algorithm of [KM'24]: Create chain of critical languages, output from the last one (whose index is at most t)

Critical languages $C_1^{(t)}, C_2^{(t)}, \ldots$ at time t:

- Consistency: Every $C_i^{(t)}$ contains the training set S_t enumerated so far
- Inclusions: $C_1^{(t)} \supseteq C_2^{(t)} \supseteq \ldots$, where $C_1^{(t)}$ is the first consistent language

This algorithm achieves infinite coverage but changes output infinitely often. Is this avoidable?

[Kalavasis, Mehrotra, V 2024b]

Theorem (informal)

Any algorithm that achieves generation with infinite coverage and no hallucinations for all countable $\mathscr{L}$ must be unstable

Immediate Corollary: The generator cannot know it is generating correctly

Generation with Infinite Coverage and no Hallucinations

[Kalavasis, Mehrotra, V 2024b]

Theorem (informal)

Any algorithm that achieves generation with infinite coverage and no hallucinations for all countable $\mathscr{L}$ must be unstable

Immediate Corollary: The generator cannot know it is generating correctly

Proof (Sketch):

- Let $\mathscr{X} = \mathbb{N}, L_i = \mathbb{N} \backslash \{i\}, \mathscr{L} = \{\mathbb{N}, L_1, L_2, \ldots\}$
- Pretend that $K = L_1$ and start enumerating $2,3,4,\ldots,$
- At some time t_1 the learner must output $G(S_{t_1})$ that doesn't contain 1 and contains some $i_1 > t_1 + 1$ (generation property)
- Pretend that $K = L_{i_1}$: keep enumerating until you hit i_1, enumerate 1 instead of i_1 and skip i_1
- ….

The construction guarantees that **(i)** either the algorithm doesn't generate correctly or **(ii)** changes infinitely often

Main Results II

	No Hallucinations $\lvert G(S_t)\setminus K \rvert = 0$	Finite Hallucinations $\lvert G(S_t)\setminus K \rvert < \infty$	Infinite Hallucinations $\lvert G(S_t)\setminus K \rvert = \infty$
Zero Missing Elements $\lvert K\setminus G(S_t) \rvert = 0$	Angluin's Condition [Ang 80] *(i.e., Exact Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections ✓
Finite Missing Elements $\lvert K\setminus G(S_t) \rvert < \infty$	Weak Angluin's Condition [KMV 24b, CP 24] *(i.e., Approximate Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections ✓
Infinite Present Elements $\lvert K \cap G(S_t) \rvert = \infty$	All Countable Collections ✓	All Countable Collections ✓	All Countable Collections ✓

Main Results III

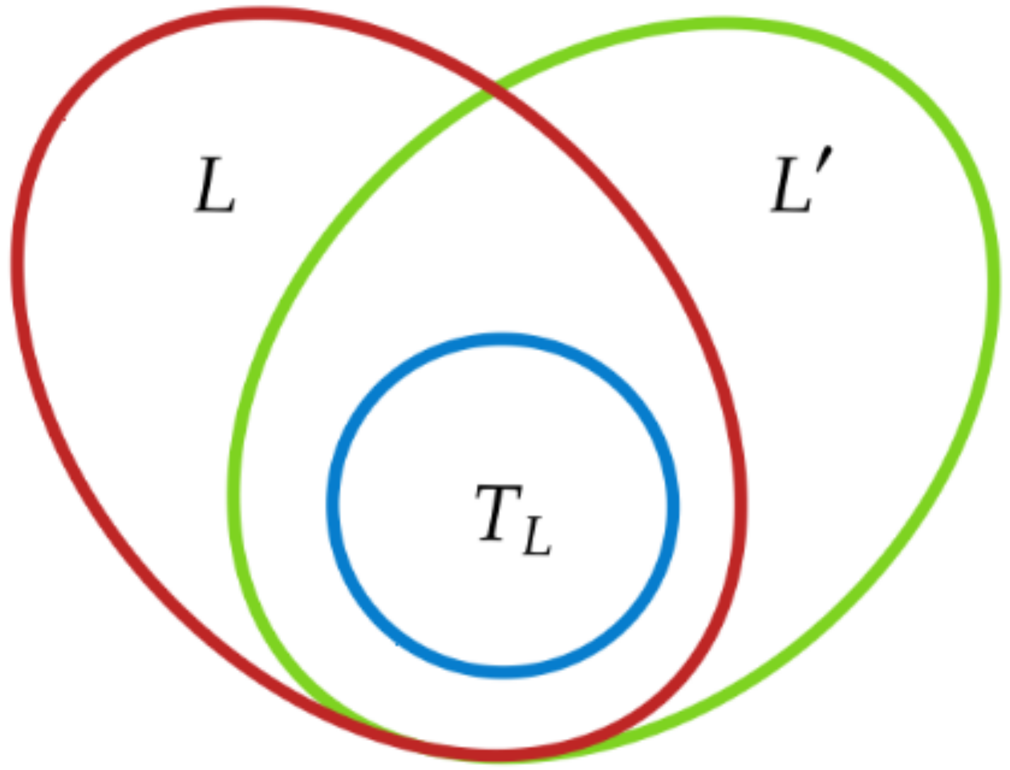
Stable Generators	**No Hallucinations** $\lvert G(S_t)\setminus K\rvert = 0$	**Finite Hallucinations** $\lvert G(S_t)\setminus K\rvert < \infty$	**Infinite Hallucinations** $\lvert G(S_t)\setminus K\rvert = \infty$
Zero Missing Elements $\lvert K\setminus G(S_t)\rvert = 0$	Angluin's Condition [Ang 80] *(i.e., Exact Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections ✓
Finite Missing Elements $\lvert K\setminus G(S_t)\rvert < \infty$	Angluin's Condition [Ang 80] *(i.e., Approximate Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections ✓
Infinite Present Elements $\lvert K\cap G(S_t)\rvert = \infty$	Characterization ? ✓ *(Not all countable collections)*	Characterization ?	All Countable Collections ✓

Background: Angluin's Condition

- Angluin completely characterized Gold's setting in 1980

[Angluin 1980]

Definition (informal):
A countable collection of languages $\mathscr{L}$ satisfies Angluin's condition if:
- For all $L \in \mathscr{L}$ there is some finite tell-tale subset $T_L \subseteq L$ for which the following holds:
- For all $L' \neq L$ either $T_L \nsubseteq L'$ or L' is not a proper subset of L



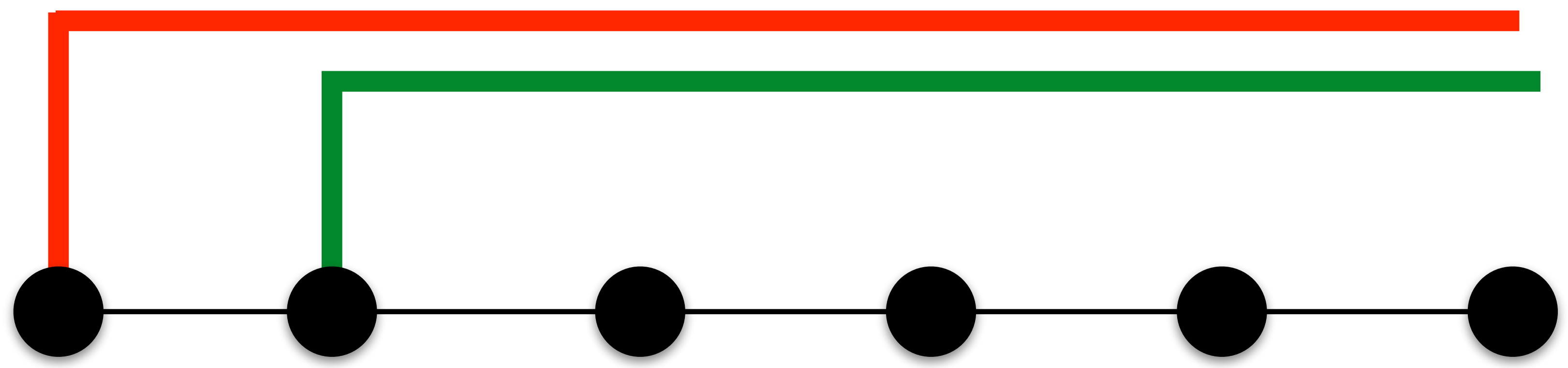
[Angluin 1980]

Theorem (informal):
A countable collection of languages $\mathscr{L}$ is identifiable in the limit if and only if it satisfies Angluin's condition

Yale

Example: Angluin's Condition

Consider the following setting [KM 24, CP 24]



- $\mathscr{X} = \mathbb{Z}$
- $L_i = \{-i, -i+1, -i+2, -i+3, \ldots\}$
- $\mathscr{L} = \{\mathbb{Z}, L_1, L_2, \ldots\}$
- $\mathscr{L}$ does not satisfy Angluin's condition:
  - Suffices to find some $L^* \in \mathscr{L}$ such that for all finite $T \subseteq L^*$ there exists some $L_T \in \mathscr{L}$:
  - $T \subseteq L_T$ and L_T is a proper subset of L^*
- Pick $L^* = \mathbb{Z}$.
- Consider any finite $T \subseteq \mathbb{Z}$ and let i_T be its smallest element
- Then, $T \subseteq L_{i_T}$ and $L_{i_T} \subsetneq L^*$

Generation with Exact Breadth and no Hallucinations

- Recall our goal is to achieve $G(S_t) = K$

[Kalavasis, Mehrotra, V 2024b, Charikar, Pabbaraju 2024]

Theorem (informal):
A countable collection of languages $\mathscr{L}$ is generatable with exact breadth and no hallucinations in the limit if and only if satisfies Angluin's condition

- Same result shown by [CP'24]

- The algorithm of [KM 24] achieves exact breadth with no hallucinations iff $\mathscr{L}$ satisfies Angluin's condition

[Kalavasis, Mehrotra, V 2024a]

Theorem (informal):
If a countable collection of languages $\mathscr{L}$ satisfies Angluin's condition, then the algorithm of [KM 24] achieves generation with exact breadth and no hallucinations in the limit

- Main idea: At some point $T_K \subseteq S_t$. Then, for all $L \neq K$ either $S_t \not\subseteq L$ or $L \not\subseteq K$, so no language after K is critical

Lower Bound Construction

- We provide a general construction which shows that every notion of breadth that satisfies a certain "uniqueness" property cannot be achieved if $\mathscr{L}$ does not satisfy Angluin's condition

- Uniqueness property: we say that a notion of breadth satisfies the uniqueness property if every generator can satisfy this property for at most one language at a time
  - e.g., exact breadth satisfies the uniqueness property

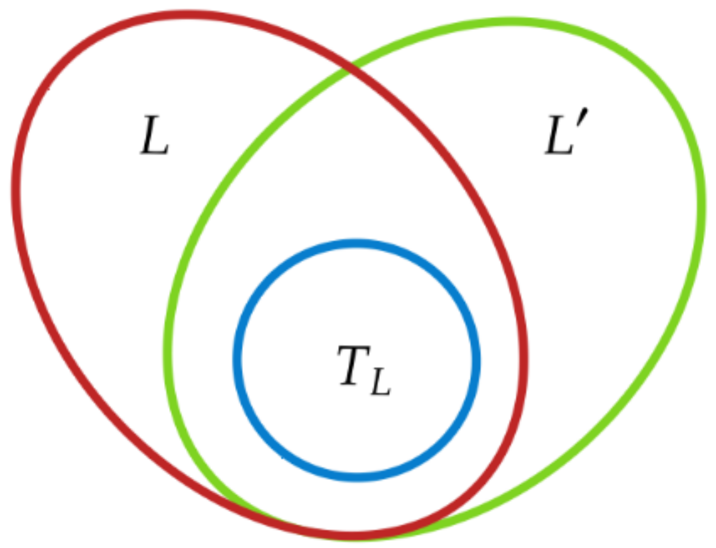
[Kalavasis, Mehrotra, V 2024a]

Theorem (informal):
If a notion of breadth satisfies the uniqueness property, then no algorithm can generate from $\mathscr{L}$ in a way that satisfies this notion of the breadth if $\mathscr{L}$ does not satisfy Angluin's condition

Lower Bound Construction

[Kalavasis, Mehrotra, V 2024a]

Theorem (informal):
If a notion of breadth satisfies the uniqueness property, then no algorithm can generate from $\mathscr{L}$ in a way that satisfies this notion of the breadth if $\mathscr{L}$ does not satisfy Angluin's condition



Proo (Sketch):

- Since $\mathscr{L}$ does not satisfy Angluin's condition there exists $L^* \in \mathscr{L}$ such that for all finite $T \subseteq L^*$ there is some $L_T \in \mathscr{L}$ with $T \subseteq L_T$ and $L_T \subsetneq L^*$
- Pretend that $K = L^*$ and start enumerating it
- At some time t_1 the generator must satisfy the notion of breadth for L^*
- Pretend that $K = L_{S_{t_1}}$ and continue the enumeration to one of $L_{S_{t_1}}$ (this can be achieved)
- At some time $t_2 > t_1$ the generator must satisfy the notion of breadth for $L_{S_{t_1}}$ (so not for L^*)
- Pretend that $K = L^*$ and continue with an enumeration of L^*….

Main Results II

	No Hallucinations $\lvert G(S_t) \setminus K \rvert = 0$	**Finite Hallucinations** $\lvert G(S_t) \setminus K \rvert < \infty$	**Infinite Hallucinations** $\lvert G(S_t) \setminus K \rvert = \infty$
Zero Missing Elements $\lvert K \setminus G(S_t) \rvert = 0$	Angluin's Condition [Ang 80] ✓ *(i.e., Exact Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections ✓
Finite Missing Elements $\lvert K \setminus G(S_t) \rvert < \infty$	Weak Angluin's Condition [KMV 24b, CP 24] *(i.e., Approximate Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections ✓
Infinite Present Elements $\lvert K \cap G(S_t) \rvert = \infty$	All Countable Collections ✓	All Countable Collections ✓	All Countable Collections ✓

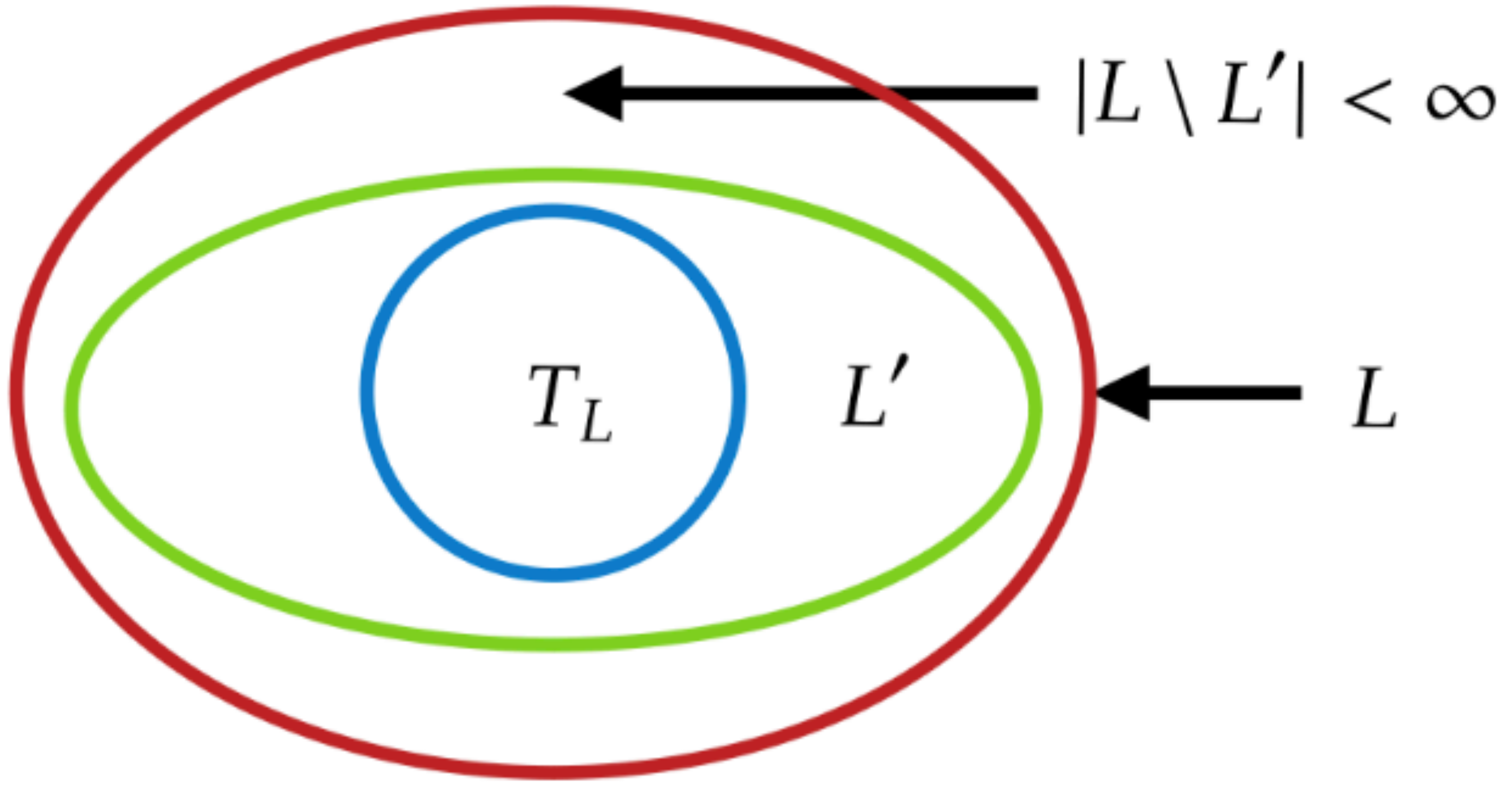
Yale

Weak Angluin's Condition

- Intuition for relaxation: it is easier to handle proper subsets of L that miss only finitely many elements of L than subsets that miss infinitely many elements

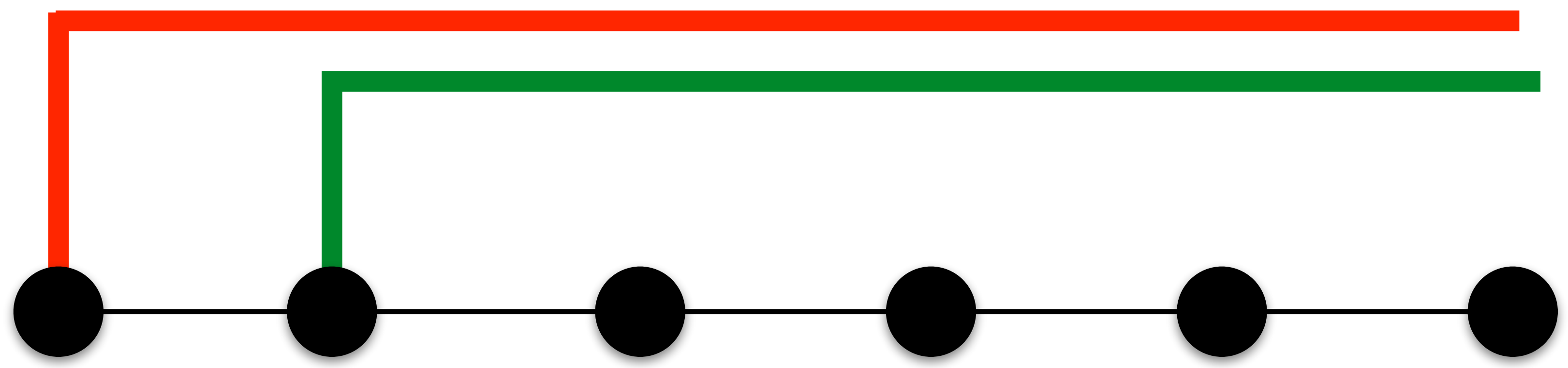
[Kalavasis, Mehrotra, V 2024b, Charikar, Pabbaraju 2024]

Definition (informal):
A countable collection of languages $\mathscr{L}$ satisfies the weak Angluin's condition if:
- For all $L \in \mathscr{L}$ there is some finite tell-tale subset $T_L \subseteq L$ for which the following holds:
- For all $L' \neq L$ either $T_L \nsubseteq L'$ or L' is not a proper subset of L or L' is a proper subset of L with $|L \backslash L'| < \infty$



Example 1: Weak Angluin's Condition

Consider the following setting [KM 24, CP 24]



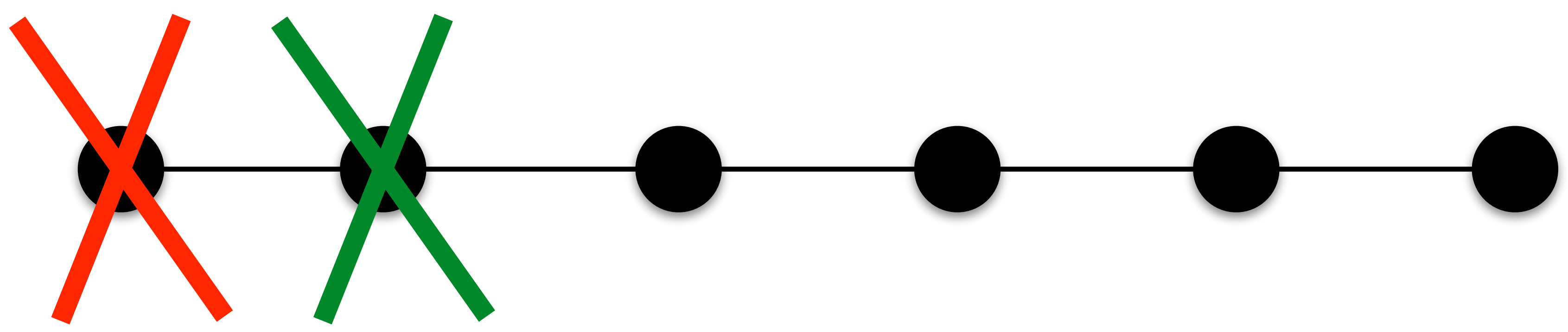
- $\mathscr{X} = \mathbb{Z}$
- $L_i = \{-i, -i+1, -i+2, -i+3, \ldots\}$
- $\mathscr{L} = \{\mathbb{Z}, L_1, L_2, \ldots\}$
- $\mathscr{L}$ does not satisfy the weak Angluin's condition:
  - Suffices to find some $L^* \in \mathscr{L}$ such that for all finite $T \subseteq L^*$ there exists some $L_T \in \mathscr{L}$:
  - $T \subseteq L_T$, L_T is a proper subset of L^*, and $|L^* \backslash L_{i_T}| = \infty$

- Pick $L^* = \mathbb{Z}$.
- Consider some finite $T \subseteq \mathbb{Z}$ and let i_T be its smallest element
- Then, $T \subseteq L_{i_T}$ and $L_{i_T} \subsetneq L^*$ and $|L^* \backslash L_{i_T}| = \infty$

Yale

Example 2: Weak Angluin's Condition

Consider the following setting [KM 24, CP 24]



- $\mathscr{X} = \mathbb{N}$
- $L_i = \mathbb{N} \setminus \{i\}$
- $\mathscr{L} = \{L_0 = \mathbb{N}, L_1, L_2, \ldots\}$
- $\mathscr{L}$ satisfies the weak Angluin's condition: choose $T_i = \{i+1\}, i \geq 0$
- Notice that for all i, j it holds that $|L_i \setminus L_j| \leq 2$, hence the condition is satisfied

Yale

Generation with Approximate Breadth and no Hallucinations

- Recall our goal is to achieve $|K \backslash G(S_t)| < \infty, G(S_t) \subseteq K$

[Kalavasis, Mehrotra, V 2024b, Charikar, Pabbaraju 2024]

Theorem (informal):
A countable collection of languages $\mathscr{L}$ is generatable with approximate breadth and no hallucinations in the limit if and only if it satisfies the weak Angluin's condition

- It is not always easy to check if $\mathscr{L}$ satisfies either Angluin's condition or the weak Angluin's condition

- Hence, it is useful to have an algorithm that achieves best-of-three-worlds

[Kalavasis, Mehrotra, V 2024b]

Theorem (informal):
The following holds for the algorithm of [KM 24]
- If $\mathscr{L}$ satisfies Angluin's condition then it generates with exact breadth and no hallucinations
- If $\mathscr{L}$ satisfies the weak Angluin's condition then it generates with approximate breadth and no hallucinations
- If $\mathscr{L}$ does not satisfy the weak Angluin's condition then it generates with infinite coverage and no hallucinations

Lower Bound Construction

- We provide a general construction which shows that every notion of breadth that satisfies a certain "finite non-uniqueness" property cannot be achieved if $\mathscr{L}$ does not satisfy the weak Angluin's condition

- Finite non-uniqueness property: we say that a notion of breadth satisfies the finite non-uniqueness property if a generator can satisfy this property simultaneously for two languages only if they differ on finitely many elements
  - e.g., approximate breadth satisfies the uniqueness property

[Kalavasis, Mehrotra, V 2024a]

Theorem (informal):
If a notion of breadth satisfies the finite non-uniqueness property, then no algorithm can generate from $\mathscr{L}$ in a way that satisfies this notion of the breadth if $\mathscr{L}$ does not satisfy the weak Angluin's condition

- Construction: modification of the "uniqueness"-based construction

Yale

Main Results II

	No Hallucinations $\lvert G(S_t)\backslash K \rvert = 0$	Finite Hallucinations $\lvert G(S_t)\backslash K \rvert < \infty$	Infinite Hallucinations $\lvert G(S_t)\backslash K \rvert = \infty$
Zero Missing Elements $\lvert K\backslash G(S_t) \rvert = 0$	Angluin's Condition [Ang 80] ✓ *(i.e., Exact Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24]	All Countable Collections ✓
Finite Missing Elements $\lvert K\backslash G(S_t) \rvert < \infty$	Weak Angluin's Condition [KMV 24b, CP 24] ✓ *(i.e., Approximate Breadth)*	Weak Angluin's Condition [KMV 24b, CP 24] ✓	All Countable Collections ✓
Infinite Present Elements $\lvert K \cap G(S_t) \rvert = \infty$	All Countable Collections ✓	All Countable Collections ✓	All Countable Collections ✓

Yale

Generation with Zero Missing Element and Finite Hallucinations

- Recall our goal is to achieve $|G(S_t)\backslash K| < \infty, G(S_t) \supseteq K$

[Kalavasis, Mehrotra, V 2024b, Charikar, Pabbaraju 2024]

Theorem (informal):
A countable collection of languages $\mathscr{L}$ is generatable with finite hallucinations and zero missing breadth in the limit if and only if it satisfies the weak Angluin's condition

- For this definition, we can achieve stable generation

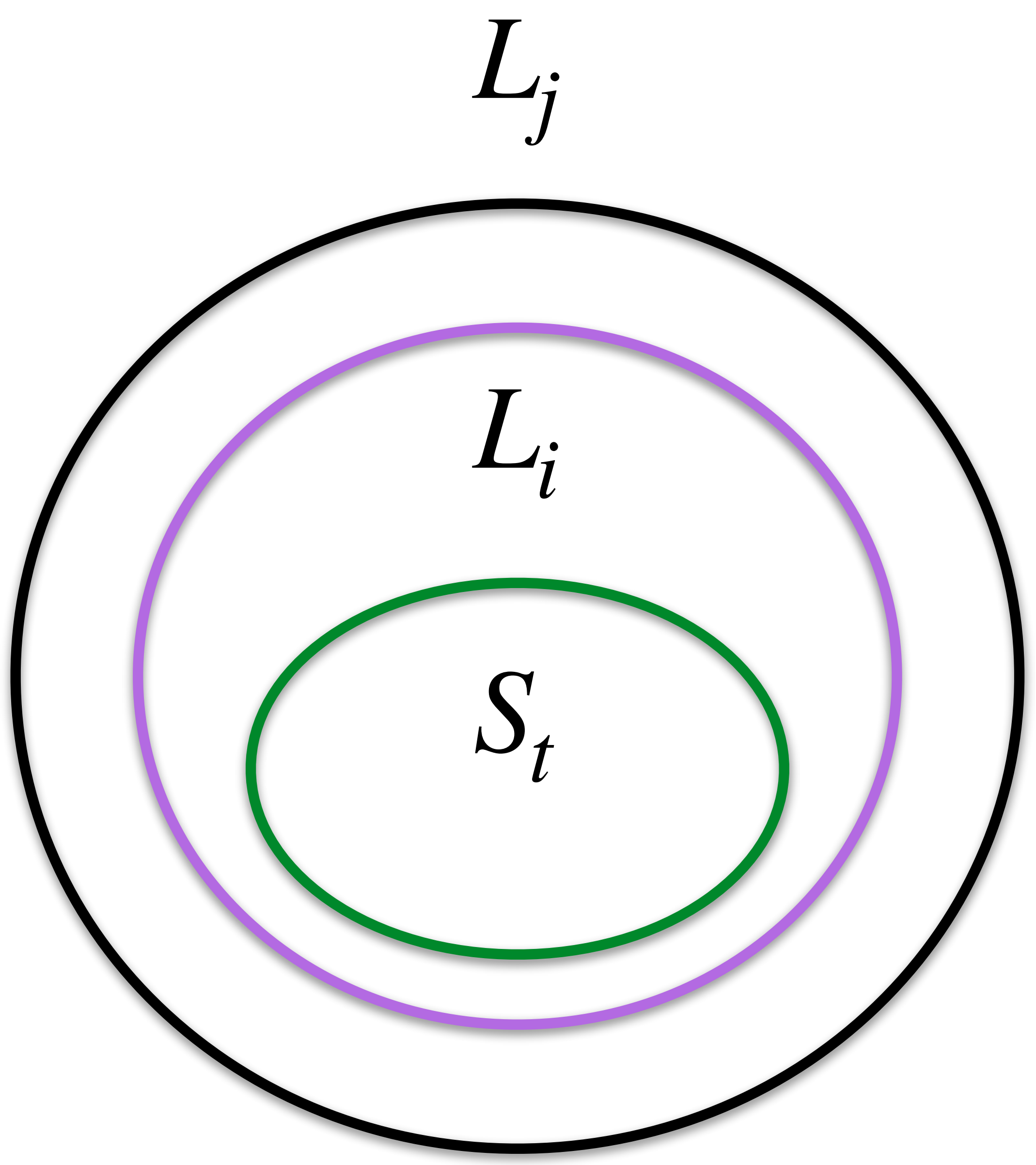
[Kalavasis, Mehrotra, V 2024b, Charikar, Pabbaraju 2024]

Theorem (informal):
A countable collection of languages $\mathscr{L}$ is generatable by a stable generator with finite hallucinations and zero missing breadth in the limit if and only if it satisfies the weak Angluin's condition

Generation with Zero Missing Element and Finite Hallucinations

- The algorithm is based on a modification of [KM 24]

- Recall the algorithm of [KM 24]. In every round t do the following:
  - Consider only the first t languages: $\mathscr{L}_t = \{L_1, \ldots, L_t\}$
  - Create the set of critical languages C_t within $\mathscr{L}_t$
  - Output the largest indexed language in C_t



- Modification: In every round t do the following:
  - Consider only the first t languages: $\mathscr{L}_t = \{L_1, \ldots, L_t\}$
  - Create the set of critical languages C_t within $\mathscr{L}_t$
  - Let L^* be the largest indexed language in C_t
  - Create the set F_t of languages in $L \in C_t$ that satisfy $|L \backslash L^*| < \infty$ (requires new oracle)
  - Output the smallest indexed language in F_t

Lower Bound

- Follows immediately from the “finite non-uniqueness” construction since this notion of breadth satisfies the finite non-uniqueness condition

Main Results II

	No Hallucinations $\lvert G(S_t) \setminus K \rvert = 0$	Finite Hallucinations $\lvert G(S_t) \setminus K \rvert < \infty$	Infinite Hallucinations $\lvert G(S_t) \setminus K \rvert = \infty$
Zero Missing Elements $\lvert K \setminus G(S_t) \rvert = 0$	Angluin's Condition [Ang 80] *(i.e., Exact Breadth)* ✓	Weak Angluin's Condition [KMV 24b, CP 24] ✓	All Countable Collections ✓
Finite Missing Elements $\lvert K \setminus G(S_t) \rvert < \infty$	Weak Angluin's Condition [KMV 24b, CP 24] *(i.e., Approximate Breadth)* ✓	Weak Angluin's Condition [KMV 24b, CP 24] ✓	All Countable Collections ✓
Infinite Present Elements $\lvert K \cap G(S_t) \rvert = \infty$	All Countable Collections ✓	All Countable Collections ✓	All Countable Collections ✓

(Immediate) Next Directions

- Extension of validity vs. breadth trade-off to the prompted generation setting of [KM 24]
- Complete the characterization of stable generation
- Extension to the “agnostic” setting where the adversary can give incorrect information [RR 25]
- Weakening of the definition (for all target languages, for all enumerations…)
  - For some collections, we can achieve validity and breadth for all except for one target language
  - Allow the learner to generate more than one texts (similar to what LLMs are doing)
- More fine-grained versions of the trade-off
  - Subsequently, Kleinbeg and Wei [KW 25] studied such versions based on a notion of “density”
- Computationally efficient algorithms for more structured settings

Conclusion

- In the era of LLMs, one of the contributions TCS can make is to provide the right *definitions* and *abstractions* to study their behavior, and formally argue about their abilities and limitations
  - In a similar spirit as in fairness, clustering, distributed systems,…

- Kleinberg and Mullainathan [KM 24] proposed an abstract model for generation and showed that generation is a sharply different problem from identification

- [KM 24] initiated the discussion about the tension between validity and breadth

- Our works and others have provided several formal notions of breadth and showed a provable tension between validity and breadth

- How can we circumvent the impossibility results?
  - A different set of our result shows that *negative* examples help (i.e., elements *not* in K,) which is also observed in practice (e.g., negative example through RLHF)
  - Other type of useful information?

Thank You!

- COLT 2025 Tutorial on “Language Generation in the Limit” [Charikar, Mehrotra, Pabbaraju, Peale, V.]