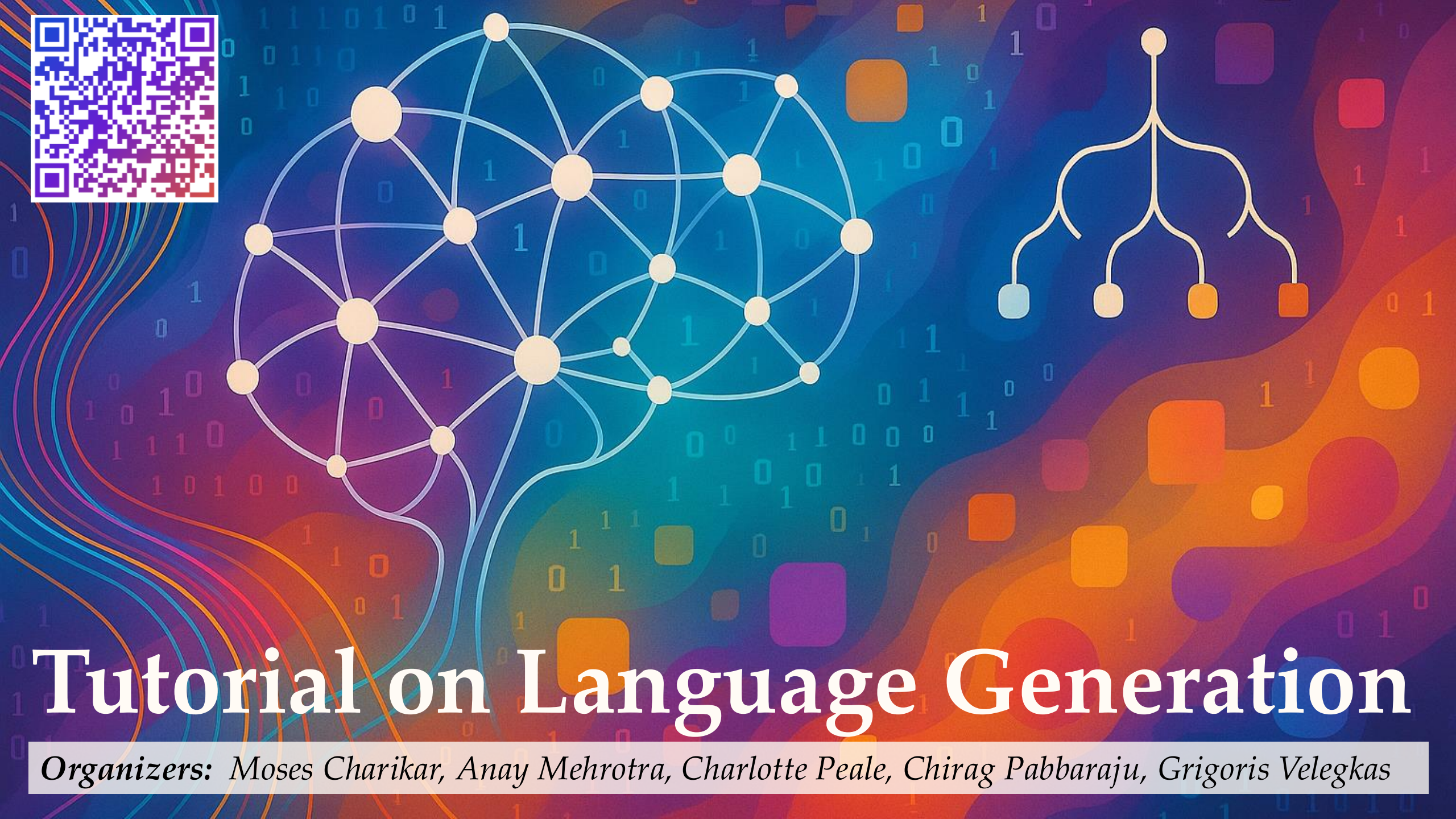


Tutorial on Language Generation

Organizers: *Moses Charikar, Anay Mehrotra, Charlotte Peale, Chirag Pabbaraju, Grigoris Velegkas*

Schedule

Session 1 (40/20 mins)

- *Language Generation and Learning Theory* — *Moses*
- *Stronger Notions of Generation and Comparison to Prediction* — *Chirag*

Session 2 (20/20/20 mins)

- *Validity–Breadth Trade-Off (Part I)* — *Anay*
- *Validity–Breadth Trade-Off (Part II)* — *Grigoris*
- *Diverse and Robust Generation* — *Charlotte*

Tutorial on Language Generation

Visit: LanguageGeneration.github.io

Organizers:

Moses Charikar
Stanford

Anay Mehrotra
Yale University

Chirag Pabbaraju
Stanford

Charlotte Peale
Stanford

Grigoris Velegkas
Yale → Google Research

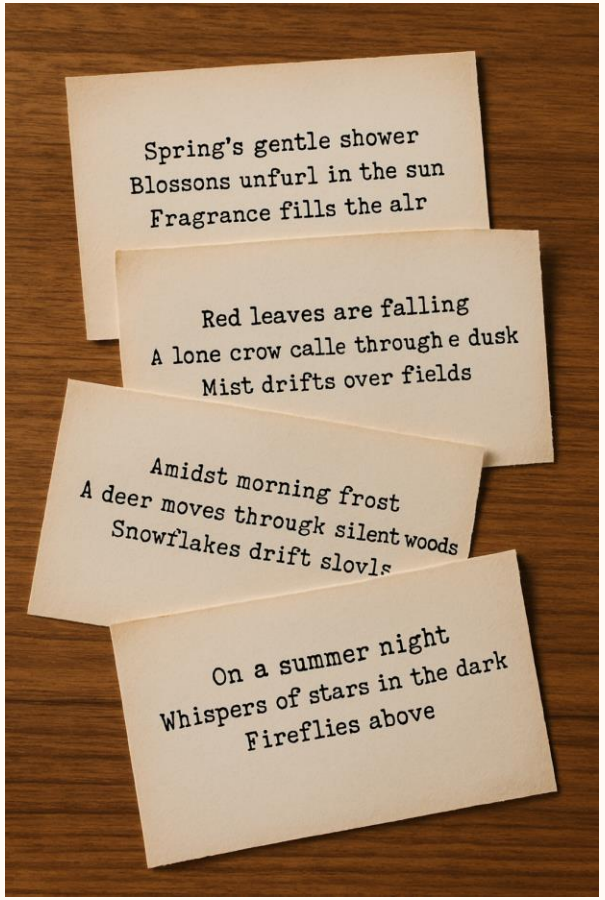
COLT 2025

LLMs and Language Generation 

Computer scientists have been fascinated by language acquisition by humans and machines for decades

LLMs: what is the problem?

Kleinberg, Mullainathan, 2024



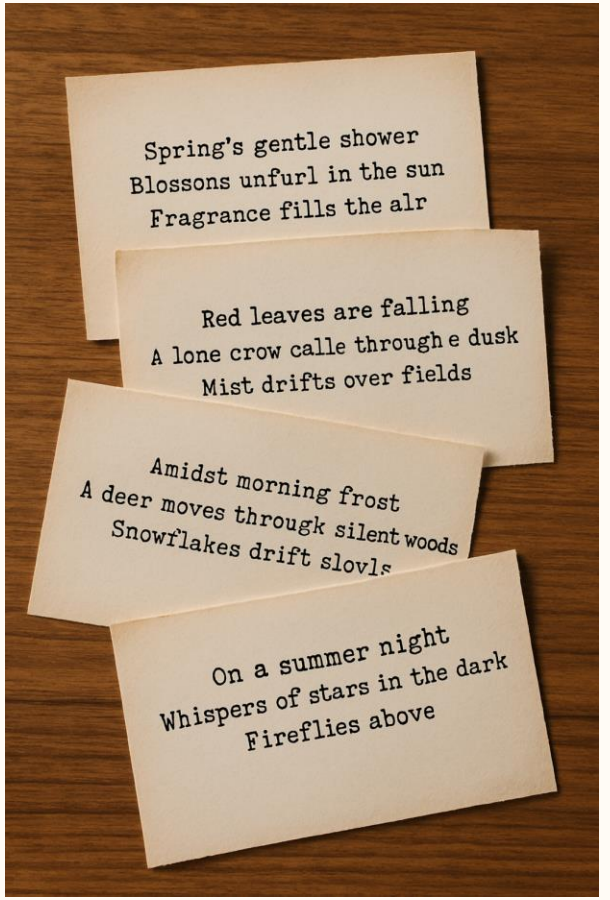
LLMs: what is the problem?

Kleinberg, Mullainathan, 2024

From a large collection of text:

1. assign probabilities to every sequence of words

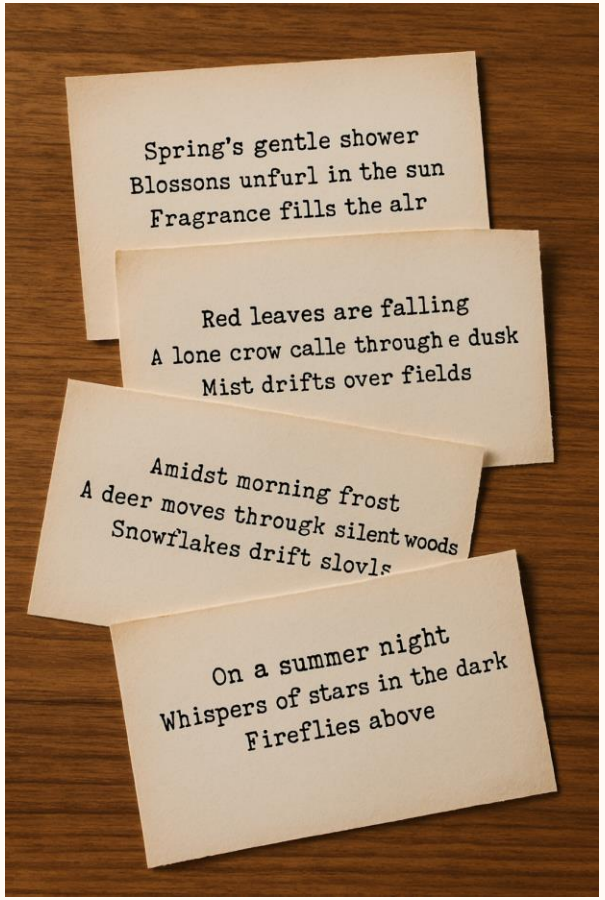
2. produce strings with high assigned probability



LLMs: simplest problem statement?

Kleinberg, Mullainathan, 2024

Given strings from an unknown language,
produce valid unseen strings from the language.



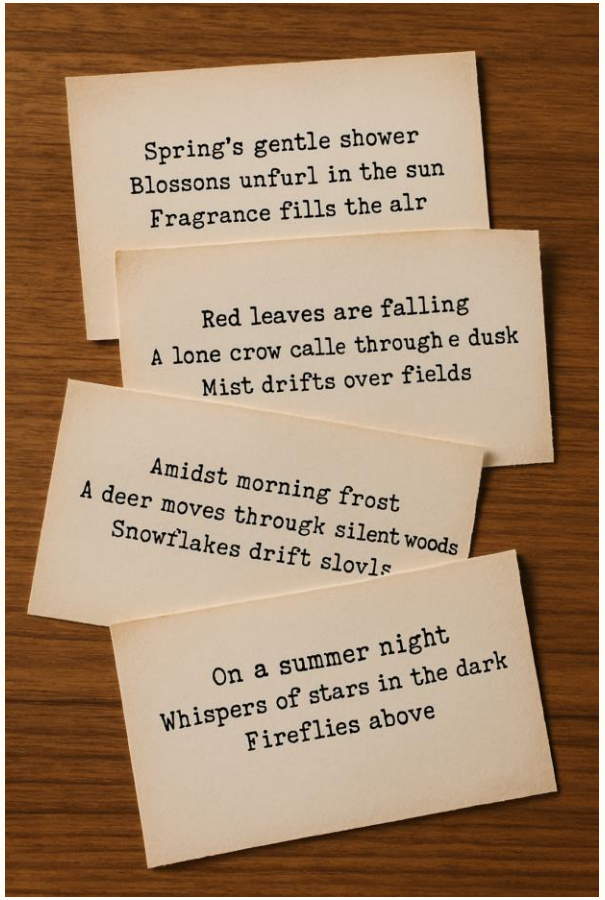
LLMs: simplest problem statement?

Kleinberg, Mullainathan, 2024

Given strings from an unknown language,
produce valid unseen strings from the language.

string: sequence of symbols

language: set of strings
e.g. C programs



Language Identification in the Limit

E Mark Gold*

Language Identification in the Limit

E Mark Gold*

I wish to construct a precise model for "able to speak English"...
to investigate theoretically how it can be achieved artificially

Language Identification in the Limit

E Mark Gold*

I wish to construct a precise model for "able to speak English"…
to investigate theoretically how it can be achieved artificially

Since we cannot explicitly write down the rules of English…
artificial intelligence… will have to learn… from implicit data…

Language Identification

Gold 1967, Angluin 1979

Game between adversary and algorithm.

Language Identification

Gold 1967, Angluin 1979

Game between adversary and algorithm.

Adversary thinks of target language K from countable list
(e.g. all context-free grammars)

Language Identification

Gold 1967, Angluin 1979

Game between adversary and algorithm.

Adversary thinks of target language K from countable list
(e.g. all context-free grammars)

Adversary enumerates strings one by one

Language Identification

Gold 1967, Angluin 1979

Game between adversary and algorithm.

Adversary thinks of target language K from countable list
(e.g. all context-free grammars)

Adversary enumerates strings one by one

In each step, algorithm guesses index i_t (goal: $L_{i_t} = K$)

Language Identification

Gold 1967, Angluin 1979

Game between adversary and algorithm.

Adversary thinks of target language K from countable list
(e.g. all context-free grammars)

Adversary enumerates strings one by one

In each step, algorithm guesses index i_t (goal: $L_{i_t} = K$)

Success: guess correct for every $t > t'$
(We say that algorithm has identified K in the limit)

Language Identification

Gold 1967, Angluin 1979

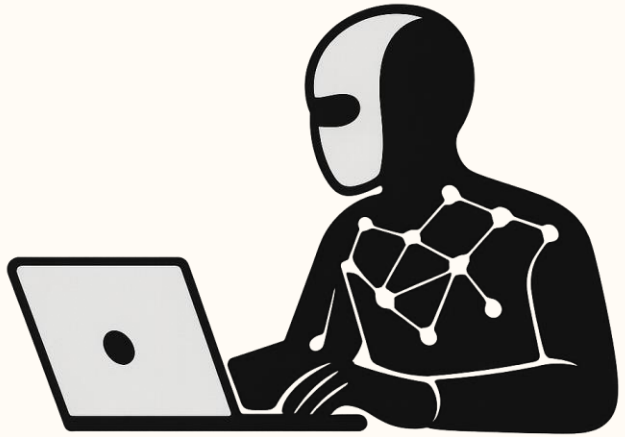
string of K

guess for index i

string of K

guess for index i

…



Language Identification

Gold 1967, Angluin 1979

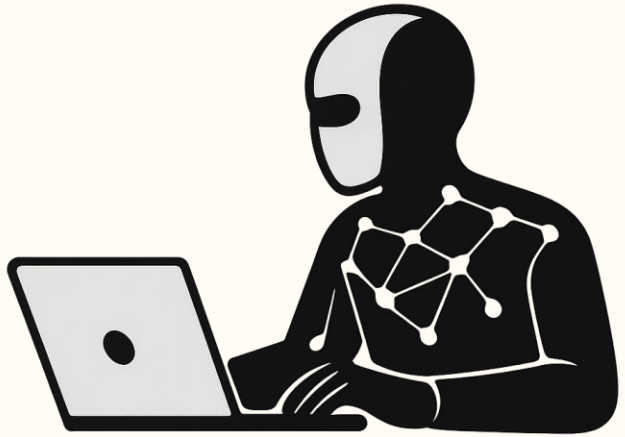
string of K

guess for index i

string of K

guess for index i

…



Algorithm never sees string not in K

Language Identification

Gold 1967, Angluin 1979

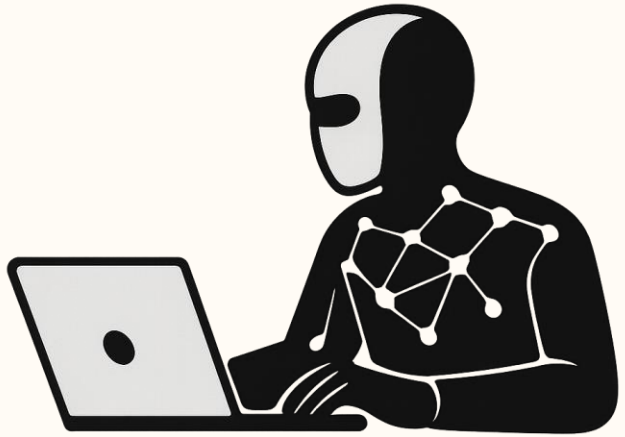
string of K

guess for index i

string of K

guess for index i

…



Algorithm never sees string not in K

Never told whether guess is correct

Language Identification

Gold 1967, Angluin 1979

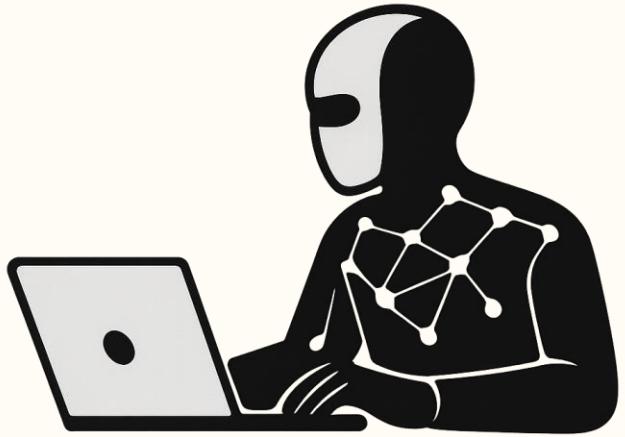
string of K

guess for index i

string of K

guess for index i

…



Algorithm never sees string not in K

Never told whether guess is correct

Cannot ask if string is in K

Language Identification in the Limit

Theorem [Gold 1967]
Language identification in the limit is impossible
even for simple collections such as all regular languages

Language Identification in the Limit

Theorem [Gold 1967]
Language identification in the limit is impossible
even for simple collections such as all regular languages

Angluin 1979, 1980

Precise characterization of language collections where identification possible

Language Identification in the Limit

Theorem [Gold 1967]
Language identification in the limit is impossible
even for simple collections such as all regular languages

Angluin 1979, 1980

Precise characterization of language collections where identification possible

Given strings from an unknown language,
produce valid unseen strings from the language.

Language Generation

Kleinberg, Mullainathan, 2024

New game between adversary and algorithm.

Language Generation

Kleinberg, Mullainathan, 2024

New game between adversary and algorithm.

Adversary thinks of target language K from countable list

Language Generation

Kleinberg, Mullainathan, 2024

New game between adversary and algorithm.

Adversary thinks of target language K from countable list

Adversary enumerates strings one by one

$S_t \in K$: strings enumerated up to time t

Language Generation

Kleinberg, Mullainathan, 2024

New game between adversary and algorithm.

Adversary thinks of target language K from countable list

Adversary enumerates strings one by one
$S_t \in K$: strings enumerated up to time t

In each step, algorithm guesses string a_t (goal: $a_t \in K \setminus S_t$)

Language Generation

Kleinberg, Mullainathan, 2024

New game between adversary and algorithm.

Adversary thinks of target language K from countable list

Adversary enumerates strings one by one
$S_t \in K$: strings enumerated up to time t

In each step, algorithm guesses string a_t (goal: $a_t \in K \setminus S_t$)

Success: guess correct for every $t > t'$
(We say that algorithm has generated K in the limit)

Language Generation

Kleinberg, Mullainathan, 2024

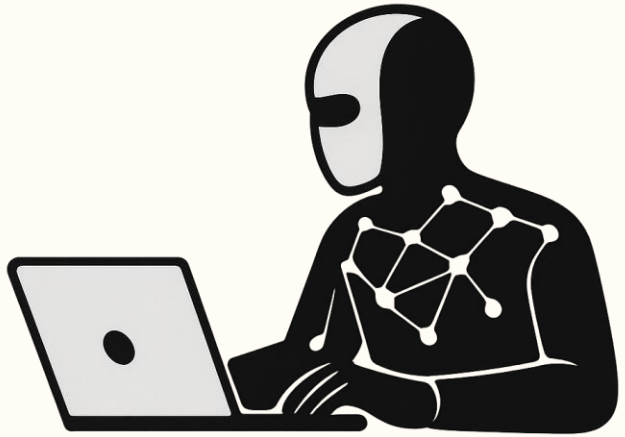
string of K

guess new string from K

string of K

guess new string from K

…



Language Generation

Kleinberg, Mullainathan, 2024

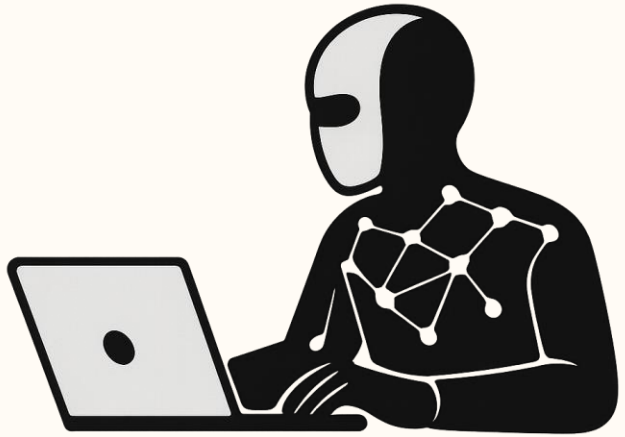
string of K

guess new string from K

string of K

guess new string from K

…



Algorithm never sees negative examples

Language Generation

Kleinberg, Mullainathan, 2024

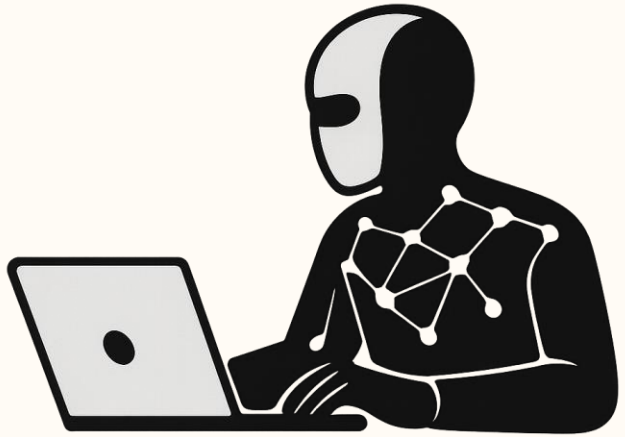
string of K

guess new string from K

string of K

guess new string from K

…



Algorithm never sees negative examples

No feedback

Language Generation

Kleinberg, Mullainathan, 2024

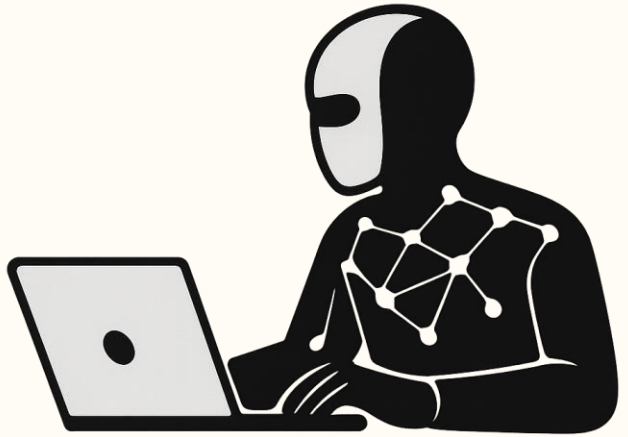
string of K

guess new string from K

string of K

guess new string from K

…



Algorithm never sees negative examples

No feedback

Assume all languages infinite

Language Generation in the Limit

Theorem [Kleinberg, Mullainathan 2024]
Language generation in the limit is possible
for any countable collection of languages

Language Generation in the Limit

Theorem [Kleinberg, Mullainathan 2024]
Language generation in the limit is possible for any countable collection of languages

Generation vs Identification

After seeing many C programs

(generation) output valid C programs

(identification) output valid grammar for C

Language Generation in the Limit

Theorem [Kleinberg, Mullainathan 2024]
Language generation in the limit is possible
for any countable collection of languages

Language Generation in the Limit

Theorem [Kleinberg, Mullainathan 2024]
Language generation in the limit is possible for any countable collection of languages

Algorithm only needs to generate from infinite subset of K

Language Generation in the Limit

Theorem [Kleinberg, Mullainathan 2024]
Language generation in the limit is possible for any countable collection of languages

Algorithm only needs to generate from infinite subset of K

validity (only generate valid strings) vs
breadth (large subset of K) tradeoff:

Language Generation in the Limit

Theorem [Kleinberg, Mullainathan 2024]
Language generation in the limit is possible for any countable collection of languages

Algorithm only needs to generate from infinite subset of K

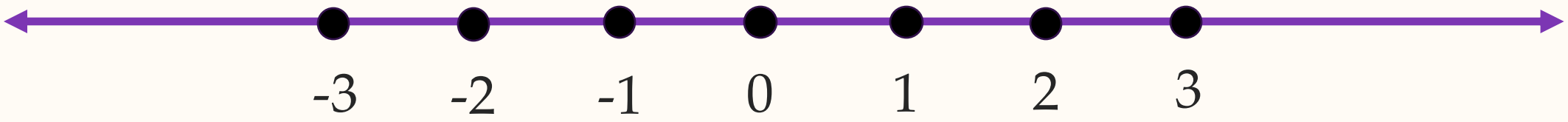
validity (only generate valid strings) vs
breadth (large subset of K) tradeoff:

hallucination vs mode-collapse

Language Identification Lower Bound

$$L_i = \{-i, -i+1, -i+2, \ldots\}$$

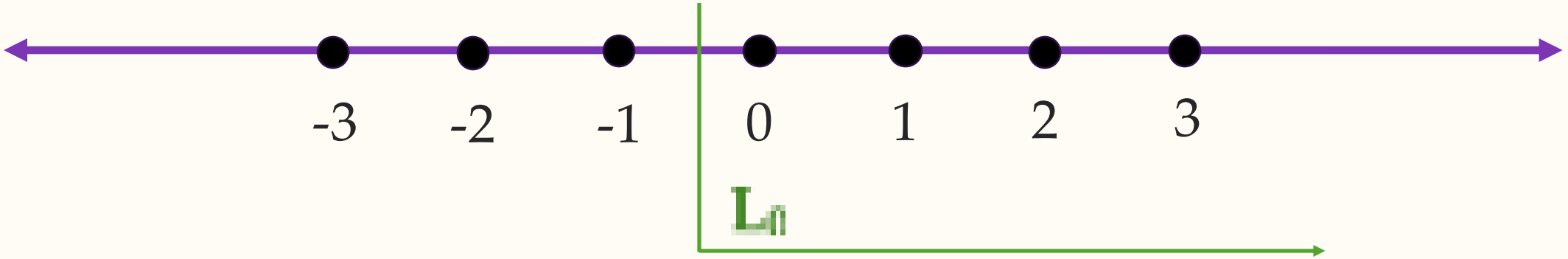
$$L_\infty = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$$



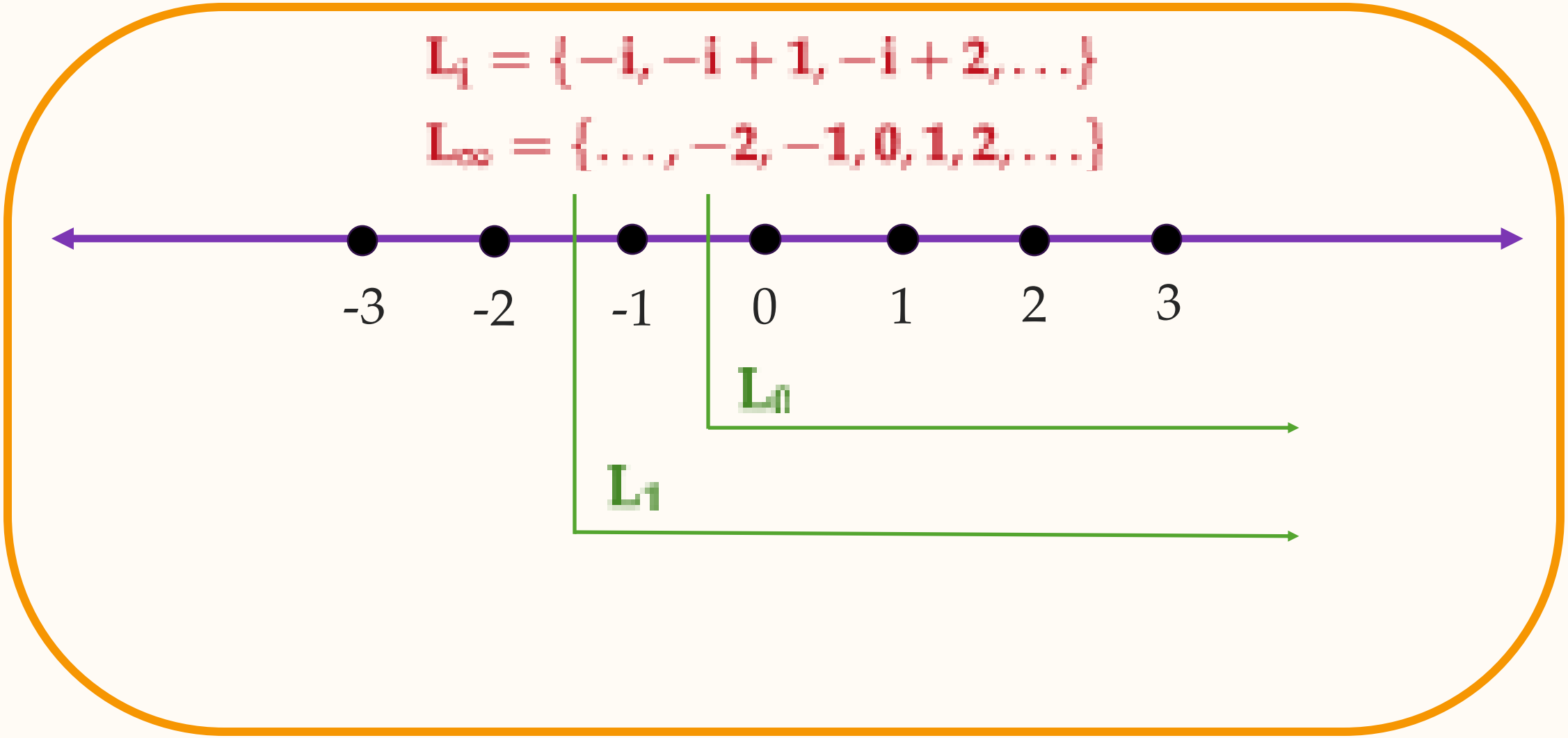
Language Identification Lower Bound

$L_i = \{-i, -i+1, -i+2, \ldots\}$

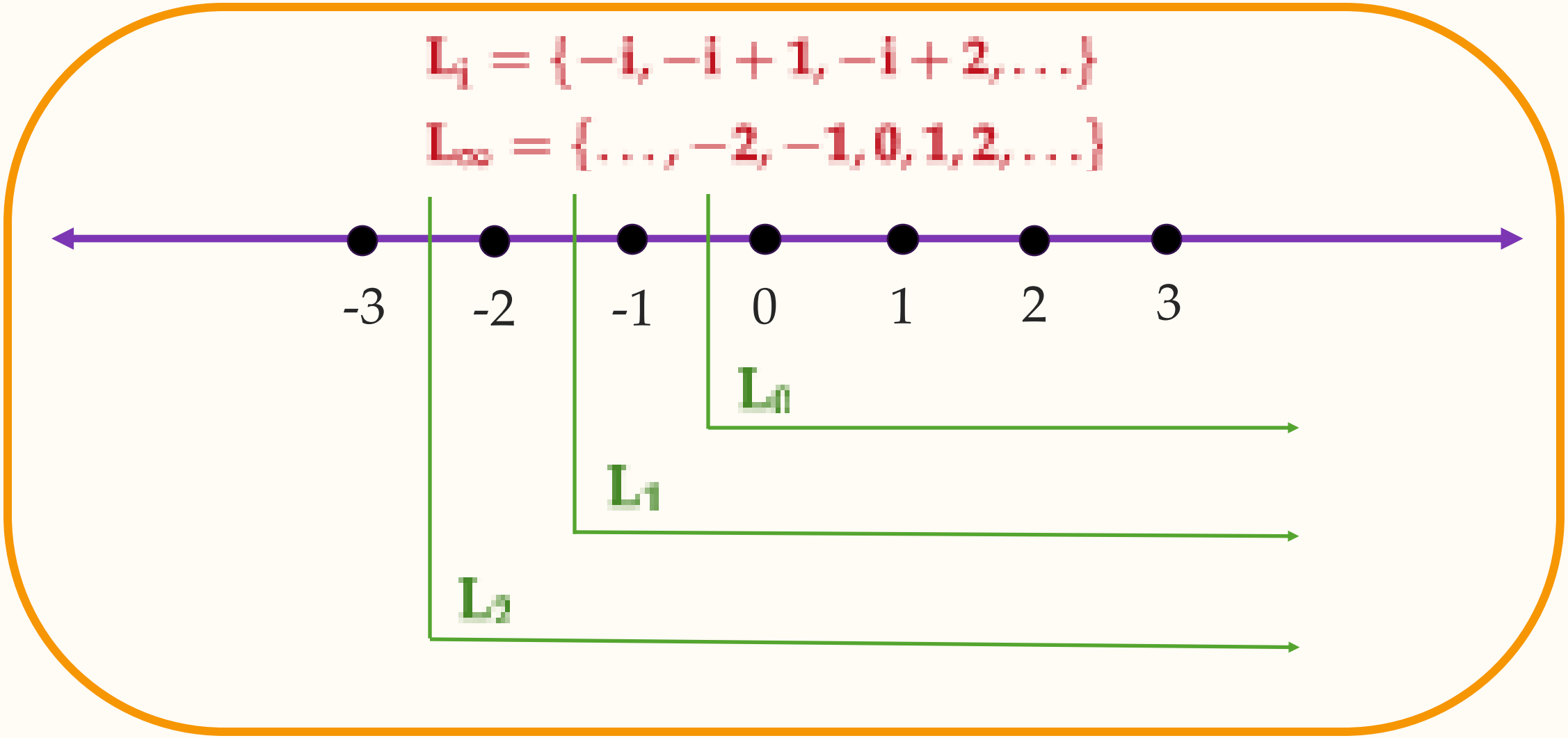
$L_\infty = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$



Language Identification Lower Bound



Language Identification Lower Bound



Closure operation

$\langle S \rangle$: intersection of all languages containing S

Output an element of $\langle S_t \rangle \setminus S_t$?

Closure operation

$\langle S \rangle$: intersection of all languages containing S

Output an element of $\langle S_t \rangle \setminus S_t$?

Problem: $\langle S_t \rangle \setminus S_t$ can be empty for all t !

Closure operation

$\langle S \rangle$: intersection of all languages containing S

Output an element of $\langle S_t \rangle \setminus S_t$?

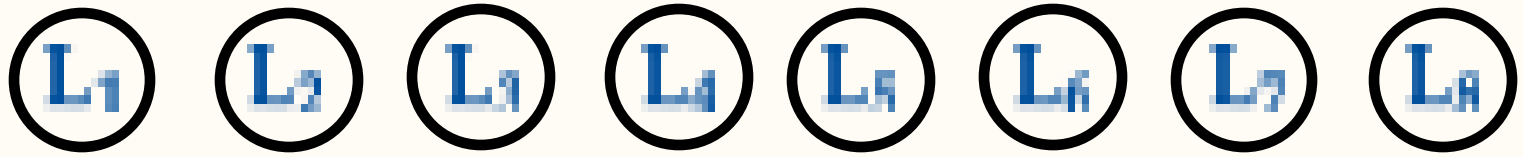
Problem: $\langle S_t \rangle \setminus S_t$ can be empty for all t !

e.g. modify previous example

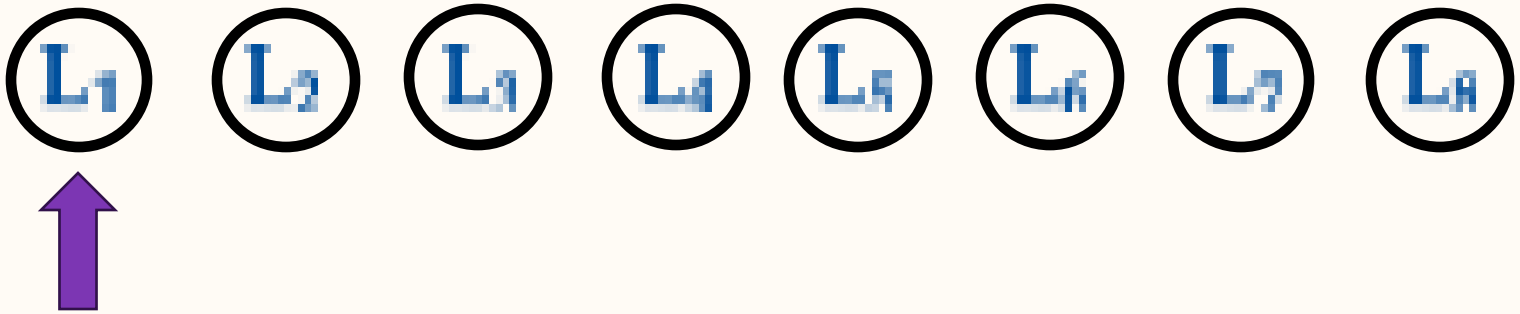
$$L_{t,V} = V \cup \{-t, -t+1, -t+2, \ldots\}$$

where V ranges over all finite sets

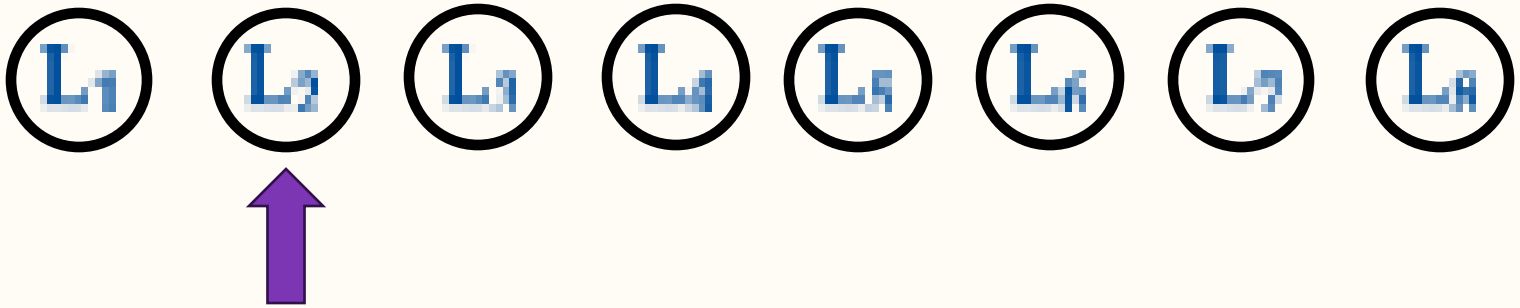
Natural language elimination strategy



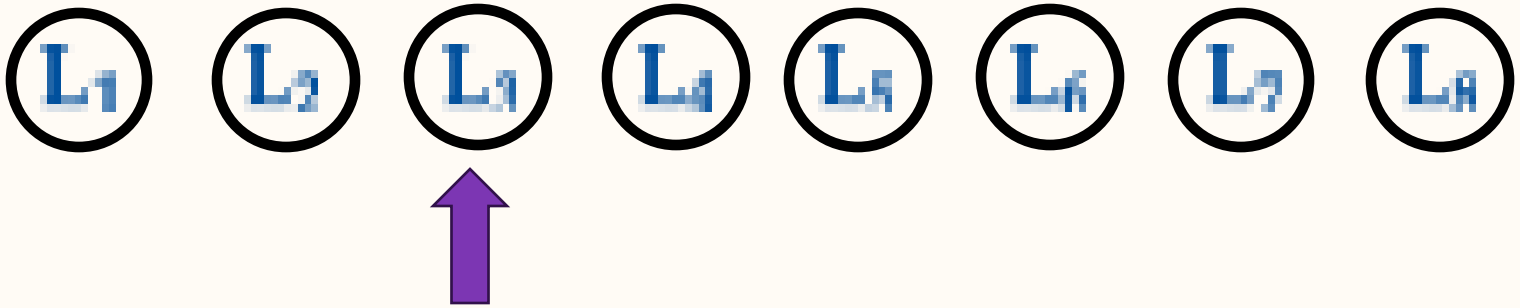
Natural language elimination strategy



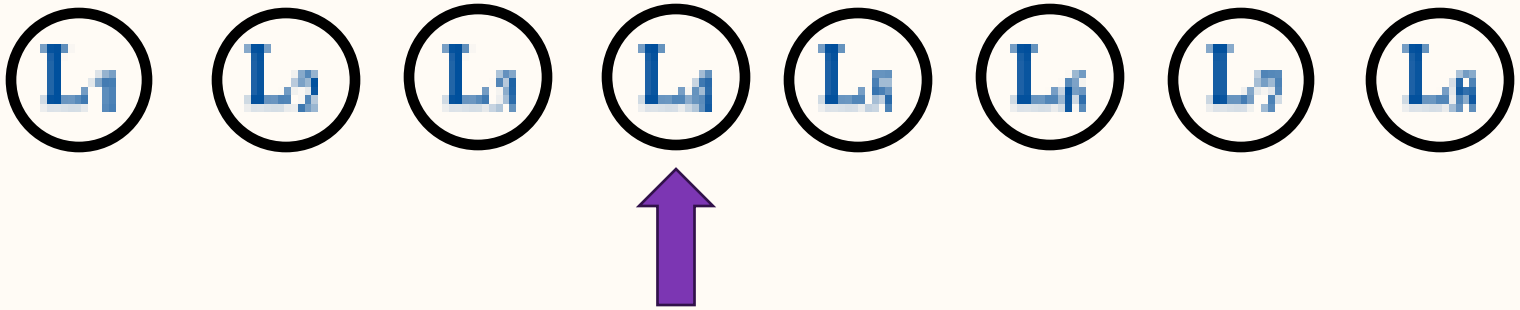
Natural language elimination strategy



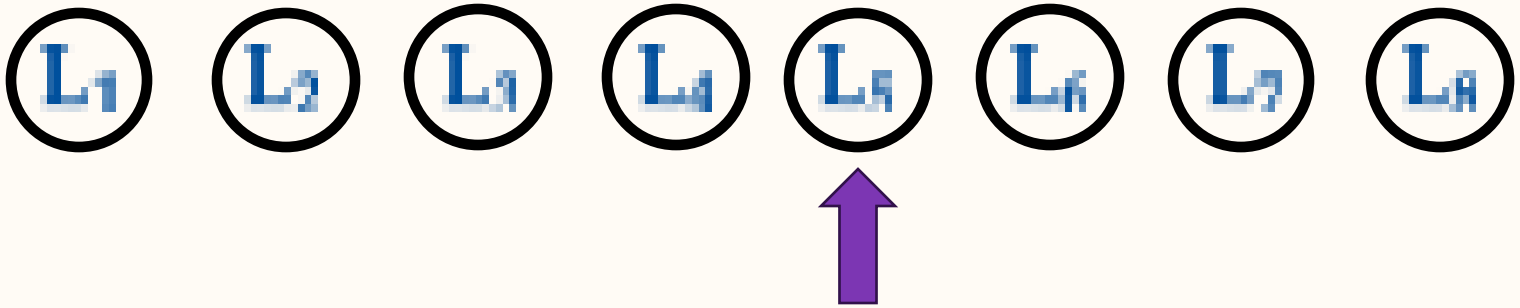
Natural language elimination strategy



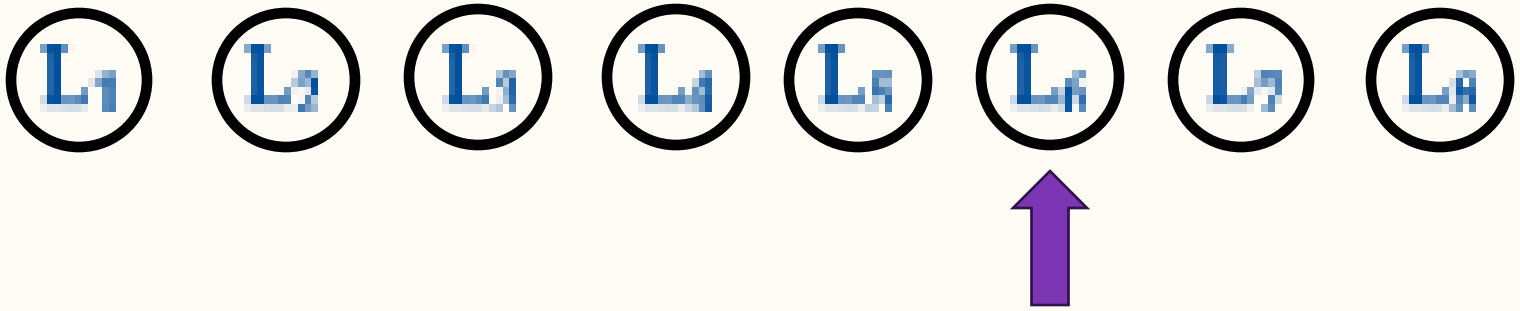
Natural language elimination strategy



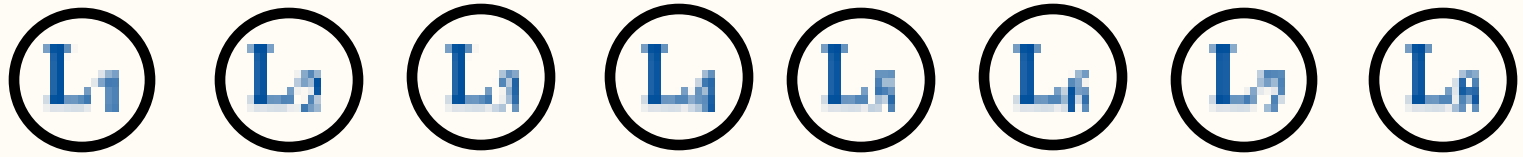
Natural language elimination strategy



Natural language elimination strategy



Natural language elimination strategy



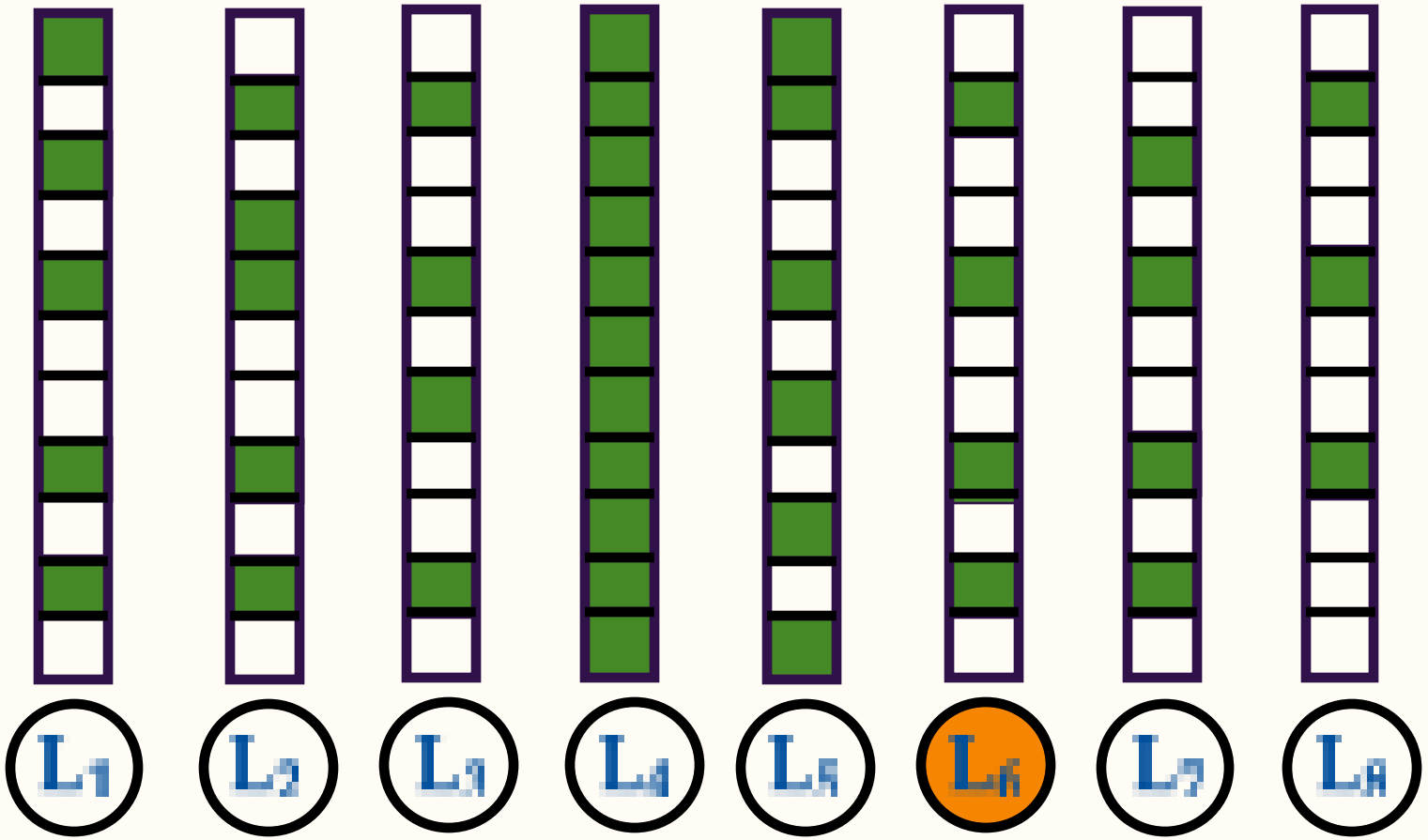
(wrong) Proof :

If $L_i \neq K$ enumeration of K will eventually reveal this

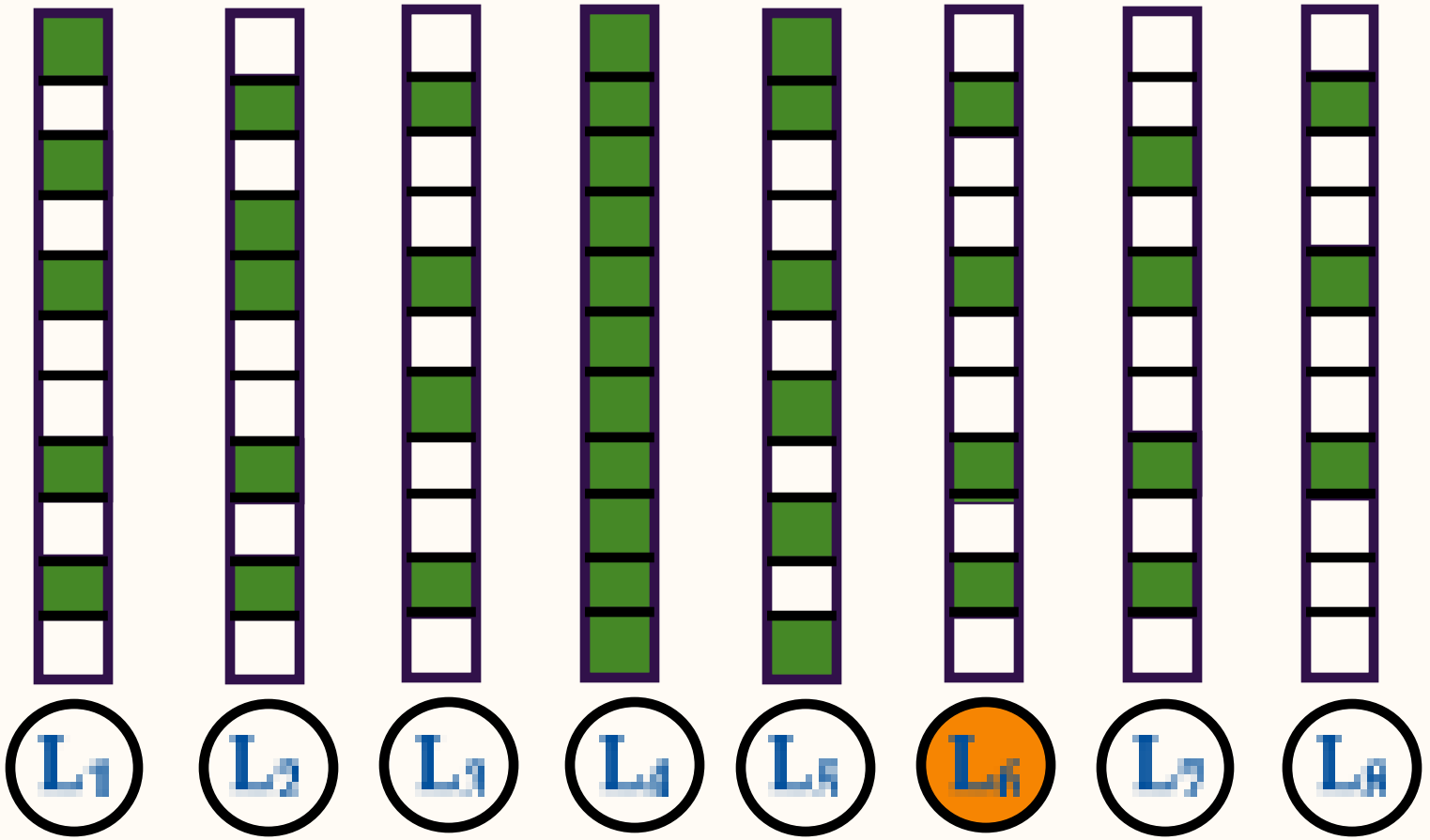
Since $K = L_z$ for some z, we will eventually get to $K = L_z$

We will never move beyond K

Natural language elimination strategy

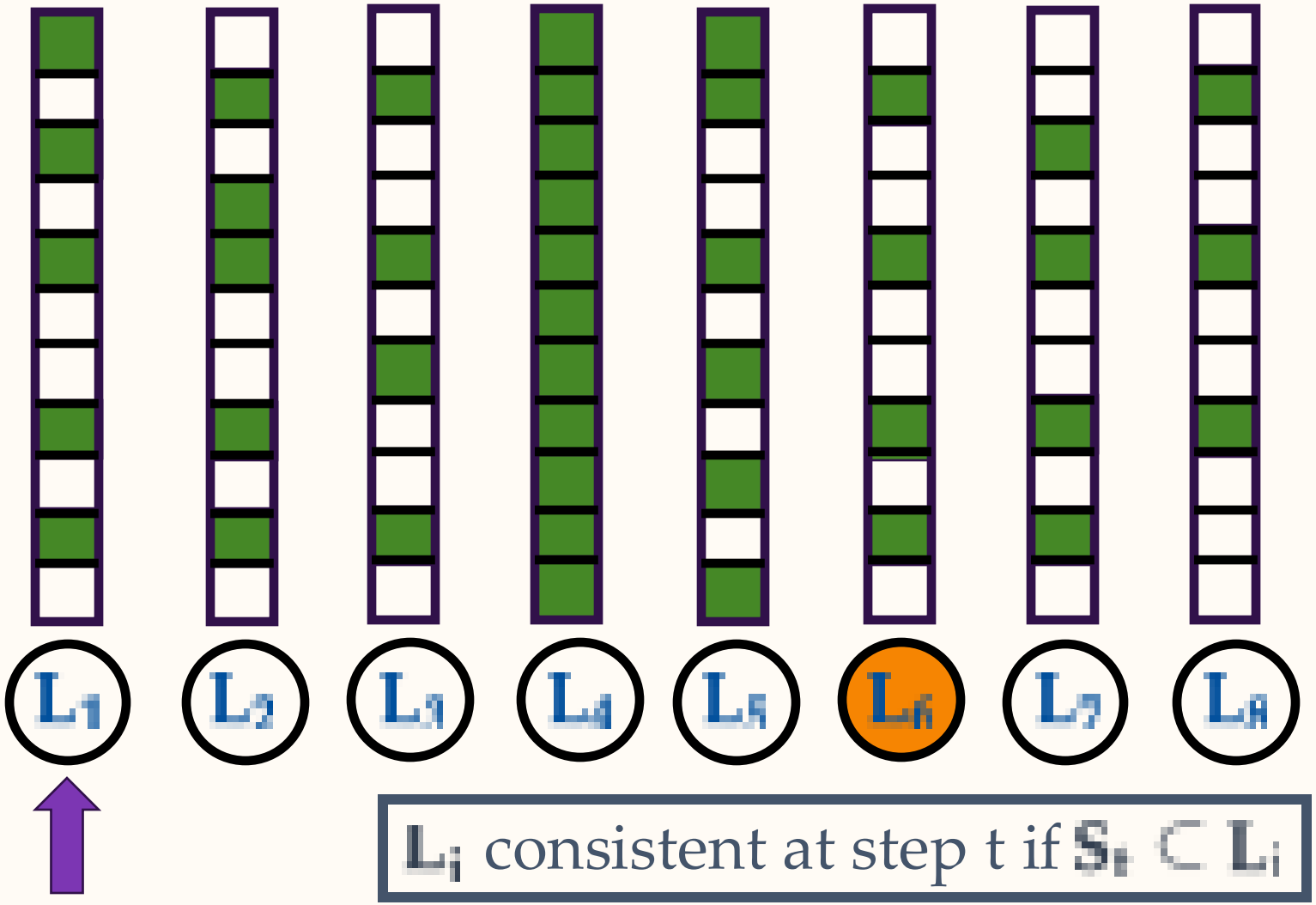


Natural language elimination strategy

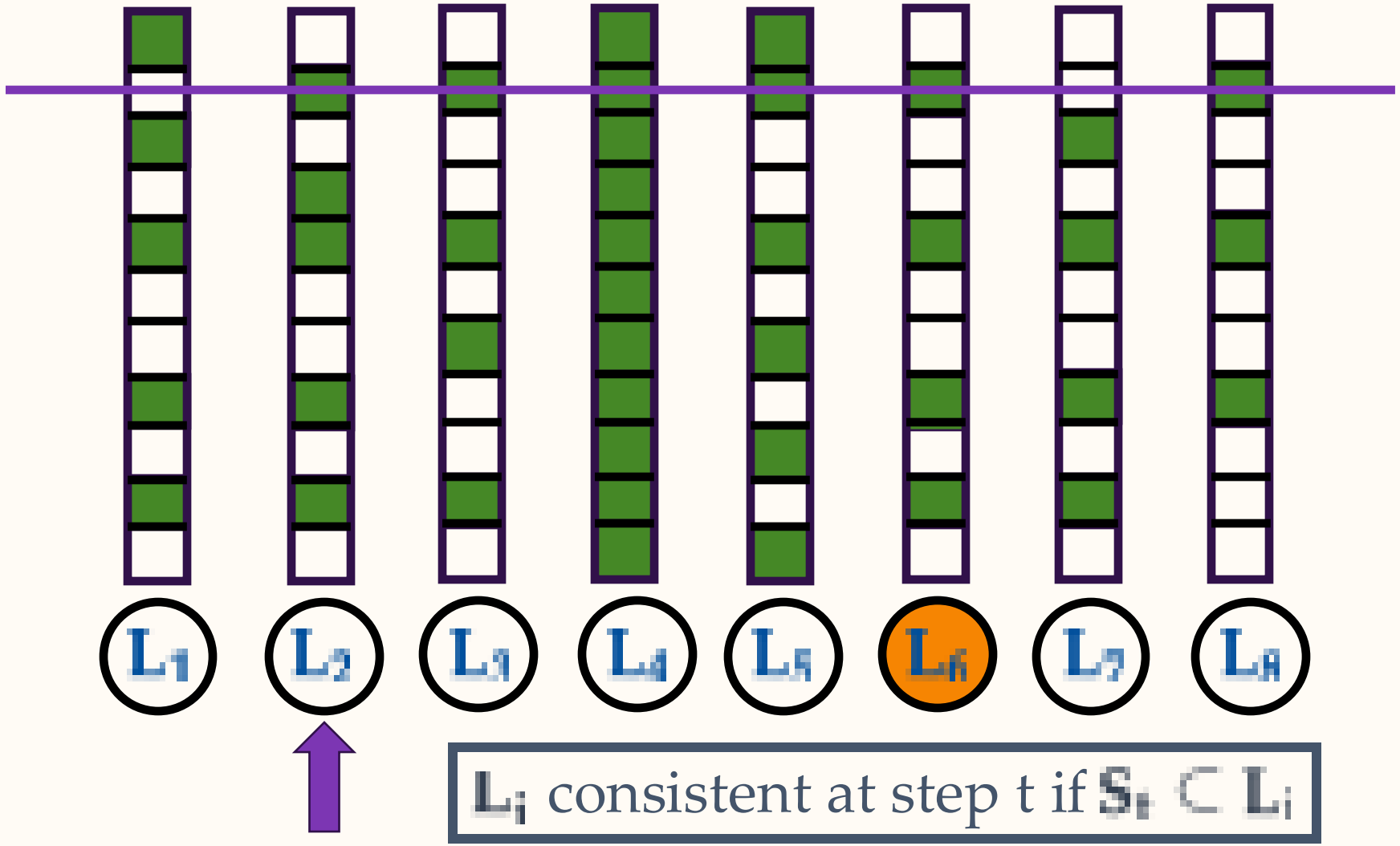


L_i consistent at step t if $S_t \subset L_i$

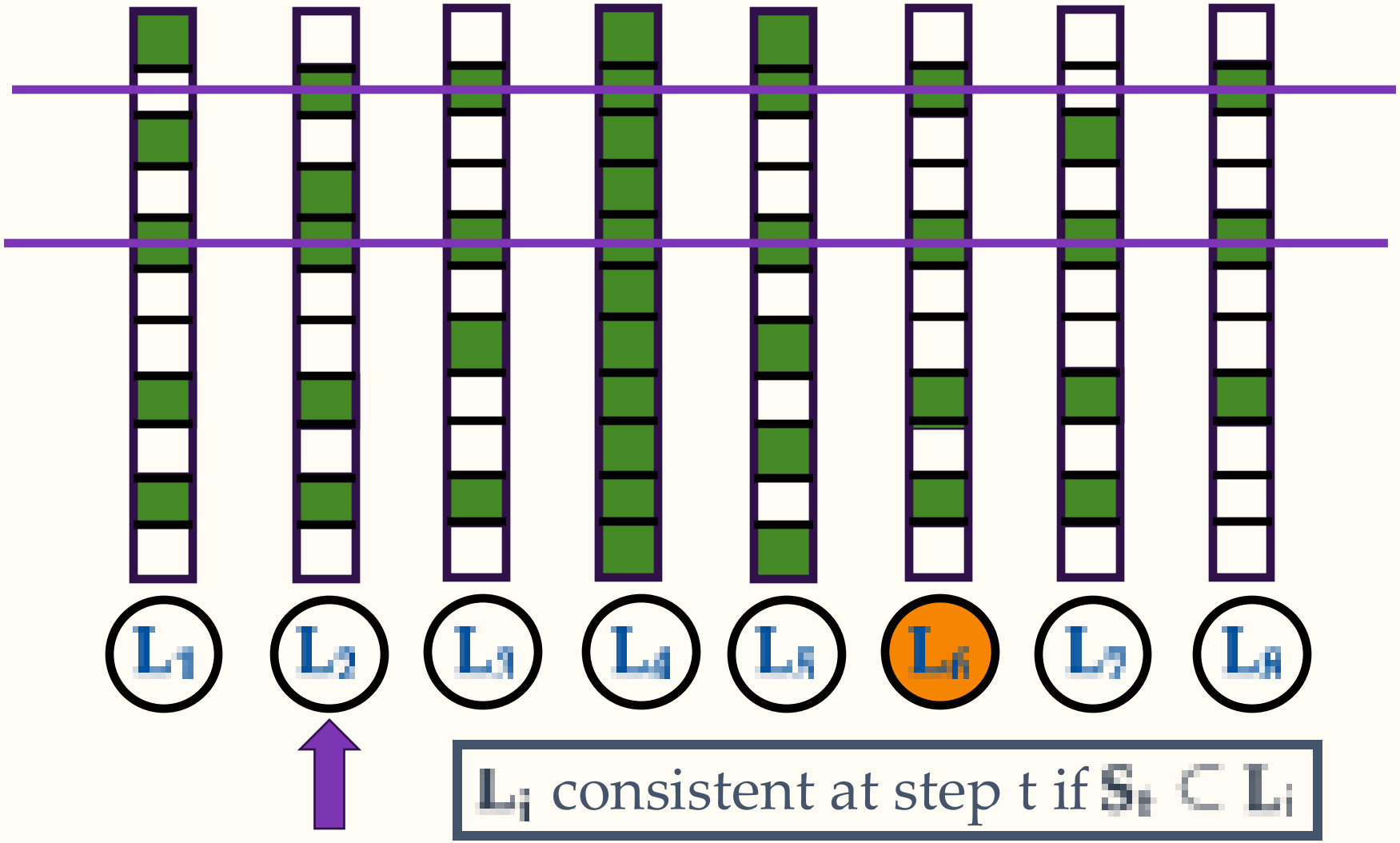
Natural language elimination strategy



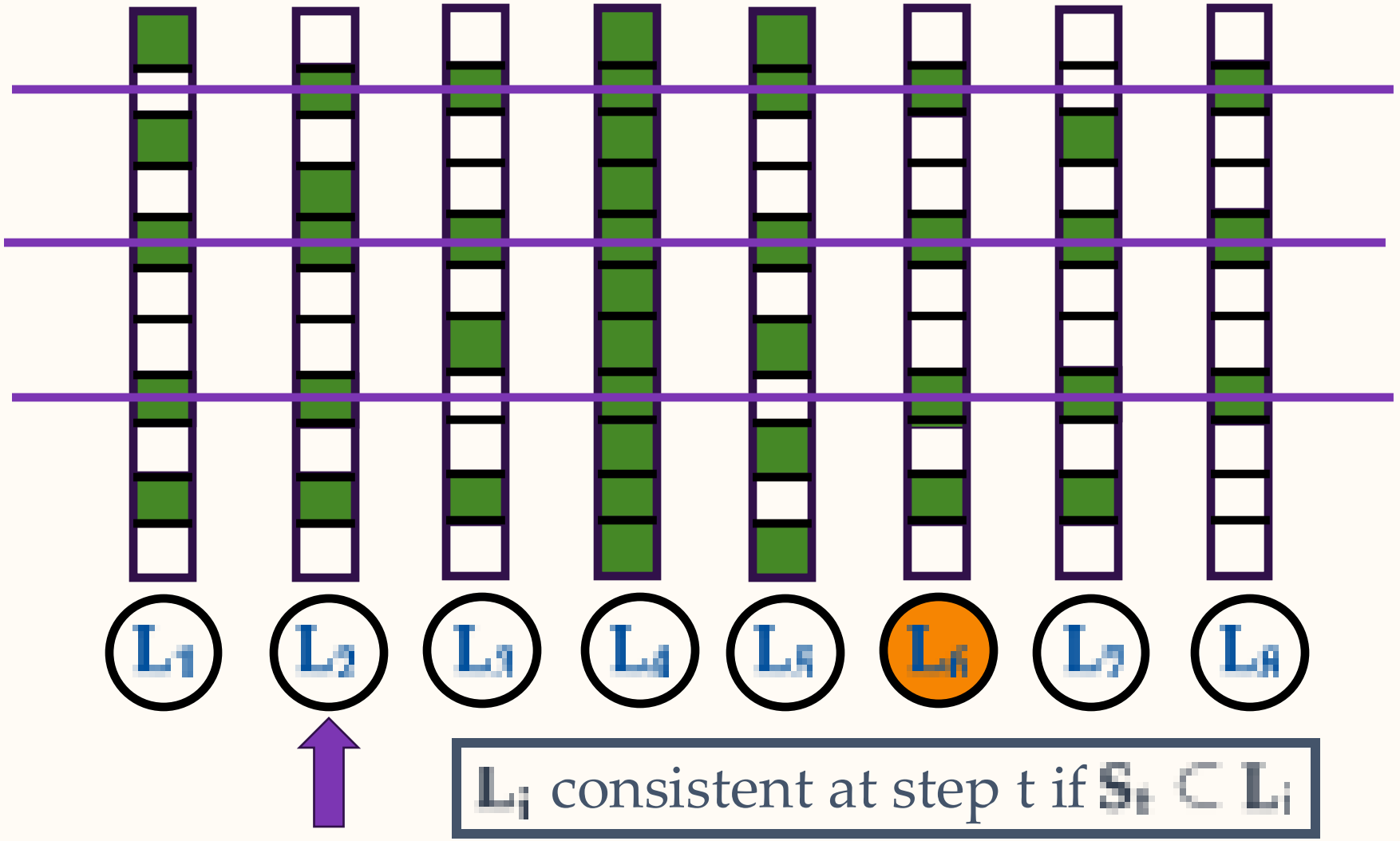
Natural language elimination strategy



Natural language elimination strategy



Natural language elimination strategy



Minimal Consistent Languages?

Consistent language L_i such that there is no consistent L_j with $L_j \subsetneq L_i$

Minimal Consistent Languages?

Consistent language L_i such that there is no consistent L_j with $L_j \subsetneq L_i$

Problem: There may not be a minimal consistent language!

Critical Languages

Definition: L_n is critical at step t if
1. L_n is consistent at step t
2. If L_i is consistent, $i < n$, then $L_n \subseteq L_i$

Critical Languages

Definition: L_n is critical at step t if
1. L_n is consistent at step t
2. If L_i is consistent, $i < n$, then $L_n \subseteq L_i$

Key facts:

1. critical language exists: lowest-indexed consistent language

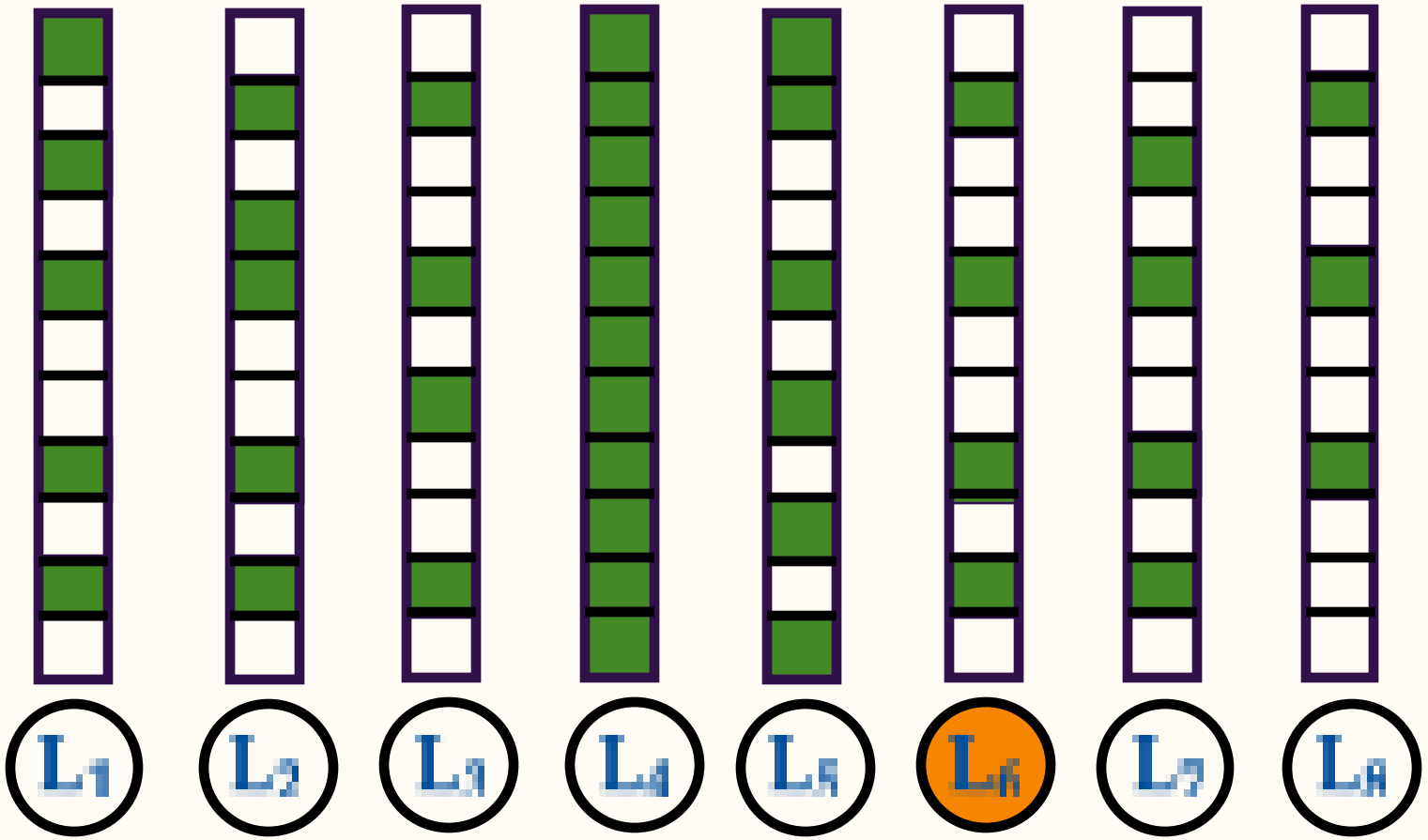
Critical Languages

Definition: L_n is critical at step t if
1. L_n is consistent at step t
2. If L_i is consistent, $i < n$, then $L_n \subseteq L_i$

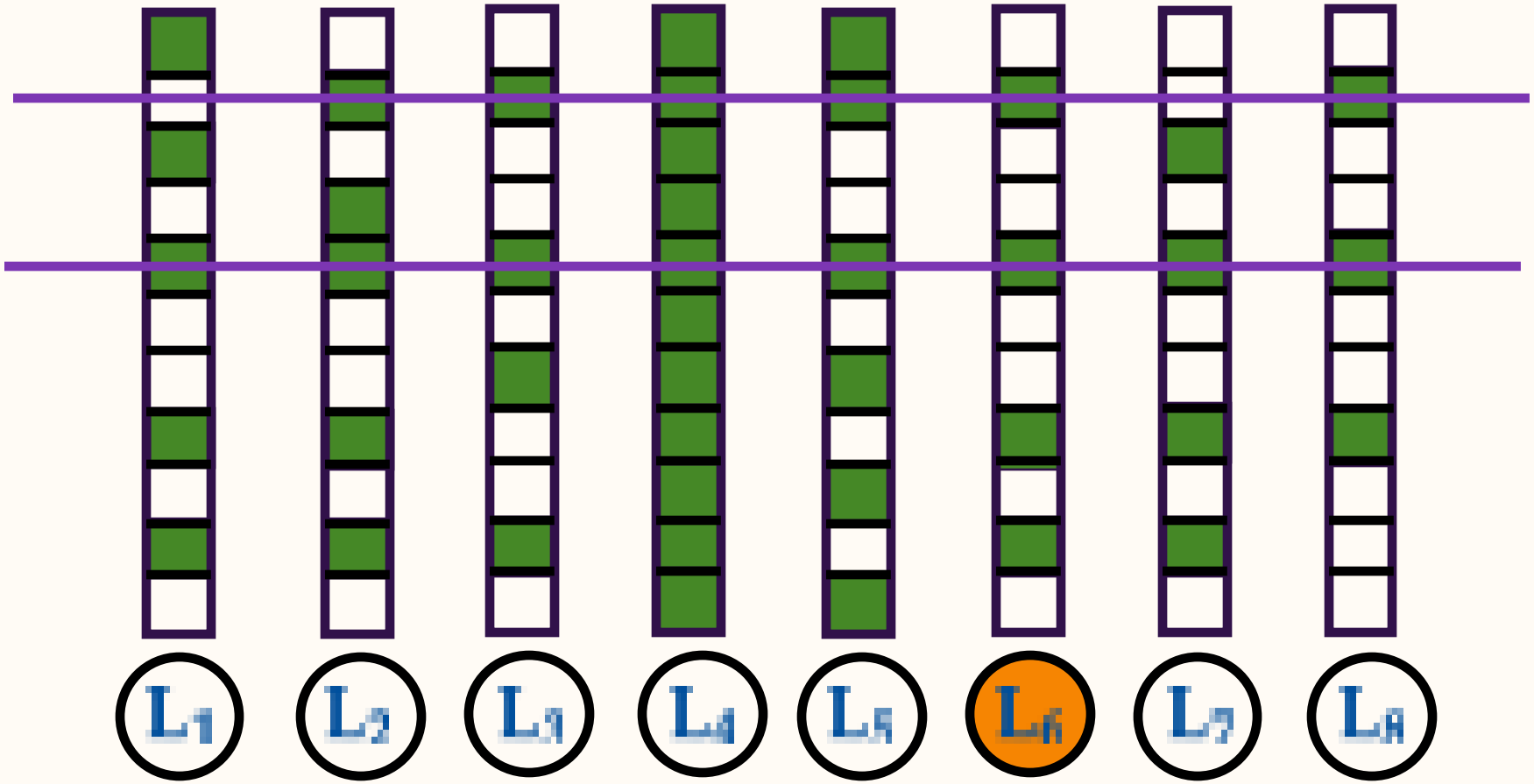
Key facts:

1. critical language exists: lowest-indexed consistent language
2. Target language will eventually become critical

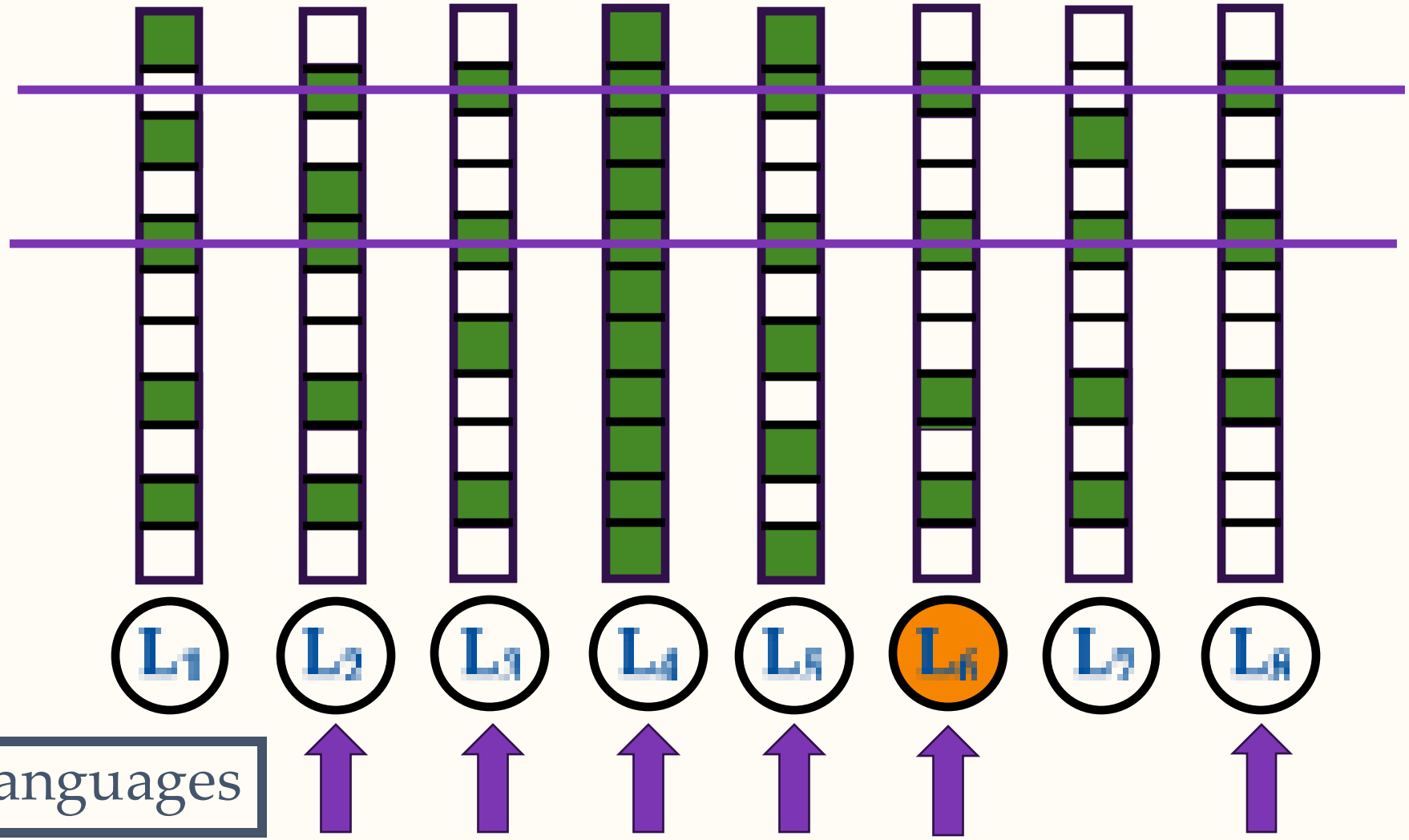
Target language will become critical



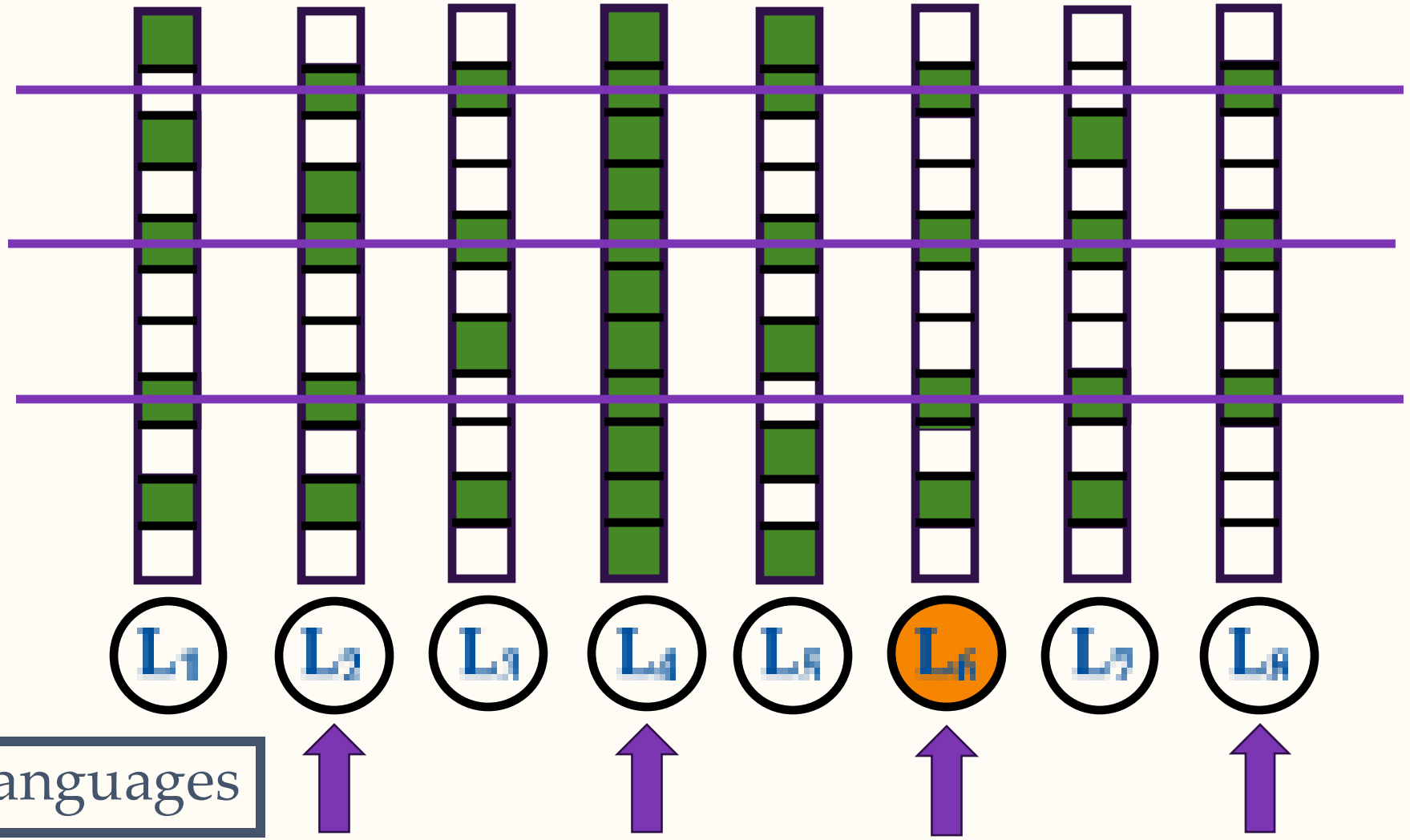
Target language will become critical



Target language will become critical



Target language will become critical



Algorithm for Generation in the Limit

At step t, only consider $L_1, \ldots, L_t$

L_{n_t} : critical language with highest index $n_t < t$

Generate a string from $L_{n_t} \setminus S_t$

Algorithm for Generation in the Limit

At step t, only consider $L_1, \ldots, L_t$

L_{n_t} : critical language with highest index $n_t < t$

Generate a string from $L_{n_t} \setminus S_t$

Proof sketch:

For large enough t, target language L_z is critical and in $L_1, \ldots, L_t$

Algorithm for Generation in the Limit

At step t, only consider $L_1, \ldots, L_t$

L_{n_t} : critical language with highest index $n_t < t$

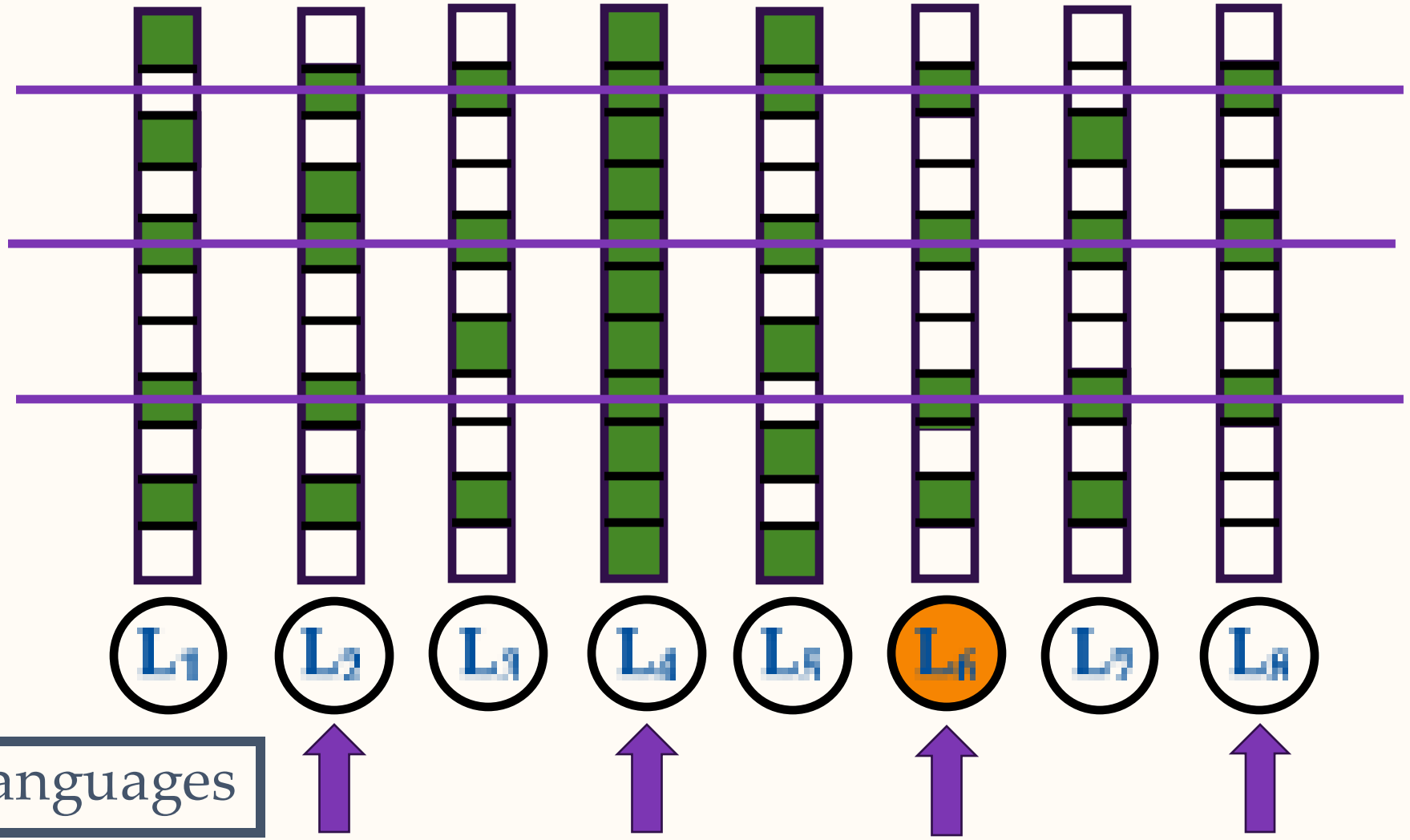
Generate a string from $L_{n_t} \setminus S_t$

Proof sketch:

For large enough t, target language L_z is critical and in $L_1, \ldots, L_t$

$L_{n_t} \subseteq L_z$, so any string from $L_{n_t} \setminus S_t$ also belongs to $L_z \setminus S_t$

Generation in the Limit correctness



Language Generation: Tradeoffs & Extensions

Mode-collapse in [Kleinberg, Mullainathan 2024] algorithm.
validity vs breadth tradeoff: Anay and Grigoris

Stronger requirements (uniform/non-uniform generation): Chirag

Diversity constraints and noisy data: Charlotte