Tutorial on Language Generation

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

Validity–Breadth Trade-Off (Part I)

This Talk:

- Why Breadth?
- Definitions of Breadth + Abstractions
- Results
- Step Back
- Future Directions

Proofs in Part II by Grigoris

Recap of Language Generation in the Limit

Language Generation

Kleinberg, Mullainathan, 2024

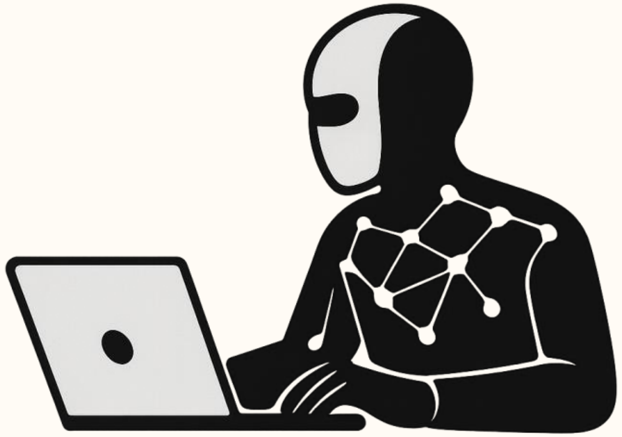
string of K →

← guess new string from K

string of K →

← guess new string from K

…



Success: guess correct (i.e., valid and unseen) for every $t > t^*$
(We say that algorithm has generated K in the limit)

Language Generation

Kleinberg, Mullainathan, 2024

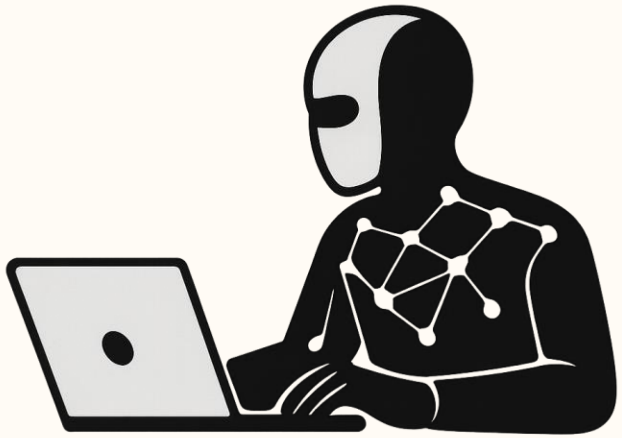
string of K →

← guess new string from K

string of K →

← guess new string from K

…



Algorithm never sees negative examples

Language Generation

Kleinberg, Mullainathan, 2024

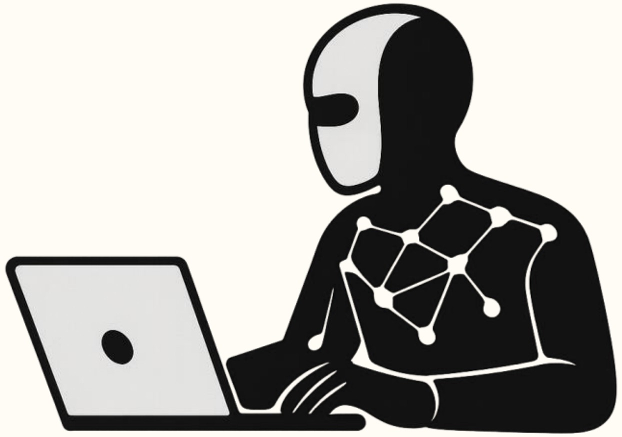
string of K

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

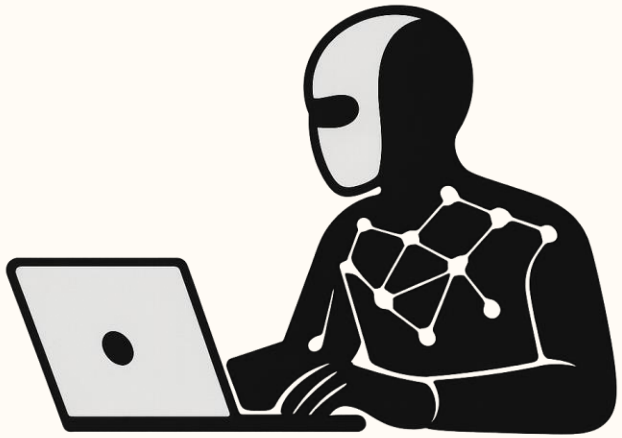
string of K →

← guess new string from K

string of K →

← guess new string from K

…



Algorithm never sees negative examples

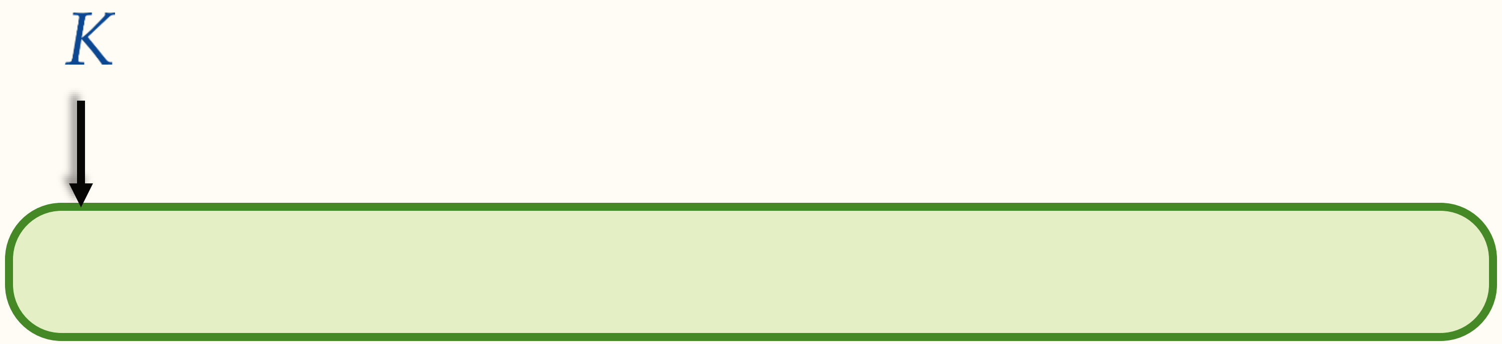
No feedback

Assume all languages infinite

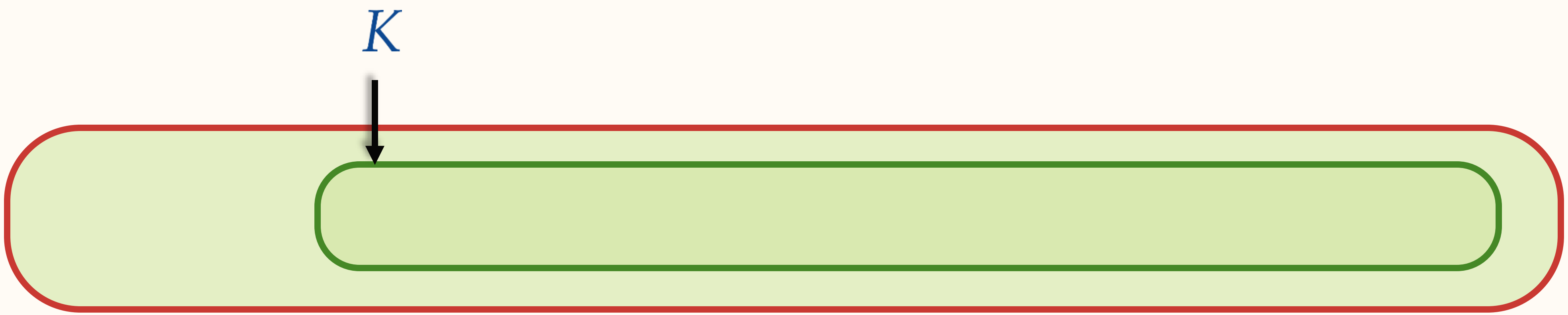
Language Generation in the Limit

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

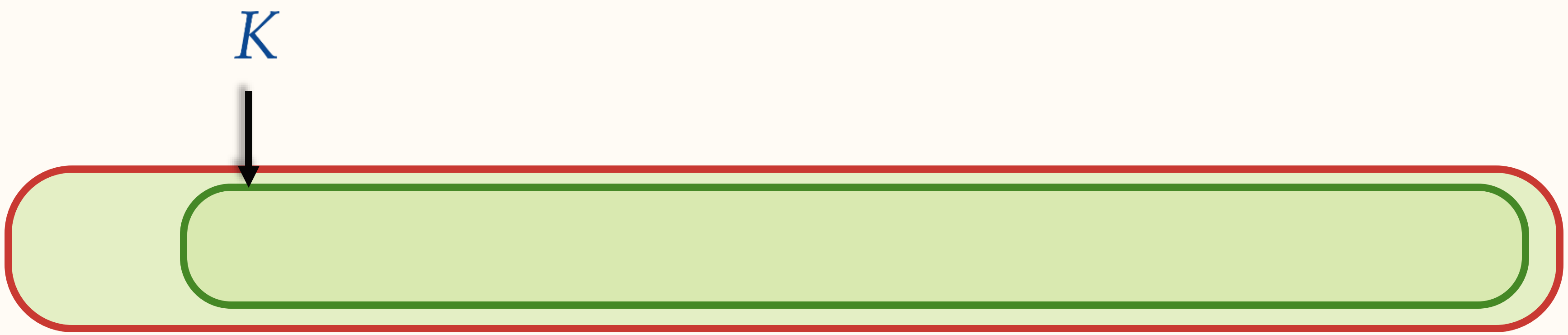
[KM’24]’s Generator Loses Breadth



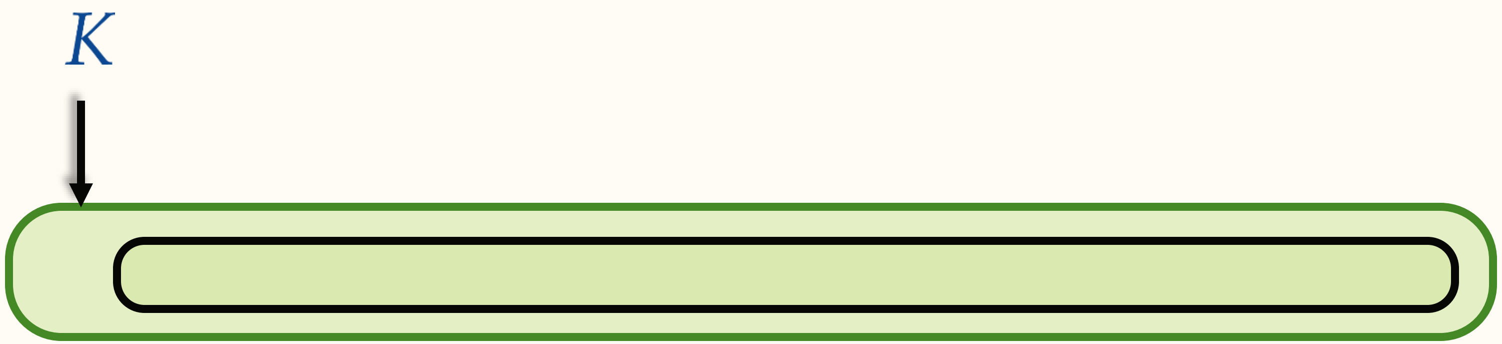
[KM'24]'s Generator Loses Breadth



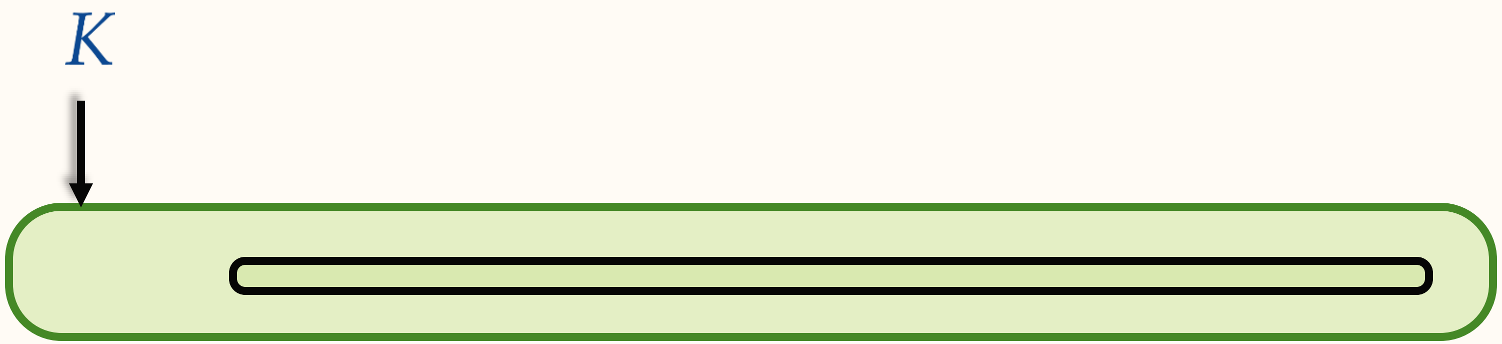
[KM'24]'s Generator Loses Breadth



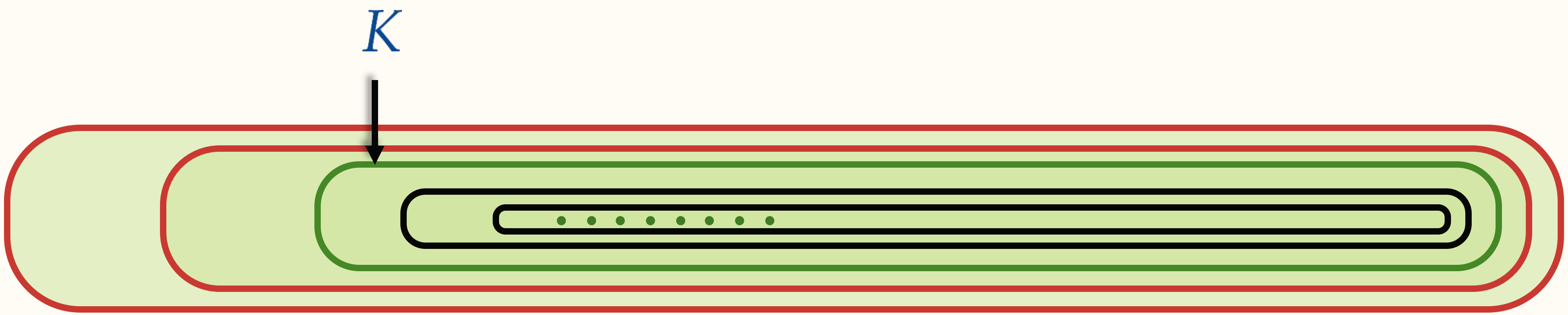
[KM'24]'s Generator Loses Breadth



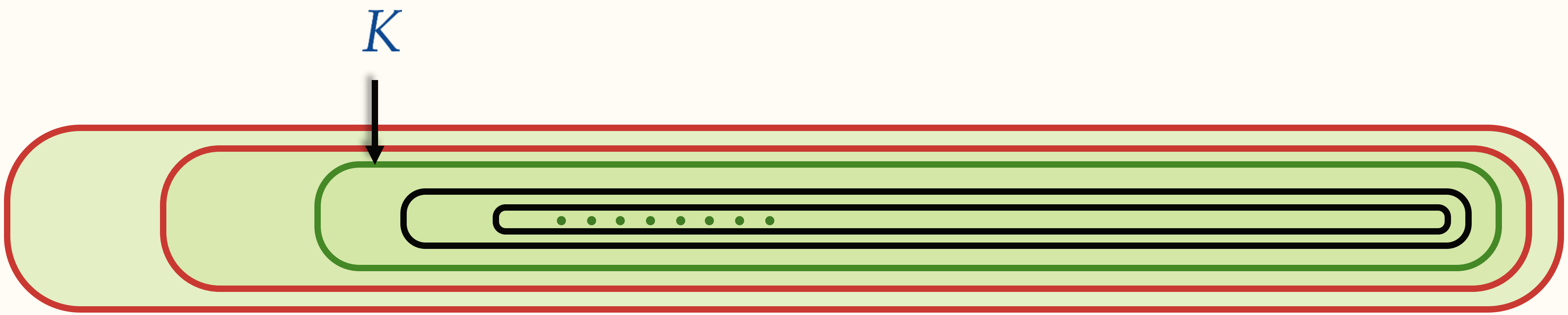
[KM'24]'s Generator Loses Breadth



[KM'24]'s Generator Loses Breadth

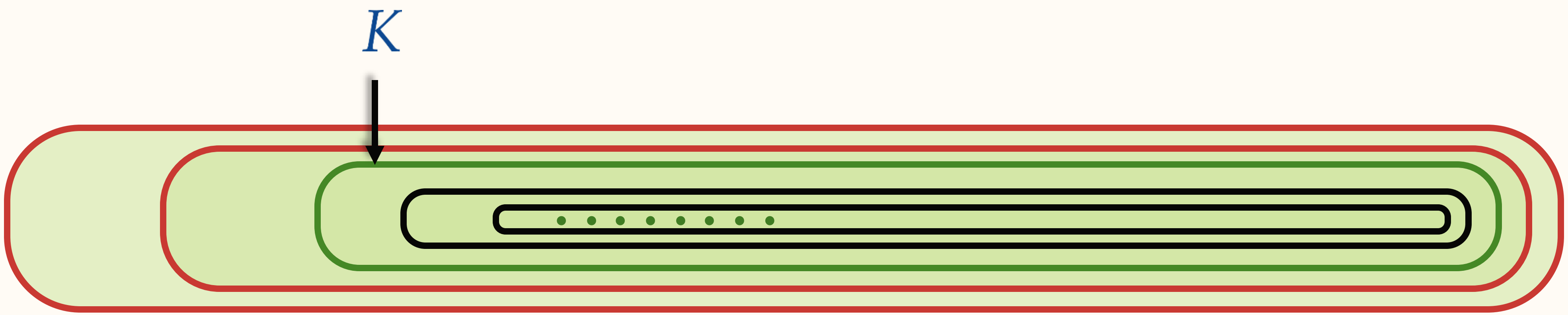


[KM'24]'s Generator Loses Breadth



One question asked by [KM24]: Can a generator avoid hallucinations while maintaining some notion of "breadth"

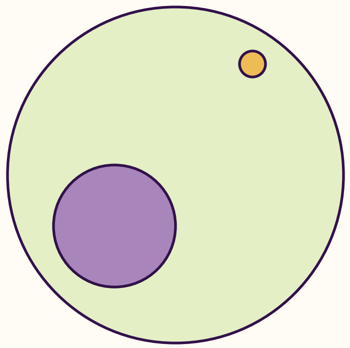
[KM'24]'s Generator Loses Breadth



One question asked by [KM24]: Can a generator avoid hallucinations while maintaining some notion of "breadth"

Or what is *the limit in generation in the limit*

What is Breadth?

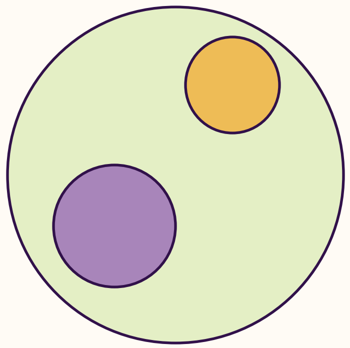


Generator's support G

Training data S

Target language K

What is Breadth?

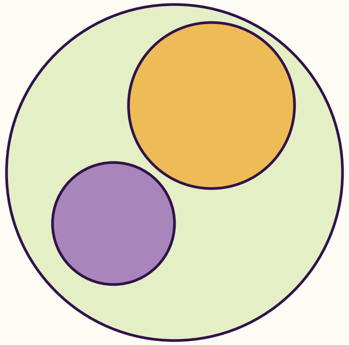


Generator's support G

Training data S

Target language K

What is Breadth?

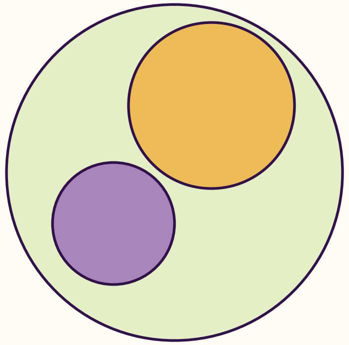


Generator's support G

Training data S

Target language K

What is Breadth?



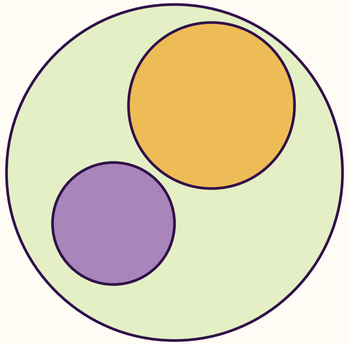
Generator's support G

Training data S

Target language K

Why is this relevant?

What is Breadth?



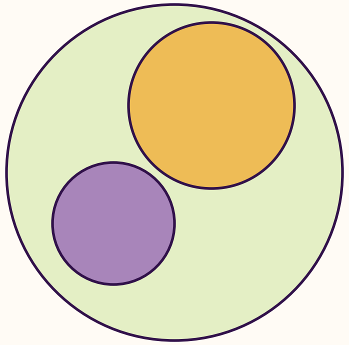
Generator's support G

Training data S

Target language K

Why is this relevant? Captures how much the *"knows"*

What is Breadth?



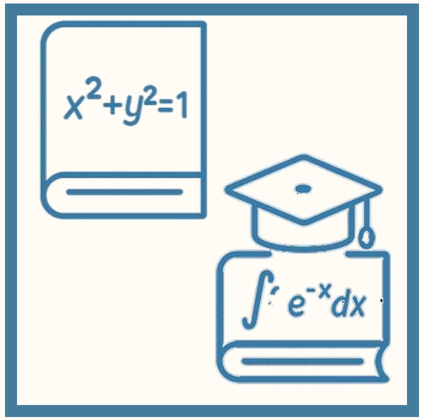
Generator's support G

Training data S

Target language K

Why is this relevant? Captures how much the *“knows”*

“panda surfing in a watermelon juice with sunglasses while singing.”



Solve new math problems

Some Definitions of Breadth

For generator G, let $G(S) \subseteq \mathcal{X}$ be the output-set of G trained on S

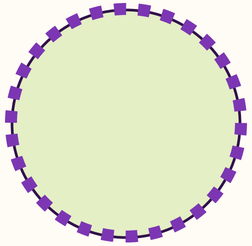
Some Definitions of Breadth

For generator G, let $G(S) \subseteq \mathcal{X}$ be the output-set of G trained on S

Set Theoretic Definitions

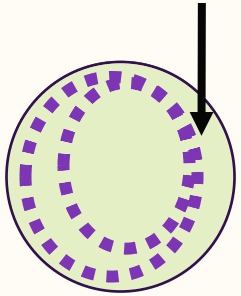
Exact Breadth

$G(S) = K \setminus S$



Approximate Breadth

$|(K \setminus S) \setminus G(S)| < \infty$



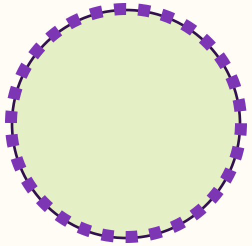
Some Definitions of Breadth

For generator G, let $G(S) \subseteq \mathcal{X}$ be the output-set of G trained on S

Set Theoretic Definitions

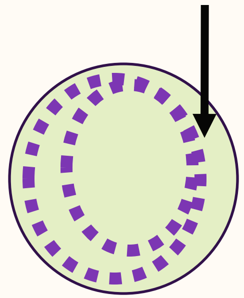
Exact Breadth

$G(S) = K \setminus S$



Approximate Breadth

$|(K \setminus S) \setminus G(S)| < \infty$



Consider $K = \mathbb{N}$, $\quad G(S) = \{i, i+1, \dots\} \quad$ and $\quad G(S) = \{2, 4, 6, \dots\}$

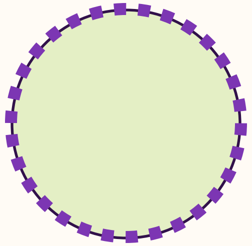
Some Definitions of Breadth

For generator G, let $G(S) \subseteq \mathcal{X}$ be the output-set of G trained on S

Set Theoretic Definitions

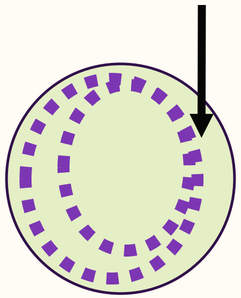
Exact Breadth

$G(S) = K \setminus S$



Approximate Breadth

$|(K \setminus S) \setminus G(S)| < \infty$



Consider $K = \mathbb{N}, \quad G(S) = \{i, i+1, \ldots\} \quad \text{and} \quad G(S) = \{2, 4, 6, \ldots\}$

Can also require/relax additional properties [CP'25] [KMV'24]

Abstractions of Breadth

Most lower bounds for breadth use diagonalization [CP'25] [KMV'24]

Core Question (in diagonalization-based proofs): How many languages can G achieve the property for *simultaneously*?

Abstractions of Breadth

Most lower bounds for breadth use diagonalization [CP'25] [KMV'24]

Core Question (in diagonalization-based proofs): How many languages can G achieve the property for *simultaneously*?

This view gives two abstractions:

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Abstractions of Breadth

Most lower bounds for breadth use diagonalization [CP'25] [KMV'24]

Core Question (in diagonalization-based proofs): How many languages can G achieve the property for *simultaneously*?

This view gives two abstractions:

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth:

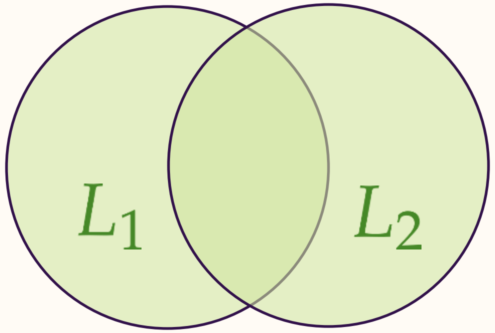
Suppose generator $G(S)$ achieves breadth for L_1, L_2 with $L_1 \neq L_2$

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth: 

Suppose generator $G(S)$ achieves breadth for L_1, L_2 with $L_1 \neq L_2$

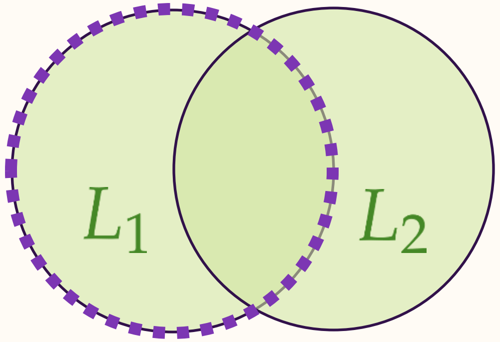


Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth:

Suppose generator $G(S)$ achieves breadth for L_1, L_2 with $L_1 \neq L_2$

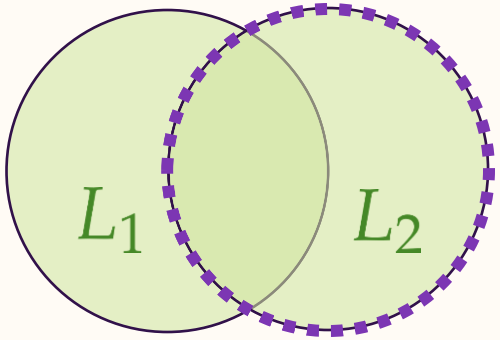


Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth:

Suppose generator $G(S)$ achieves breadth for L_1, L_2 with $L_1 \neq L_2$

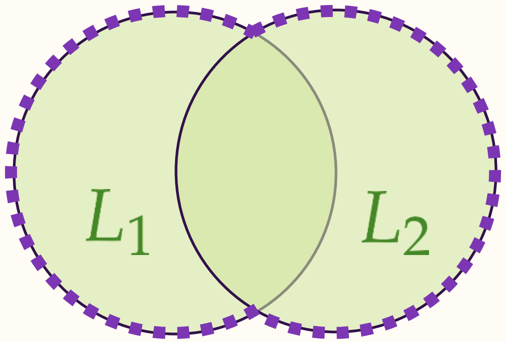


Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth:

Suppose generator $G(S)$ achieves breadth for L_1, L_2 with $L_1 \neq L_2$



Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✅

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth

Approximate-Breadth:

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✅

Approximate-Breadth:

Consider $L_1 \subsetneq L_2$ with $|L_2 \setminus L_1| < \infty$

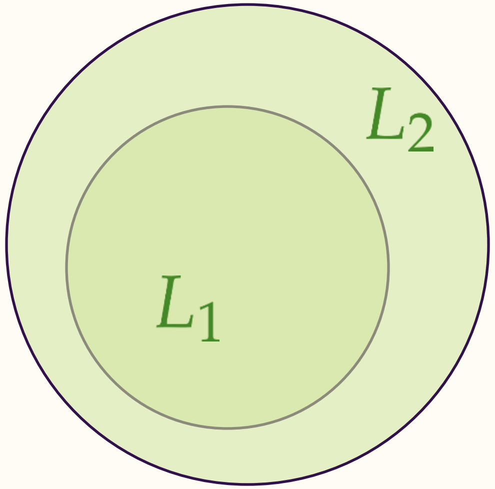
Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✓

Approximate-Breadth:

Consider $L_1 \subsetneq L_2$ with $|L_2 \setminus L_1| < \infty$



Which Notions of Breadth have Uniqueness?

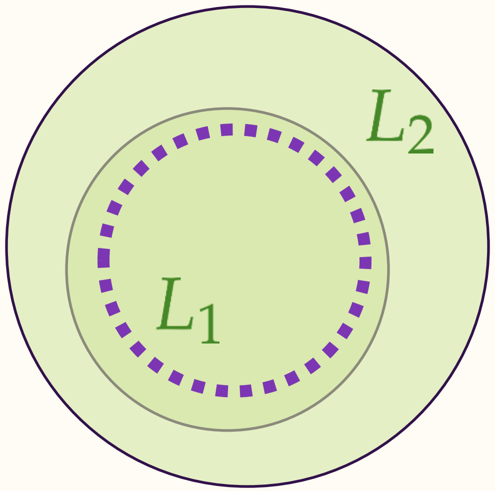
Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✓

Approximate-Breadth:

Consider $L_1 \subsetneq L_2$ with $|L_2 \setminus L_1| < \infty$

Let $G(S)$ have approximate breadth for L_2



Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

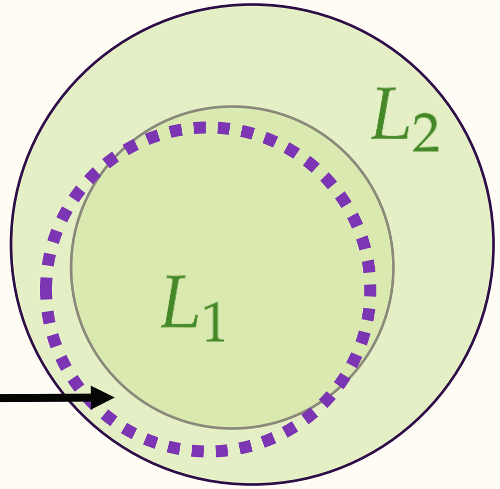
Exact Breadth

Approximate-Breadth:

Consider $L_1 \subsetneq L_2$ with $|L_2 \setminus L_1| < \infty$

Let $G(S)$ have approximate breadth for L_2

Can generate things not in L_1!



Which Notions of Breadth have Uniqueness?

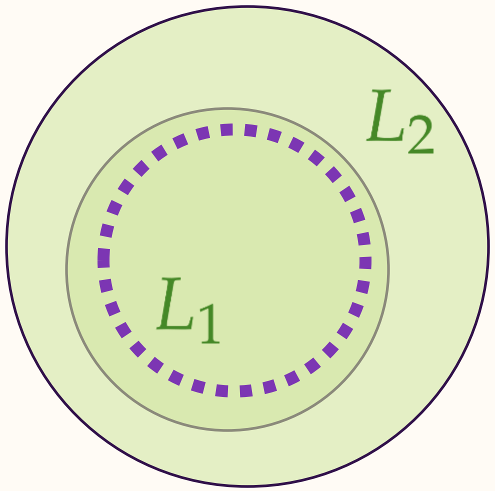
Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✓

Approximate-Breadth:

Consider $L_1 \subsetneq L_2$ with $|L_2 \setminus L_1| < \infty$

Let $G(S)$ have approximate breadth for $\cancel{L_2}$ L_1



Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✓ Approximate Breadth ✗

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✅ Approximate Breadth ❌

Unambiguous generation has uniqueness:

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✓ Approximate Breadth ✗

Unambiguous generation has uniqueness:

Definition. G generates unambiguously from K w.r.t. $\mathcal{L}$ if G is *strictly better* generator for K than $L \neq K$ in $\mathcal{L}$ (w.r.t. any metric)

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✓ Approximate Breadth ✗

Unambiguous generation has uniqueness:

Definition. G generates unambiguously from K w.r.t. $\mathcal{L}$ if G is *strictly better* generator for K than $L \neq K$ in $\mathcal{L}$ (w.r.t. <u>any metric</u>)

Which Notions of Breadth have Uniqueness?

Definition (Uniqueness). Property P has uniqueness, if generator G can achieve P for only one language at once.

Exact Breadth ✓ Approximate Breadth ✗ Unambiguous ✓

Unambiguous generation has uniqueness:

Definition. G generates unambiguously from K w.r.t. $\mathcal{L}$ if G is *strictly better* generator for K than $L \neq K$ in $\mathcal{L}$ (w.r.t. <u>any metric</u>)

Which Notions have Finite Non-Uniqueness?

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

Any property with uniqueness:

Which Notions have Finite Non-Uniqueness?

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

Any property with uniqueness:

Exact Breadth ✓ Approximate Breadth ? Unambiguous ✓

Which Notions have Finite Non-Uniqueness?

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

Any property with uniqueness:

Exact Breadth ✓ Approximate Breadth ? Unambiguous ✓

Approximate Breadth:

Which Notions have Finite Non-Uniqueness?

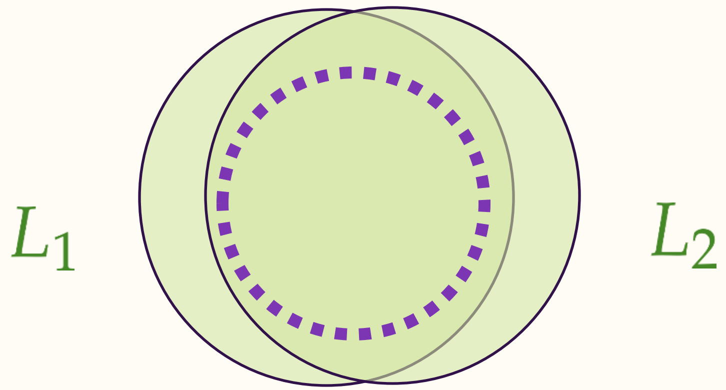
Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

Any property with uniqueness:

Exact Breadth ✓ Approximate Breadth ? Unambiguous ✓

Approximate Breadth:

Suppose $G(S)$ has approximate breadth for L_1 and L_2



Which Notions have Finite Non-Uniqueness?

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

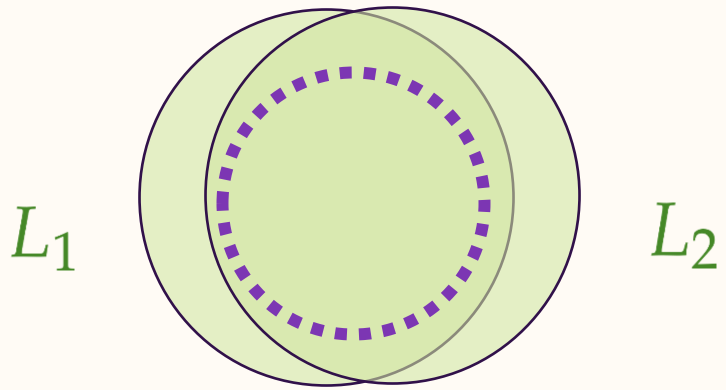
Any property with uniqueness:

Exact Breadth ✓ Approximate Breadth ? Unambiguous ✓

Approximate Breadth:

Suppose $G(S)$ has approximate breadth for L_1 and L_2

$$L_1 \triangle L_2 \subseteq (L_1 \setminus L_2) \cup (L_2 \setminus L_1)$$



Which Notions have Finite Non-Uniqueness?

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

Any property with uniqueness:

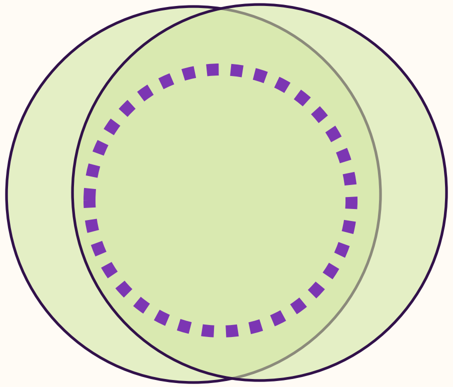
Exact Breadth   Approximate Breadth   Unambiguous

Approximate Breadth:

Suppose $G(S)$ has approximate breadth for L_1 and L_2

$$\begin{aligned} L_1 \triangle L_2 &\subseteq (L_1 \setminus L_2) \cup (L_2 \setminus L_1) \\ &\subseteq (L_1 \setminus G(S)) \cup (L_2 \setminus G(S)) \end{aligned}$$

because $G(S) \subseteq L_1, L_2$

L_1  L_2

Which Notions have Finite Non-Uniqueness?

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

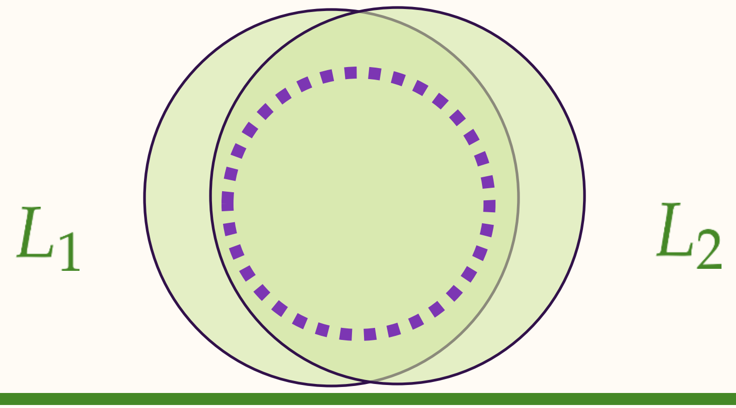
Any property with uniqueness:

Exact Breadth ✓ Approximate Breadth ✓ Unambiguous ✓

Approximate Breadth:

Suppose $G(S)$ has approximate breadth for L_1 and L_2

$$\begin{aligned} L_1 \triangle L_2 &\subseteq (L_1 \setminus L_2) \cup (L_2 \setminus L_1) \\ &\subseteq (L_1 \setminus G(S)) \cup (L_2 \setminus G(S)) \end{aligned}$$



because $G(S) \subseteq L_1, L_2$

Which Notions have Finite Non-Uniqueness?

Definition (Finite Non-Uniqueness). Property P has finite non-uniqueness, if G can achieve P for L_1 and L_2 only if $|L_1 \Delta L_2| < \infty$

Any property with uniqueness:

Exact Breadth ✓ Approximate Breadth ✓ Unambiguous ✓

Other examples: *Allow generators to have finite hallucinations*

Results with Breadth

Theorem [CP'25][KMV'24]. For any language collection $\mathcal{L}$:

Results with Breadth

Theorem [CP'25][KMV'24]. For any language collection $\mathcal{L}$:

- Generator $\mathcal{G}$ can achieve a property P with *uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is identifiable

Results with Breadth

Theorem [CP'25][KMV'24]. For any language collection $\mathcal{L}$:

- ▷ Generator $\mathcal{G}$ can achieve a property P with *uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is identifiable
- ▷ Generator $\mathcal{G}$ can achieve a property P with *finite non-uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is "almost" identifiable

Results with Breadth

Theorem [CP'25][KMV'24]. For any language collection $\mathcal{L}$:

- Generator $\mathcal{G}$ can achieve a property P with *uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is identifiable
- Generator $\mathcal{G}$ can achieve a property P with *finite non-uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is "almost" identifiable

Will be formalized in the next talk!

Results with Breadth

Theorem [CP'25][KMV'24]. For any language collection $\mathcal{L}$:

- ▷ Generator $\mathcal{G}$ can achieve a property P with *uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is identifiable
- ▷ Generator $\mathcal{G}$ can achieve a property P with *finite non-uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is "almost" identifiable

Will be formalized in the next talk!

Proof strategy? *Diagonalization via characterization of identification*

Results with Breadth

Theorem [CP'25][KMV'24]. For any language collection $\mathcal{L}$:

- ▷ Generator $\mathcal{G}$ can achieve a property P with *uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is identifiable
- ▷ Generator $\mathcal{G}$ can achieve a property P with *finite non-uniqueness* for $\mathcal{L}$ only if $\mathcal{L}$ is "almost" identifiable

Will be formalized in the next talk!

Proof strategy? *Diagonalization via characterization of identification*

[KMV'25]: (a) extends to statistical model, (b) extends conditionally

Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

Some “Fine-Grained” Definitions of Breadth

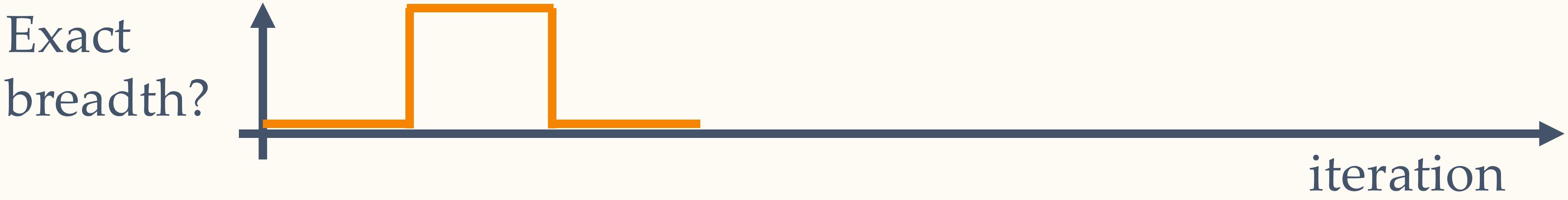
Implication: For non-identifiable collections exact breadth is violated infinitely many times

Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

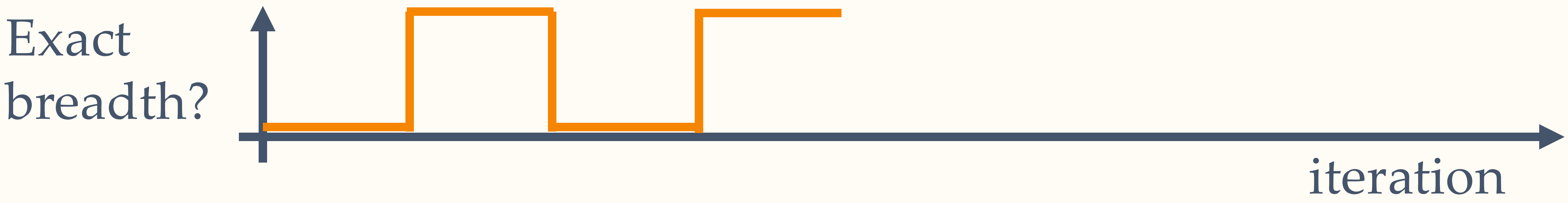
Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times



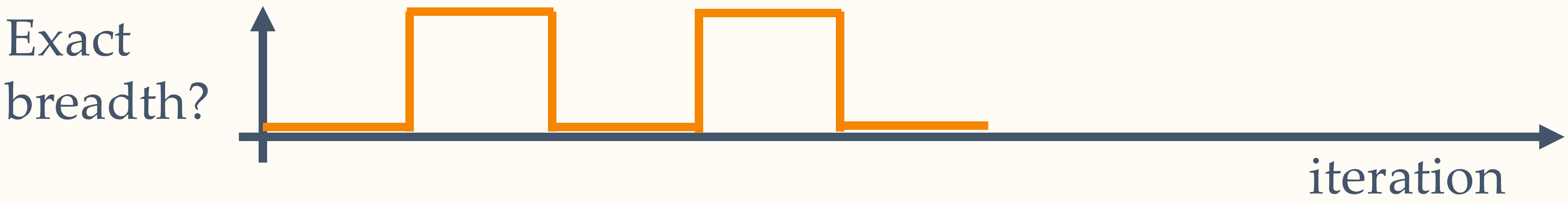
Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times



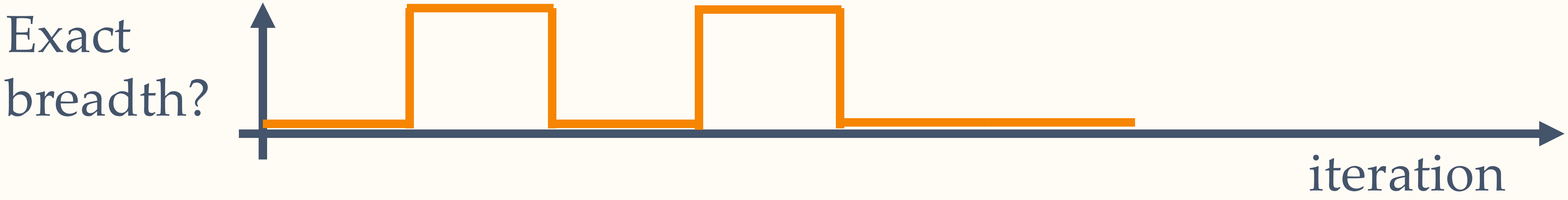
Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times



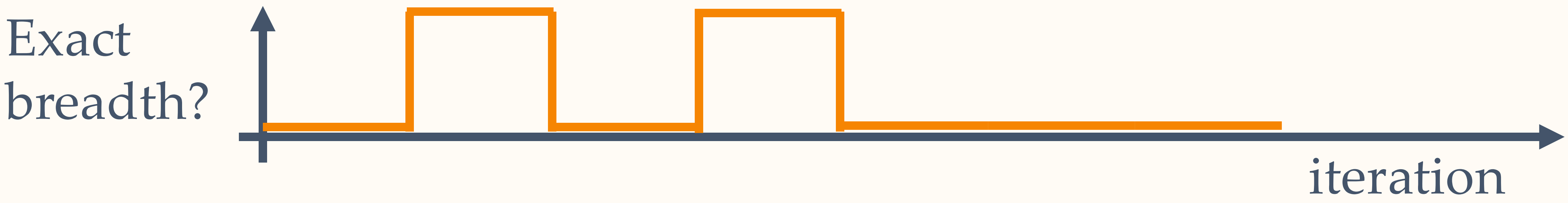
Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times



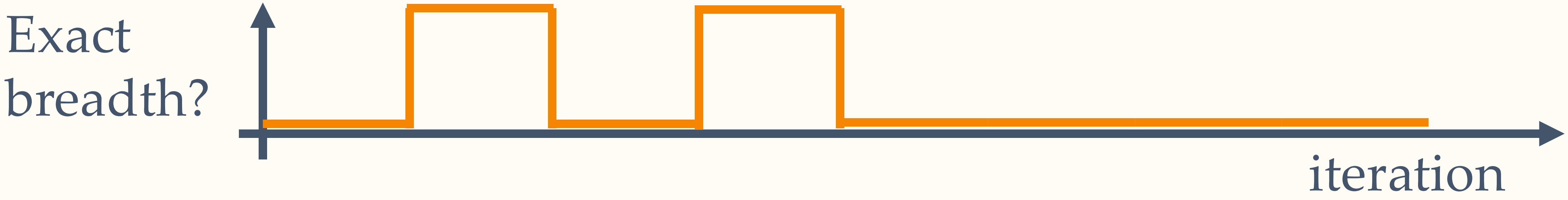
Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times



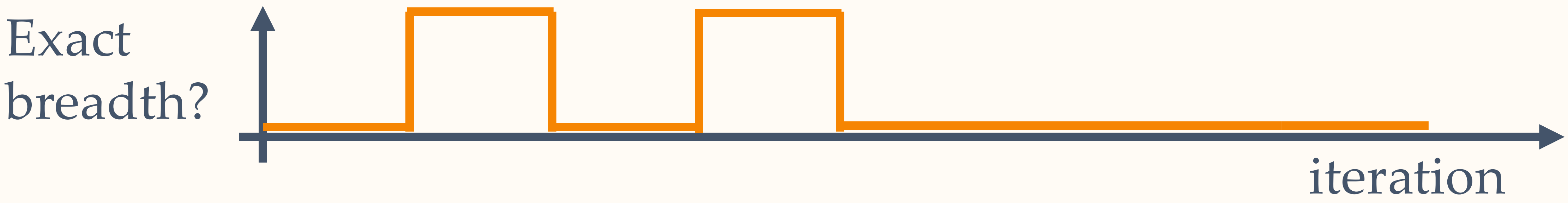
Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times



Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times



Question: Is it possible to achieve exact breadth infinitely many times?

Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

Theorem. [KW’25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$

Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

Theorem. [KW'25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$

Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

Theorem. [KW’25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$

Some "Fine-Grained" Definitions of Breadth

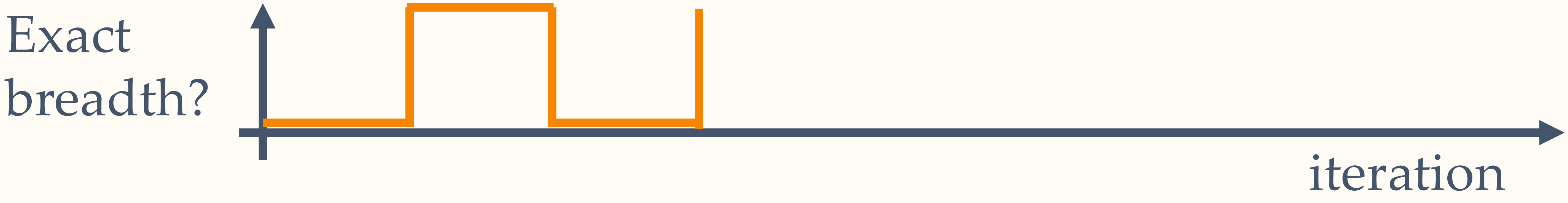
Implication: For non-identifiable collections exact breadth is violated infinitely many times

Theorem. [KW'25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$

Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

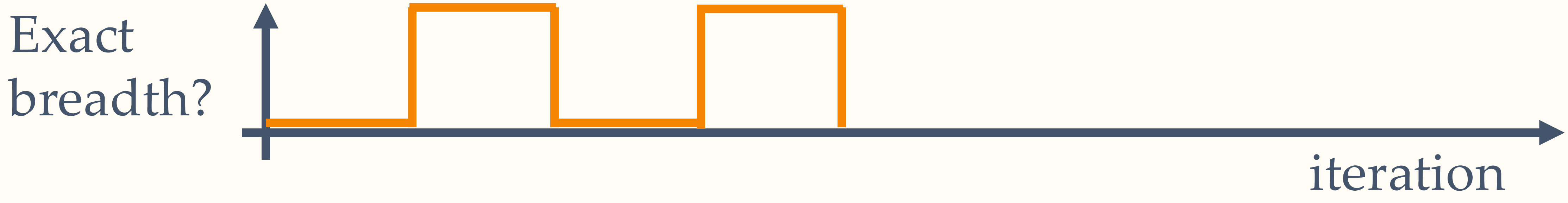
Theorem. [KW'25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$



Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

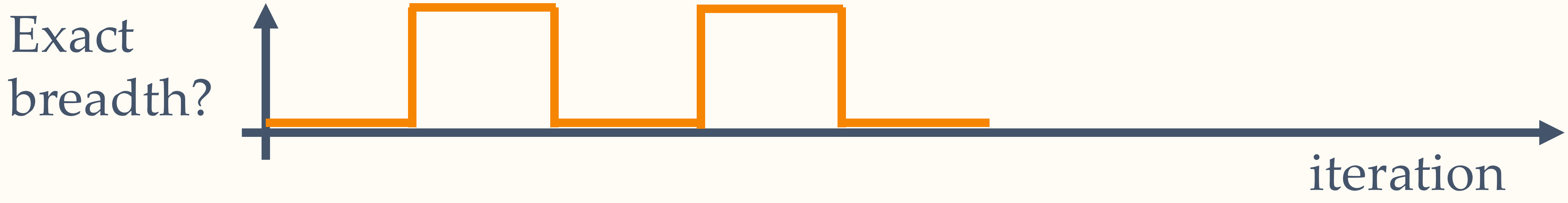
Theorem. [KW’25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$



Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

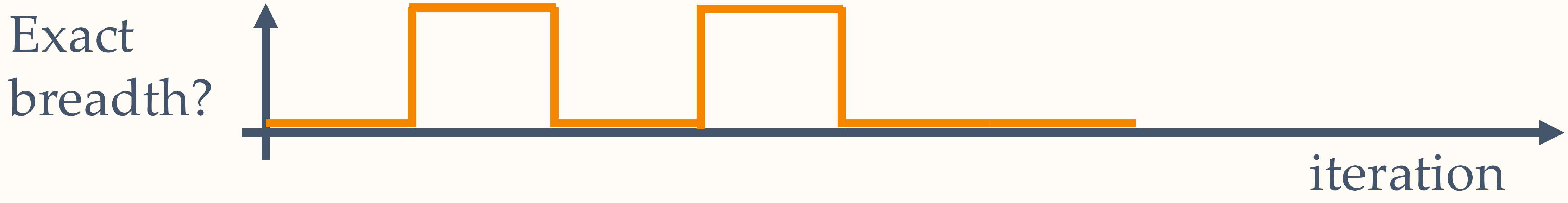
Theorem. [KW’25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$



Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

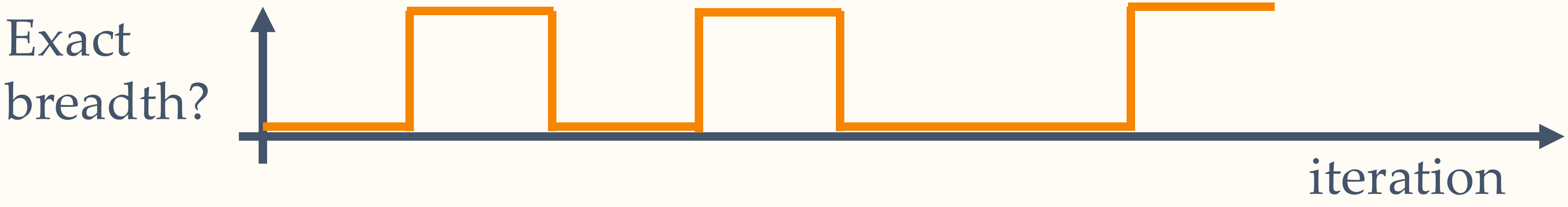
Theorem. [KW'25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$



Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

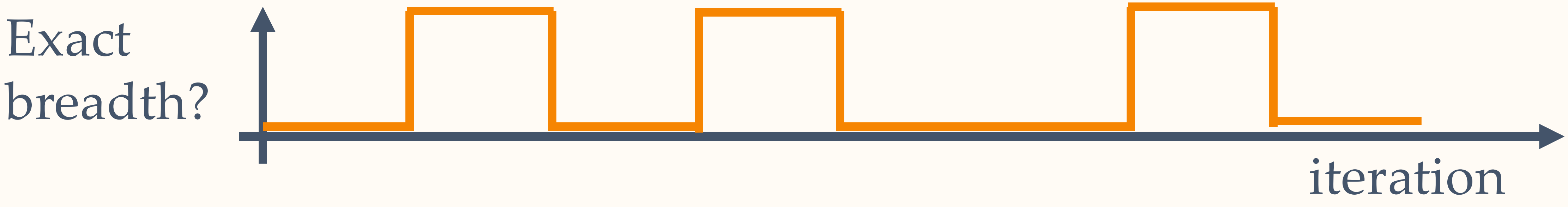
Theorem. [KW'25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$



Some "Fine-Grained" Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

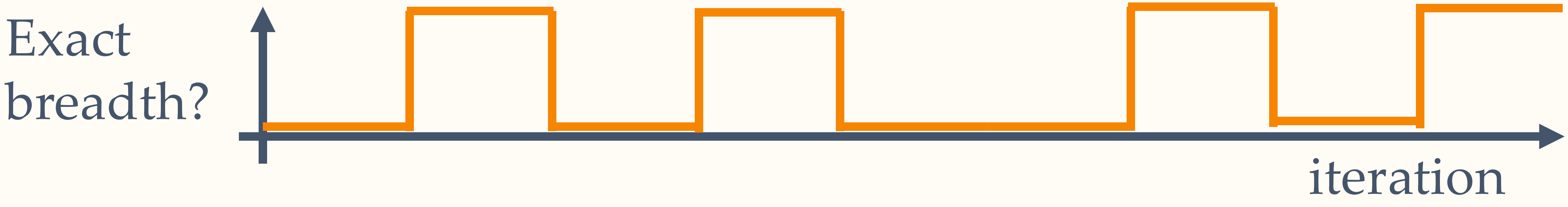
Theorem. [KW'25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$



Some “Fine-Grained” Definitions of Breadth

Implication: For non-identifiable collections exact breadth is violated infinitely many times

Theorem. [KW’25] There is a generator $\mathcal{G}$, that achieves exact breadth *infinitely* many times for *any* countable collection $\mathcal{L}$



Immediate Open Questions

1. Fine-grained trade-offs between hallucinations and breadth

   Partial results [CP'25], [KMV'24], [KW'25]

2. Allow multiple responses (could bypass impossiblity results)
3. What other type of feedback is useful?

   Partial results [KMV'25a], [CP'25]

Immediate Open Questions

1. Fine-grained trade-offs between hallucinations and breadth

   Partial results [CP'25], [KMV'24], [KW'25]

2. Allow multiple responses (could bypass impossiblity results)

3. What other type of feedback is useful?

   Partial results [KMV'25a], [CP'25]

COLT 2025
On Thursday in the
Language Model Session

Step Back – How Strong is Exact Breadth?

Example

Let $\mathcal{L}$ be a countable collection indexed by axiomatic systems

Step Back – How Strong is Exact Breadth?

Example

Let $\mathcal{L}$ be a countable collection indexed by axiomatic systems

1. $L \in \mathcal{L}$ is defined by some axiomatic system (e.g., ZFC)

Step Back – How Strong is Exact Breadth?

Example

Let $\mathcal{L}$ be a countable collection indexed by axiomatic systems

1. $L \in \mathcal{L}$ is defined by some axiomatic system (e.g., ZFC)
2. L has strings of the form "$\langle$Theorem$\rangle$ $\langle$Proof$\rangle$"

Step Back – How Strong is Exact Breadth?

Example

Let $\mathcal{L}$ be a countable collection indexed by axiomatic systems

1. $L \in \mathcal{L}$ is defined by some axiomatic system (e.g., ZFC)
2. L has strings of the form "$\langle$Theorem$\rangle$ $\langle$Proof$\rangle$"

Let K be defined by ZFC system (there is a TM that enumerates K)

Step Back – How Strong is Exact Breadth?

Example

Let $\mathcal{L}$ be a countable collection indexed by axiomatic systems

1. $L \in \mathcal{L}$ is defined by some axiomatic system (e.g., ZFC)
2. L has strings of the form "$\langle$Theorem$\rangle$ $\langle$Proof$\rangle$"

Let K be defined by ZFC system (there is a TM that enumerates K)

If a generator achieves exact breadth for K (in a prompted model), it can be used to prove all *provable* statements

Step Back – How Strong is Exact Breadth?

Example

Let $\mathcal{L}$ be a countable collection indexed by axiomatic systems

1. $L \in \mathcal{L}$ is defined by some axiomatic system (e.g., ZFC)
2. L has strings of the form "$\langle$Theorem$\rangle$ $\langle$Proof$\rangle$"

Let K be defined by ZFC system (there is a TM that enumerates K)

If a generator achieves exact breadth for K (in a prompted model), it can be used to prove all *provable* statements

...even without knowing the axiomatic system

Step Back – How Strong is Exact Breadth?

Example

Let $\mathcal{L}$ be a countable collection indexed by axiomatic systems

1. $L \in \mathcal{L}$ is defined by some axiomatic system (e.g., ZFC)
2. L has strings of the form "$\langle$Theorem$\rangle$ $\langle$Proof$\rangle$"

Let K be defined by ZFC system (there is a TM that enumerates K)

If a generator achieves exact breadth for K (in a prompted model), it can be used to prove all *provable* statements

...even without knowing the axiomatic system

Does not contradict Godel (does contradict Turing's decidability)

References

[CP'25] Exploring Facets of Language Generation in the Limit, COLT'25
[HKMV'25] On Union-Closedness of Language Generation, arXiv'25
[KM'24] Language Generation in the Limit, NeurIPS'24
[KW'24] Density Measures for Language Generation, arXiv'25
[KMV'25] On the Limits of Language Generation: … STOC'25
[KMV'24] Characterizations of Language Generation With Breadth, arXiv'24
[PRR'25] Representative Language Generation, ICML'25

Thank you!