Linguistics and formal languages

Marina Ermolaeva

University of Chicago

LING 21020: Formal Foundations of Linguistics
April 6, 2020

Hi!

Welcome to Formal Foundations of Linguistics: Remote Edition!

Communication outside class

- Canvas
  - homework, readings, and other class materials
  - all announcements (please keep your notifications on!)

- Zoom chat channel: *Formal Foundations of Linguistics*
  - office hours: WTh 3:00–4:00pm Central Time
  - general discussion and quick questions

Sources

- No required textbook, but (optional) readings will be posted

- Special mention:

  An introductory linguistics textbook to consult for review:

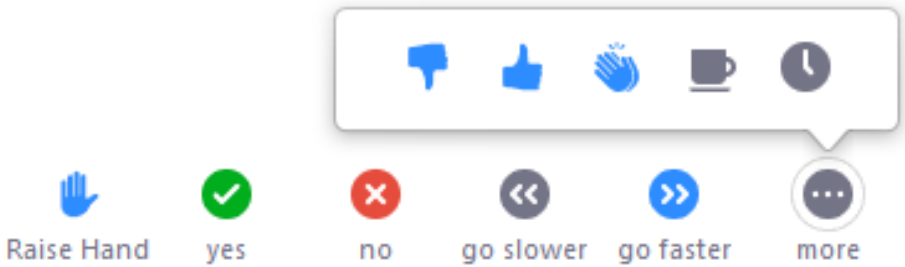
  Hope C. Dawson and Michael Phelan (eds), *Language Files* (2016)

  A major inspiration and source of examples used in this course:

  Thomas Graf, *Computational Linguistics as Language Science* (2019)

Communication in class

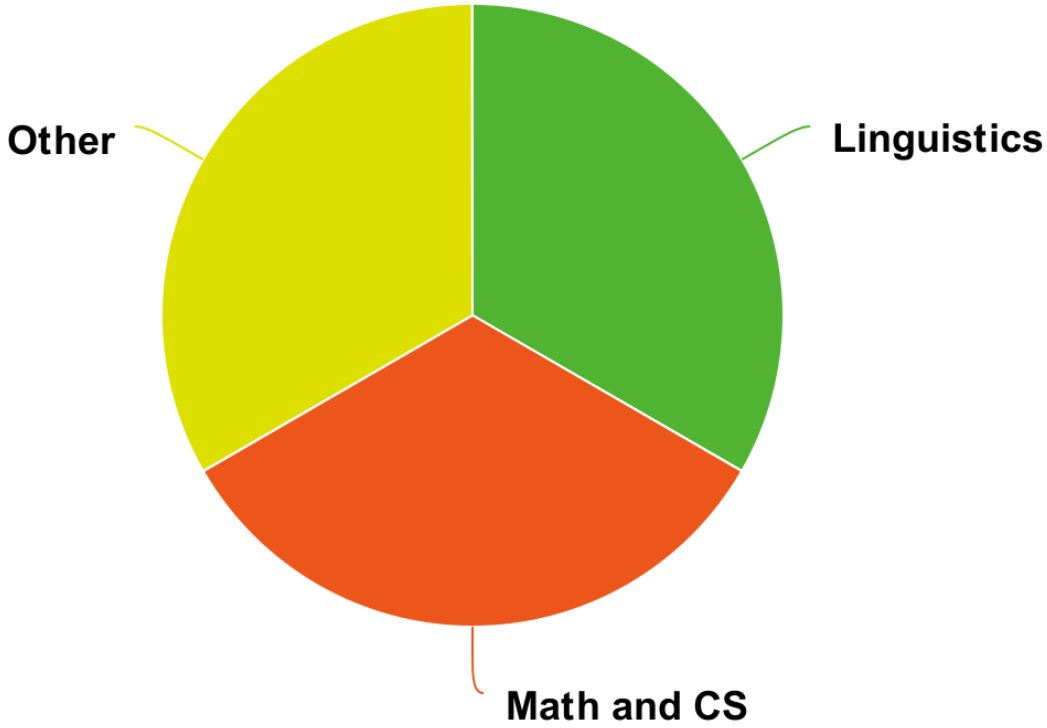
- Chat messages
  - for short questions and comments



- Nonverbal feedback
  - to give me an idea of how the class is going
  - I will sometimes ask you to use this feature for a quick poll

- Audio + video
  - for extended discussion; please raise your hand first
  - make sure to keep your mic muted when not talking

Background survey (preliminary results)



(if you haven't taken the survey yet, please do so after class!)

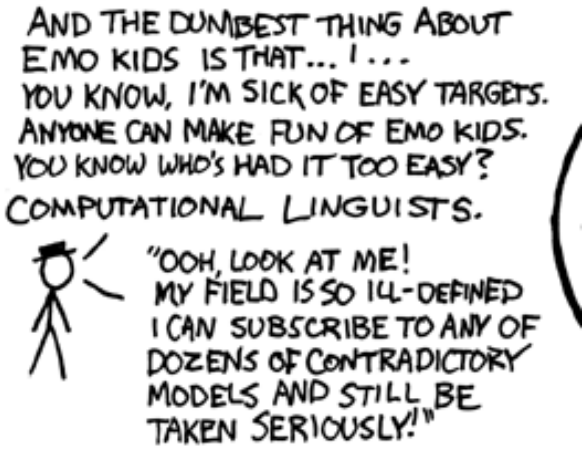
Before we continue...

- Record our Zoom meetings: yes or no?

- Let's have a vote!

Update: "yes" – 6, "no" – 0, "either is OK" – 6.

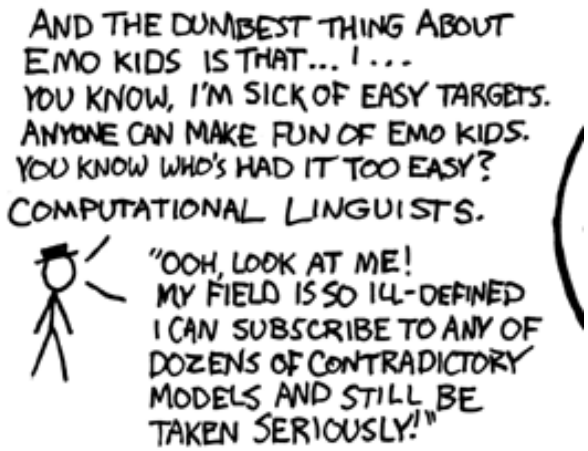
Language + Math = ?



(source: xkcd)

- Computational linguistics
  – often used as an umbrella term
- Natural language processing
  – using computers to solve practical language-related tasks
- Mathematical linguistics
  – applying mathematical methods to understand natural language
  – **formal language theory**: languages as mathematical objects generated by rule systems

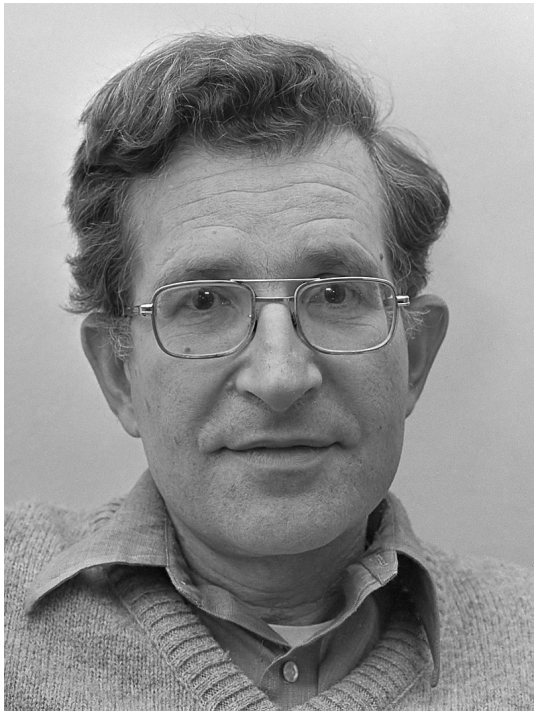
Language + Math = ?



(source: xkcd)

- Computational linguistics
  – often used as an umbrella term
- Natural language processing
  – using computers to solve practical language-related tasks
- Mathematical linguistics ☜
  – applying mathematical methods to understand natural language
  – **formal language theory**: languages as mathematical objects generated by rule systems

Formalization in linguistics



"The search for rigorous formulation in linguistics has a much more serious motivation than mere concern for logical niceties or the desire to purify well-established methods of linguistic analysis. [...] By pushing a precise but inadequate formulation to an unacceptable conclusion, we can often expose the exact source of this inadequacy and, consequently, gain a deeper understanding of the linguistic data."

Noam Chomsky, *Syntactic Structures* (1957)

Formal languages

Formal languages

- **Alphabet**: a finite set of symbols

Formal languages

- **Alphabet**: a finite set of symbols
- Examples:

Formal languages

- **Alphabet**: a finite set of symbols
- Examples:
  - $\{a,\ b,\ c, ..., z\}$ is the alphabet of Latin characters

Formal languages

- **Alphabet**: a finite set of symbols
- Examples:
  - $\{a,\ b,\ c, ..., z\}$ is the alphabet of Latin characters
  - $\{x,\ y,\ +,\ -,\ =\}$ is an alphabet containing alphanumeric characters

Formal languages

- **Alphabet**: a finite set of symbols
- Examples:
  - $\{a,\ b,\ c, ..., z\}$ is the alphabet of Latin characters
  - $\{x,\ y,\ +,\ -,\ =\}$ is an alphabet containing alphanumeric characters
  - $\{the,\ be,\ to,\ of,\ and\}$ is the alphabet of five most common English words, according to Wikipedia

Formal languages

- **Alphabet**: a finite set of symbols
- Examples:
  - $\{a,\ b,\ c, ..., z\}$ is the alphabet of Latin characters
  - $\{x,\ y,\ +,\ -,\ =\}$ is an alphabet containing alphanumeric characters
  - $\{\textit{the},\ \textit{be},\ \textit{to},\ \textit{of},\ \textit{and}\}$ is the alphabet of five most common English words, according to Wikipedia
  - $\mathbb{N} = \{0, 1, 2, 3, ...\}$, the set of natural numbers, is **not** an alphabet

Formal languages

- **Alphabet**: a finite set of symbols
- Examples:
  - $\{a,\ b,\ c, ..., z\}$ is the alphabet of Latin characters
  - $\{x,\ y,\ +,\ -,\ =\}$ is an alphabet containing alphanumeric characters
  - $\{\textit{the},\ \textit{be},\ \textit{to},\ \textit{of},\ \textit{and}\}$ is the alphabet of five most common English words, according to Wikipedia
  - $\mathbb{N} = \{0, 1, 2, 3, ...\}$, the set of natural numbers, is **not** an alphabet
  - The set of all grammatical sentences of English is **not** an alphabet

Formal languages

- **Alphabet**: a finite set of symbols, often denoted by capital sigma: Σ

  Example: $\Sigma = \{a,\ b\}$

- Σ^*: the set of all finite strings of symbols from Σ

  We use ϵ to denote the **empty string**

  $\Sigma^* = \{\epsilon,\ a,\ b,\ aa,\ ab,\ ba,\ bb,\ aaa,\ ...\}$

- **Language** over Σ: a (finite or infinite) subset of Σ^*

Formal languages

- **Alphabet**: a finite set of symbols, often denoted by capital sigma: Σ

  Example: $\Sigma = \{a,\ b\}$

- Σ^*: the set of all finite strings of symbols from Σ

  We use ϵ to denote the **empty string**

  $\Sigma^* = \{\epsilon,\ a,\ b,\ aa,\ ab,\ ba,\ bb,\ aaa,\ ...\}$

- **Language** over Σ: a (finite or infinite) subset of Σ^*

  Examples of languages: $\emptyset$, Σ, Σ^*,
  all 5-symbol strings of *a*'s and *b*'s,
  all even-length strings of only *a*'s
  ...

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  [[[ ]]]

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  $[\,[\,[\,]\,]\,]$ (yes)

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  [[[ ]]] (yes)

  [[ ]] [[ ]]]

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  [[[ ]]] (yes)

  [[ ]] [[ ]]] (no)

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  [[[ ]]] (yes)

  [[ ]] [[ ]]] (no)

  [[ ] [ ] [ ]] [ ]

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  [[[ ]]] (yes)

  [[ ]] [[ ]]] (no)

  [[ ] [ ] [ ]] [ ] (yes)

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  [[[ ]]] (yes)

  [[ ]] [[ ]]] (no)

  [[ ] [ ] [ ]] [ ] (yes)

  ϵ

Example: square brackets

- Let $\Sigma = \{ [,] \}$
- Let L be a language over Σ such that:

  (1) $[\,] \in L$;

  (2) For any string $s \in L$, $[s] \in L$;

  (3) For any two strings $s, t \in L$, $st \in L$

- Are these strings in L?

  [[[ ]]] (yes)

  [[ ]] [[ ]]] (no)

  [[ ] [ ] [ ]] [ ] (yes)

  ϵ (no)

Relevant questions

- What kinds of formal languages are there?
- What languages are more complicated than others?

Relevant questions

- What kinds of formal languages are there?
- What languages are more complicated than others?

We will touch on these, but we are primarily interested in the following:

Relevant questions

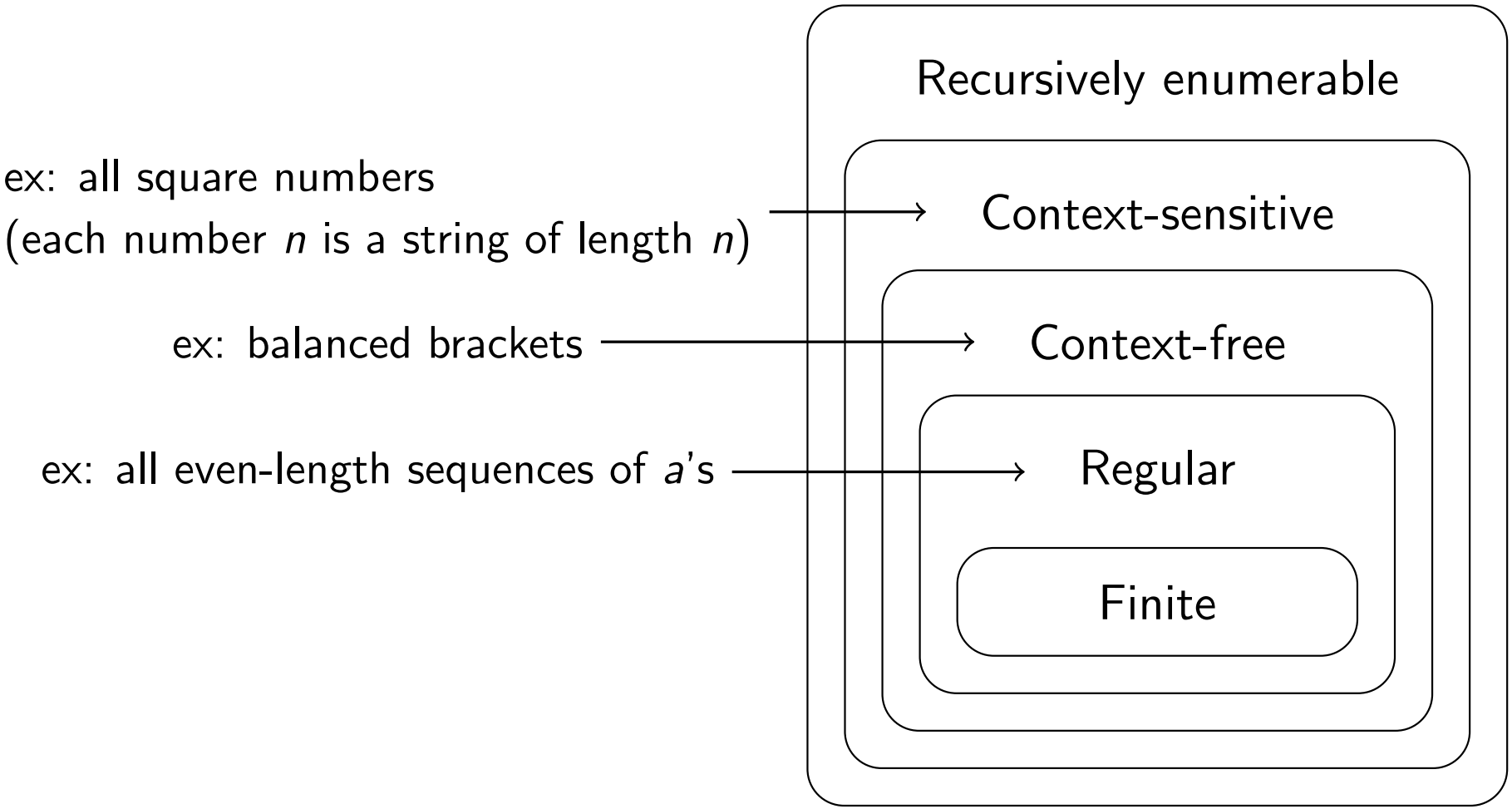
- What kinds of formal languages are there?
- What languages are more complicated than others?

  We will touch on these, but we are primarily interested in the following:

- What formal languages are a better fit to model natural language?

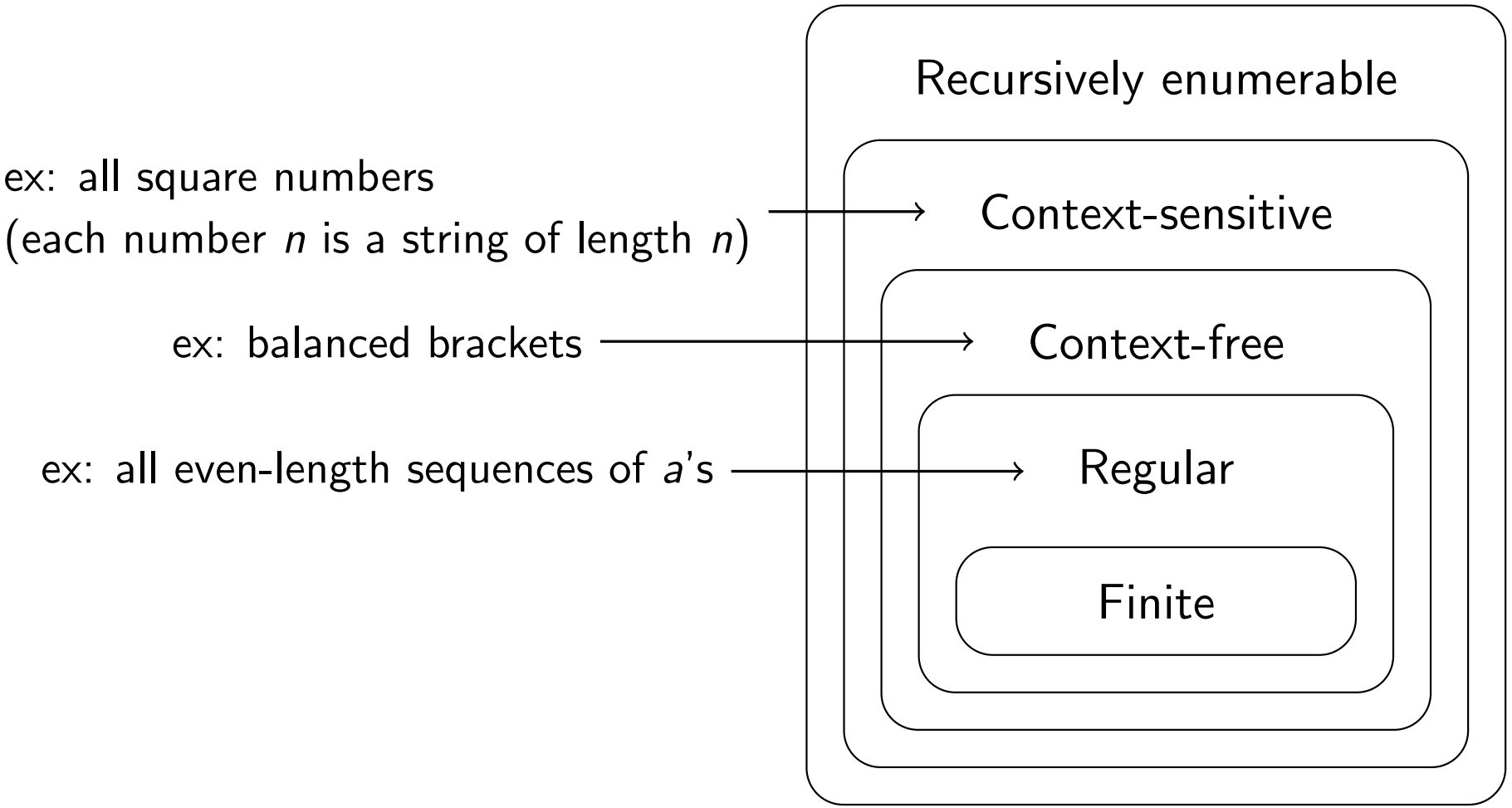
The Chomsky Hierarchy of formal languages

(also known as the Chomsky–Schützenberger hierarchy)



The Chomsky Hierarchy of formal languages

(also known as the Chomsky–Schützenberger hierarchy)



- Where do phonology, morphology, and syntax fit in this hierarchy?

What this course is NOT about

What this course is NOT about

- Intro to linguistics

What this course is NOT about

- Intro to linguistics

...but we will look at some basic concepts of linguistics in a more precise way

What this course is NOT about

- Intro to linguistics

...but we will look at some basic concepts of linguistics in a more precise way

- Intro to formal language theory

What this course is NOT about

- Intro to linguistics

  ...but we will look at some basic concepts of linguistics in a more precise way

- Intro to formal language theory

  ...but we will use formal grammars to describe patterns in natural language

What this course is NOT about

- Intro to linguistics

  ...but we will look at some basic concepts of linguistics in a more precise way

- Intro to formal language theory

  ...but we will use formal grammars to describe patterns in natural language

- Natural language processing

What this course is NOT about

• Intro to linguistics	...but we will look at some basic concepts of linguistics in a more precise way
• Intro to formal language theory	...but we will use formal grammars to describe patterns in natural language
• Natural language processing	...but we will use various tools implementing these grammars

Background survey

- Regular expressions
- Finite-state automata
- Context-free languages
- Subregular languages
- Mildly context-sensitive languages
- International Phonetic Alphabet
- Phonemes and allophones
- Phonological rewriting rules
- Autosegmental phonology
- Syntactic constituents
- Phrase-structure rules
- Minimalist syntax

Background survey

- Regular expressions
- Finite-state automata
- Context-free languages (formal language theory)
- Subregular languages
- Mildly context-sensitive languages

- International Phonetic Alphabet
- Phonemes and allophones
- Phonological rewriting rules
- Autosegmental phonology (linguistics)
- Syntactic constituents
- Phrase-structure rules
- Minimalist syntax

What this course IS about

What this course IS about

What these...

- Regular expressions
- Finite-state automata
- Subregular languages
- Context-free languages
- Mildly context-sensitive languages
- ...

What this course IS about

What these...

- Regular expressions
- Finite-state automata
- Subregular languages
- Context-free languages
- Mildly context-sensitive languages
- ...

...have to do with these

- Phonemes and allophones
- Phonological rewriting rules
- Autosegmental phonology
- Phrase-structure rules
- Minimalist syntax
- ...

What this course IS about

What these...

- Regular expressions
- Finite-state automata
- Subregular languages
- Context-free languages
- Mildly context-sensitive languages
- ...

...have to do with these

- Phonemes and allophones
- Phonological rewriting rules
- Autosegmental phonology
- Phrase-structure rules
- Minimalist syntax
- ...

...and how we can use the connection to model natural language phenomena

Next time

- Formal grammars

- Regular languages, and why linguists should care about them