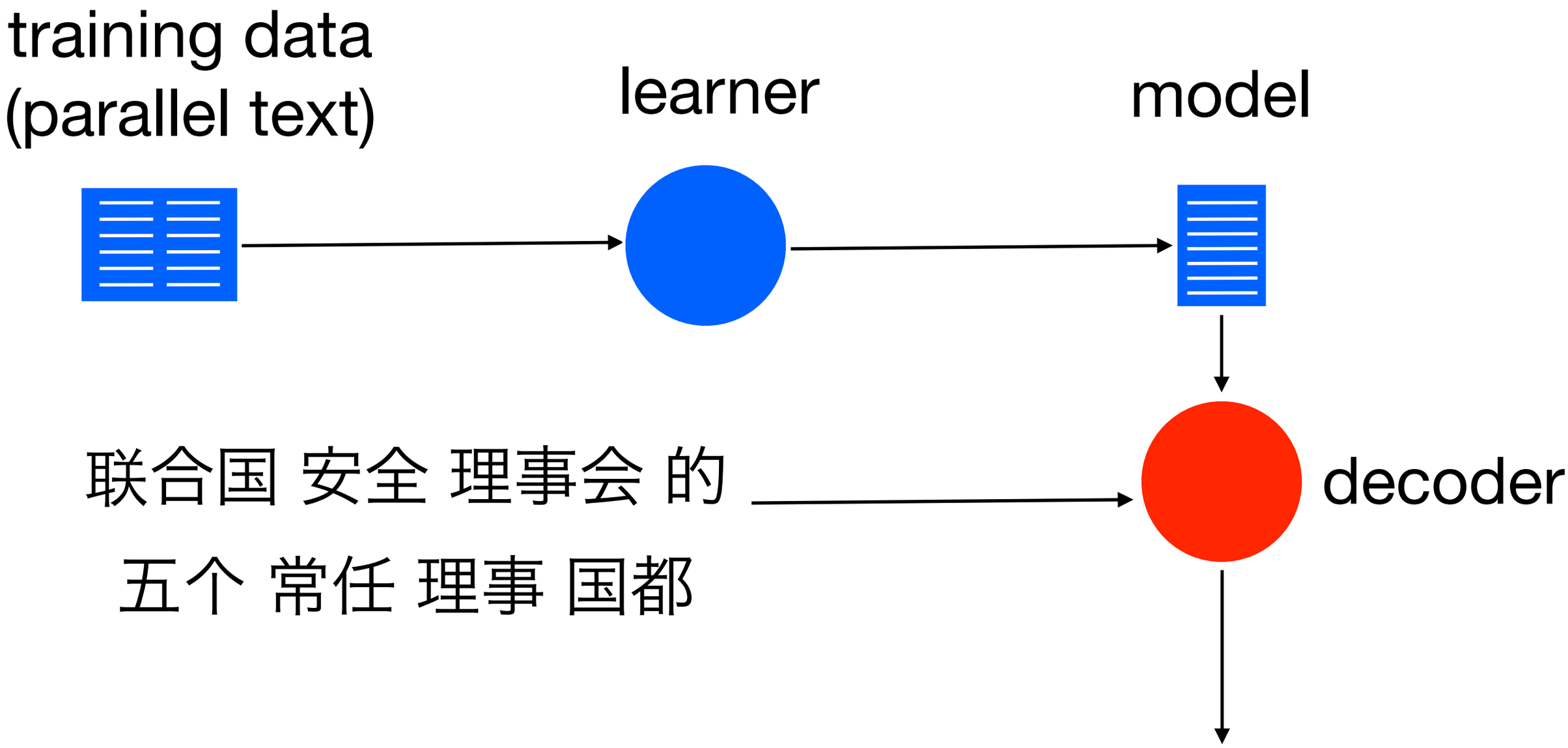
Decoding

The Story So Far...



SCHEDULE

- TODAY

    - word-based translation (conceptual)

- THURSDAY

    - incorporating model scores

    - efficiency considerations

Decoding

- alignment
  - *input:* parallel sentences
  - *task:* learn a model of translation (probabilities of word translations, fertilities, and distortions)

- decoding
  - *input*: foreign-language sentence and a model
  - *task:* find the model's favorite translation

DECODING

- *the process of producing a translation of a sentence*

- Two main problems:

  - **modeling** How do we score translations?

  - **search** How do we find the model's preferred translation?

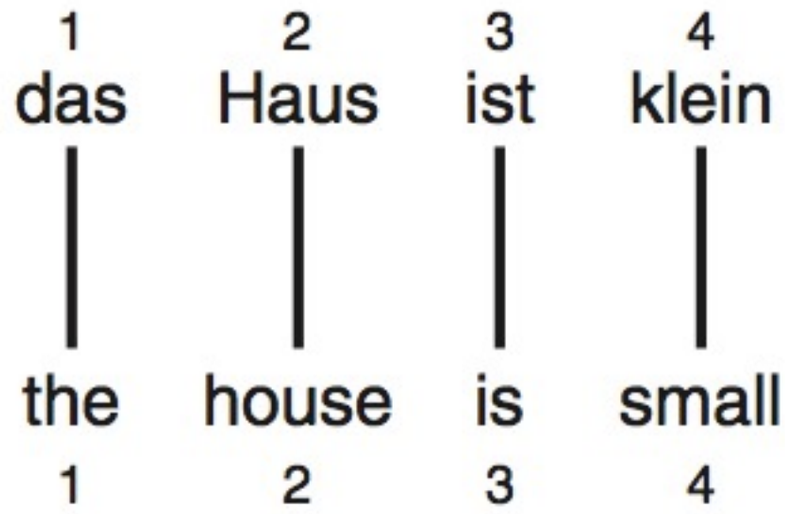
Decoding (Modeling)

- Decoding stories

  - **Model 1**: *translate each word, scramble them*

  - parameters: p(e | f)

Model 1



DECODING (MODELING)

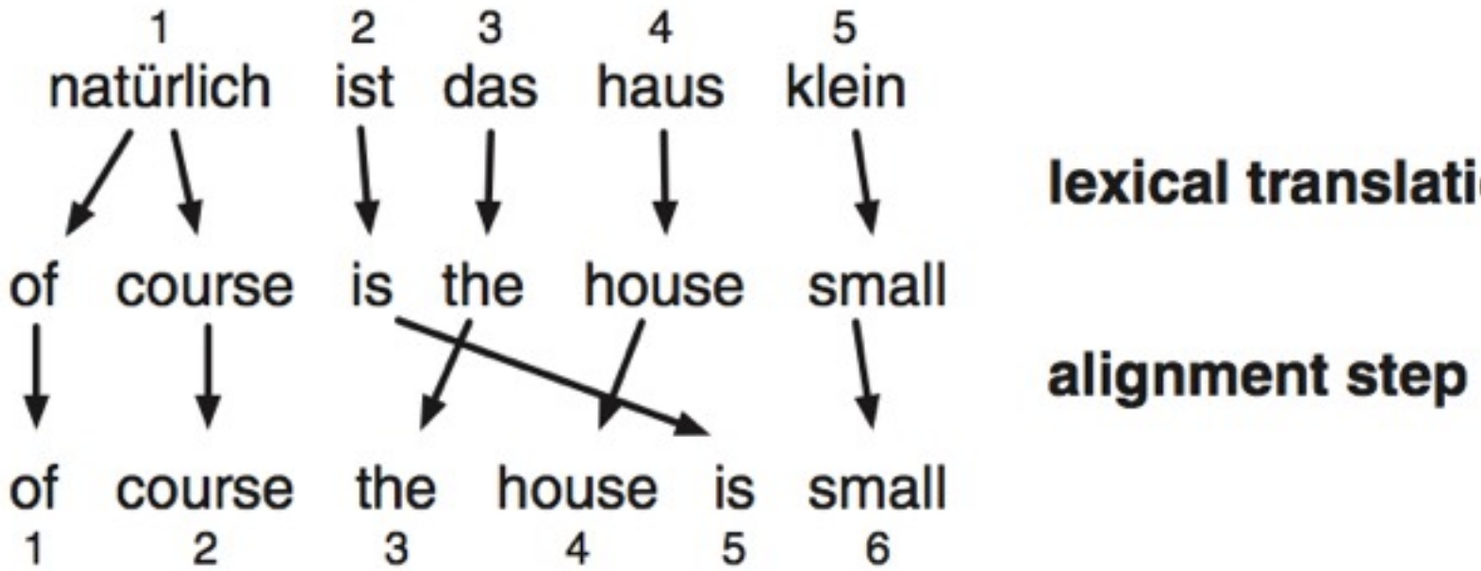
- Decoding stories

  - **Model 2**: *translate each word, order them conditioned on absolute position*

    - parameters: p(e | f)

    - new parameters: a(i | j, |E|, |F|)

MODEL 2


1 2 3 4 5
natürlich ist das haus klein
of course is the house small
of course the house is small
1 2 3 4 5 6
lexical translation step
alignment step

Decoding (Modeling)

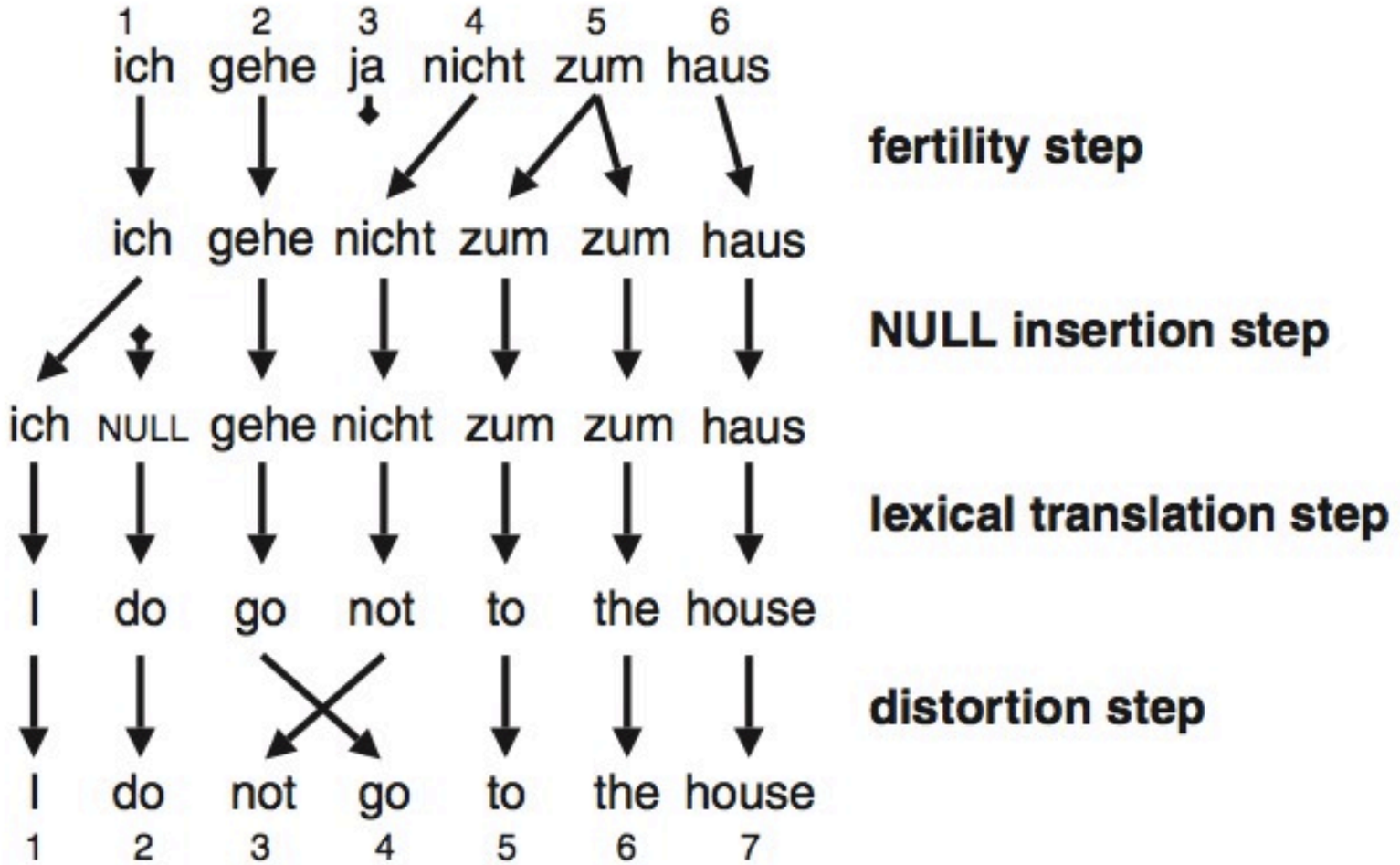
- Decoding stories

  - **Model 3**: *model how often each word likes to get translated*

    - parameters: p(e | f), a(i | j, |E|, |F|)

    - new parameters: n(ϕ | f)

Model 3


1 2 3 4 5 6
ich gehe ja nicht zum haus
fertility step
ich gehe nicht zum zum haus
NULL insertion step
ich NULL gehe nicht zum zum haus
lexical translation step
I do go not to the house
distortion step
I do not go to the house
1 2 3 4 5 6 7

Decoding (Modeling)

- These models are actually quite a bit more complicated than what we actually use in word-based decoding

- Today: you're the model

- Thursday: we'll use a real, learned model

Decoding (Search)

- How do we find the translation?

- Easy way: generate all sentences, score them with the model

- Formulate this as stack-based decoding

Decoding (Search)

- Start with a list of hypotheses, containing only the empty hypothesis

- For each stack

    - For each hypothesis

        - For each applicable word

            - Extend the hypothesis with the word

            - Place the new hypothesis on the right stack

Demo

Simple Spanish example (ttable.js)

Problem

- Which hypothesis should we extend?
  - *Long hypotheses compete with short ones*
  - *Use multiple stacks for organization*

Demo

Longer Spanish example (ttable-spanish.js)

Problem

- There are too many hypotheses!
  - *Restrict where the next word can come from.*
    - *Monotonic translations*
      - *Within* n *words, $n \in \{1,2,3\}$*
    - *Anywhere*

Activity

http://cs.jhu.edu/~post/mt-class/stack-decoder/index.html

Instructions (10 minutes)

Get in groups, and build out the complete table hypothesis chart using different constraints:

- monotonic
- 1 word distortion
- 2 word distortion
- 3 word distortion

CONCEPTS

- **distortion**: which words are permitted to extend a hypothesis

- for Thursday

  - **pruning:** *histogram* and *beam/threshold*

  - **dynamic programming**

  - **beam** and **threshold** pruning

Demo

German example (from textbook; ttable-german.js)

TYPES OF ERRORS

- *Search errors*
  - we didn't look hard enough (or smart enough)
  - shortcuts we took excluded good hypotheses
- *Model errors*
  - the solution was impossible under our assumptions
  - the ideal answer cannot be obtained by the model