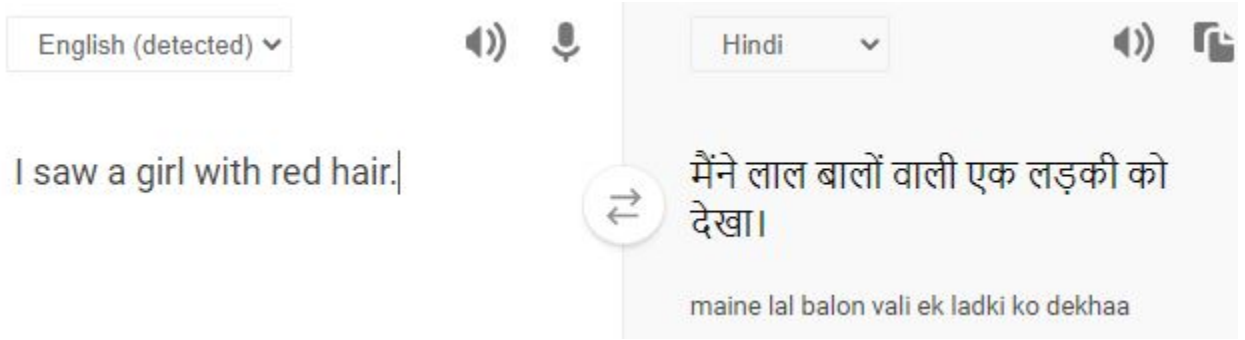
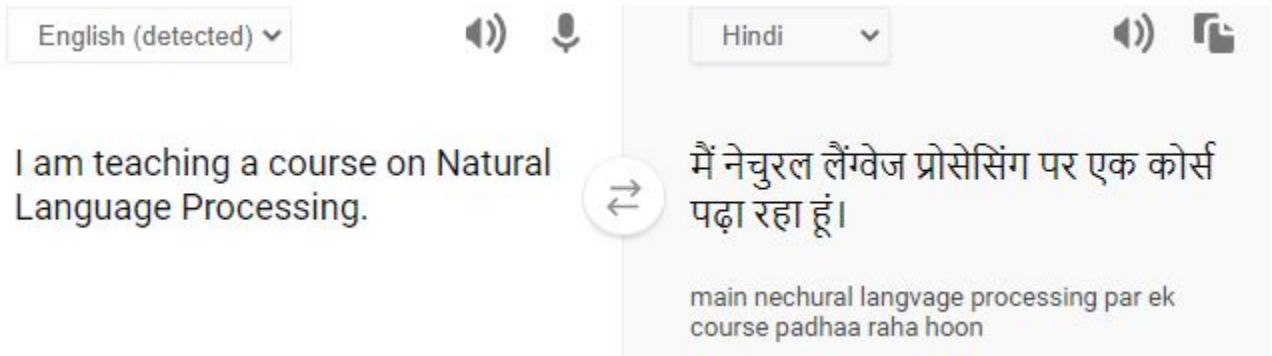
CS60075
Natural Language Processing
Autumn 2020

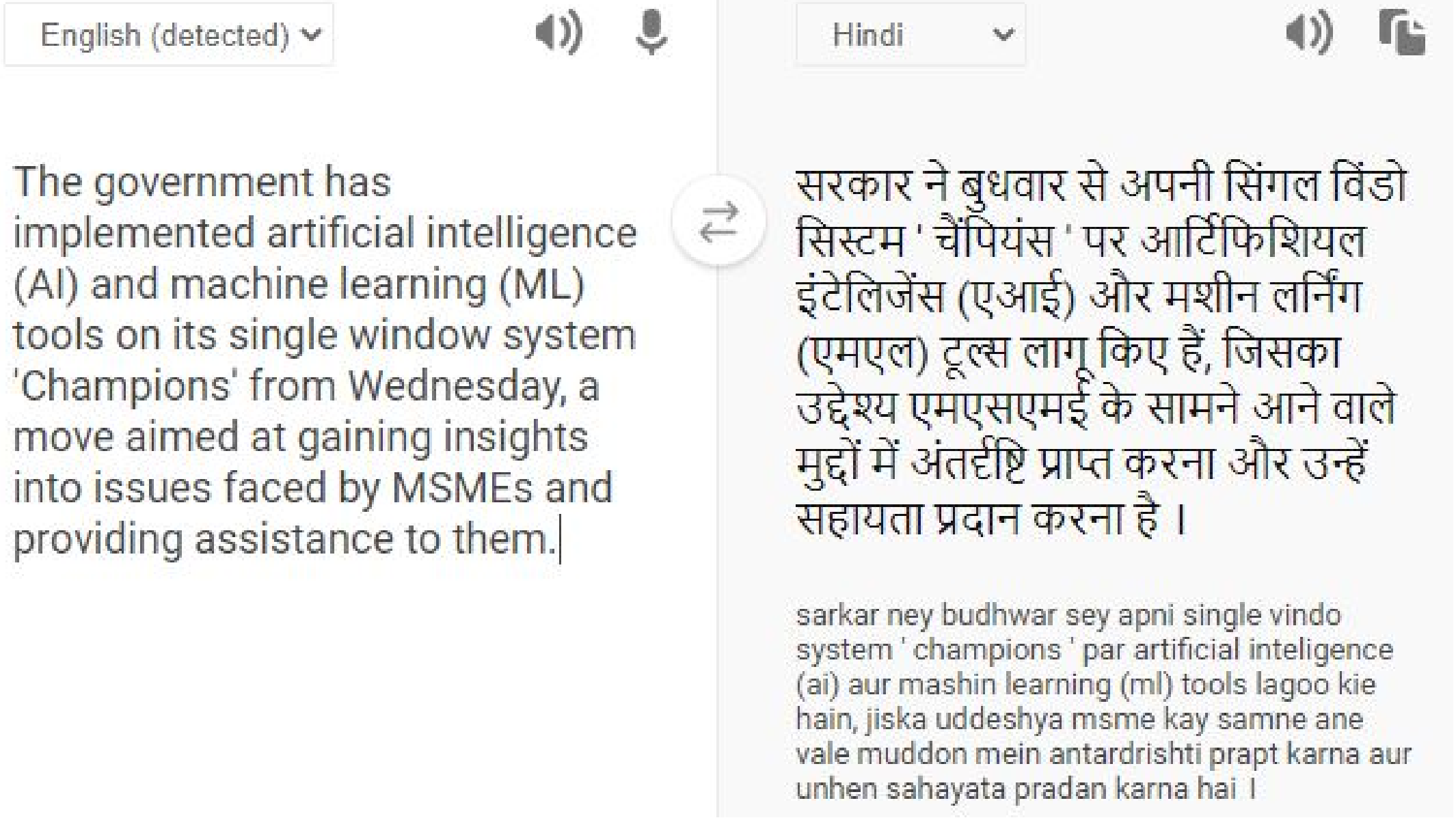
Module 7:

Machine Translation 1

15 October 2020


English (detected)
Hindi
I saw a girl with red hair.
मैंने लाल बालों वाली एक लड़की को देखा।
maine lal balon vali ek ladki ko dekhaa


English (detected)
Hindi
I am teaching a course on Natural Language Processing.
मैं नेचुरल लैंग्वेज प्रोसेसिंग पर एक कोर्स पढ़ा रहा हूं।
main nechural langvage processing par ek course padhaa raha hoon


English (detected)
Hindi
The government has implemented artificial intelligence (AI) and machine learning (ML) tools on its single window system 'Champions' from Wednesday, a move aimed at gaining insights into issues faced by MSMEs and providing assistance to them.
सरकार ने बुधवार से अपनी सिंगल विंडो सिस्टम ' चैंपियंस ' पर आर्टिफिशियल इंटेलिजेंस (एआई) और मशीन लर्निंग (एमएल) टूल्स लागू किए हैं, जिसका उद्देश्य एमएसएमई के सामने आने वाले मुद्दों में अंतर्दृष्टि प्राप्त करना और उन्हें सहायता प्रदान करना है ।
sarkar ney budhwar sey apni single vindo system ' champions ' par artificial inteligence (ai) aur mashin learning (ml) tools lagoo kie hain, jiska uddeshya msme kay samne ane vale muddon mein antardrishti prapt karna aur unhen sahayata pradan karna hai ।

Challenges for MT

Ambiguities

- Words
- Morphology
- Syntax
- Semantics
- Pragmatics

- Gaps in data
  - Availability of corpus
  - Commonsense knowledge

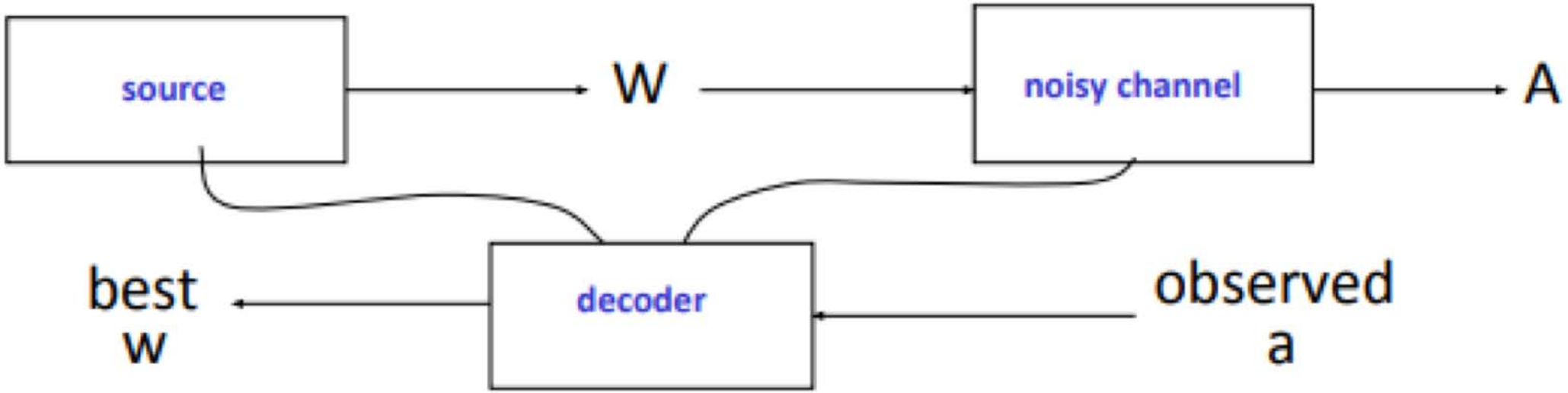
- Understanding of context, connotation, social norms, etc

- How can we formalize the process of learning to translate?
- How can we formalize the process of finding translations for new inputs?
- If our model produces many outputs, how do we find the best one?
- If we have a gold standard translation, how can we tell if our output is good or bad?

Two views of MT

1. MT as code breaking
   - The Noisy Channel Model
2. MT as Direct Modelling

The Noisy-Channel Model



$$w^* = argmax_w P(w|a)$$

The Noisy-Channel Model

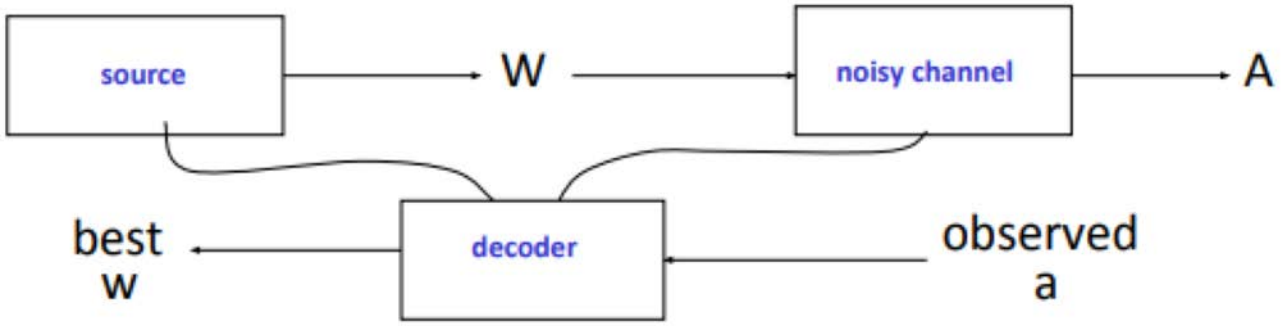
$$w^* = argmax_w P(w|a)$$

$$= argmax_w \; P(a|w) \; P(w)/P(a)$$

$$= argmax_w P(a|w)P(w)$$

Channel model

Source model

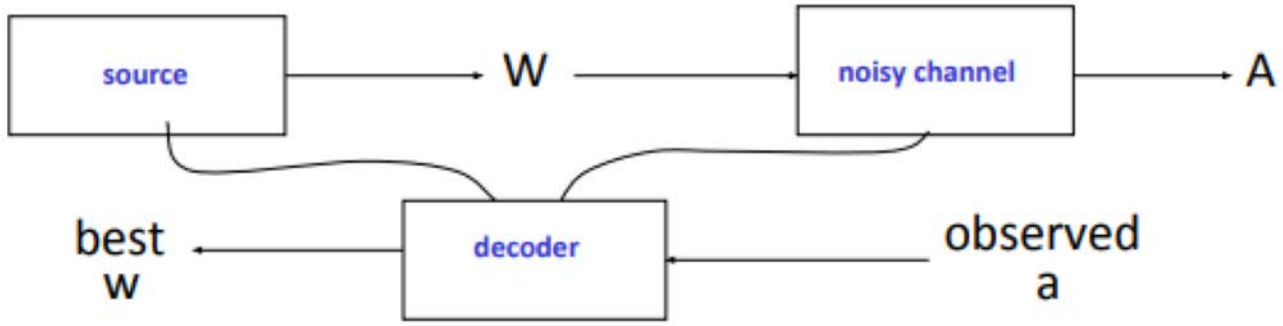


The Noisy-Channel Model

$$w^* = argmax_w P(w|a)$$

$$= argmax_w \; P(a|w) \; P(w)/P(a)$$

$$= argmax_w P(a|w)P(w)$$



Channel model

Source model

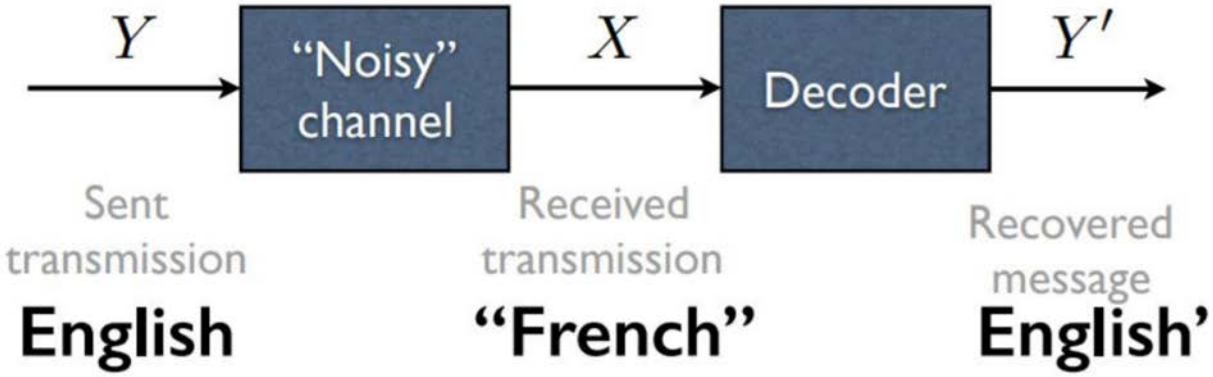
Likelihood
Acoustic model (HMMs)
Translation model

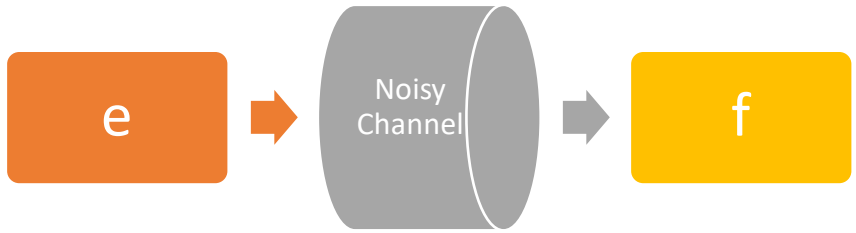
Prior
Language model: Distributions over sequence of words

The Noisy-Channel Model

- Analogy to information-theoretic model used to decode messages transmitted via a noisy communication channel.
- Assume that source sentence was generated by a "noisy" transformation of some target language sentence and then use Bayesian analysis to recover the most likely target sentence that generated it.

Translate foreign language sentence
$F=f_1, f_2, \ldots f_m$
to an English sentence
$\hat{E} = e_1, e_2, \ldots e_I$
that maximizes $\mathrm{P}(E \mid F)$





Task: To recover *e* from noisy *f.*

P(F|E): Translation model

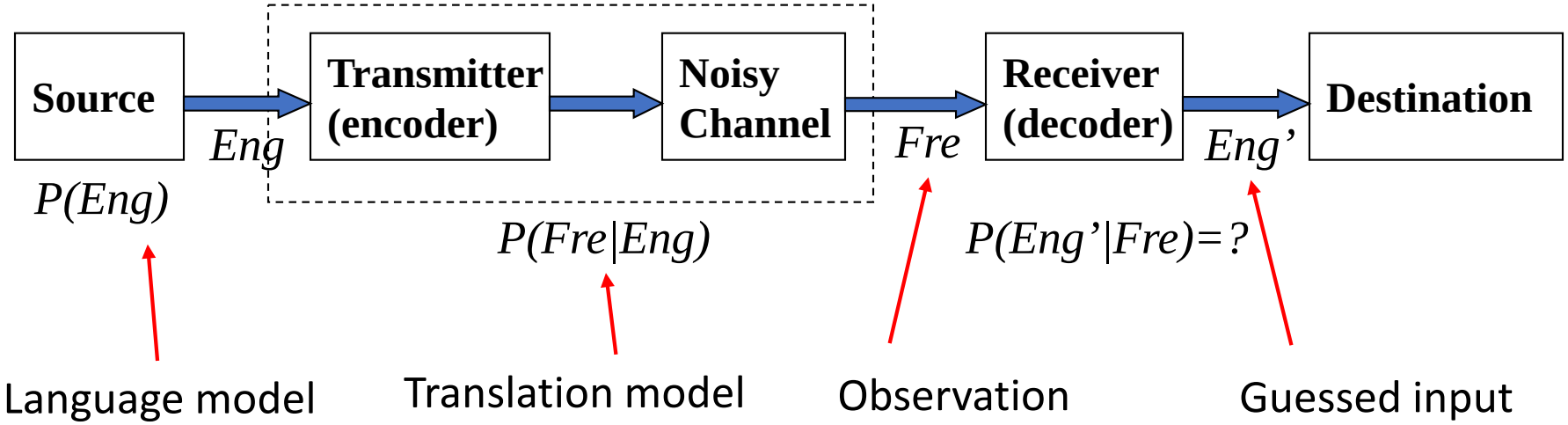
P(E): Language model

Bayesian Analysis of Noisy Channel

$$\hat{E} = \underset{E \in English}{\operatorname{argmax}} P(E \mid F)$$
$$= \underset{E \in English}{\operatorname{argmax}} \frac{P(F \mid E)P(E)}{P(F)}$$
$$= \underset{E \in English}{\operatorname{argmax}} P(F \mid E)P(E)$$

A decoder determines the most probable translation $\hat{E}$ given F

Translation Model Language Model



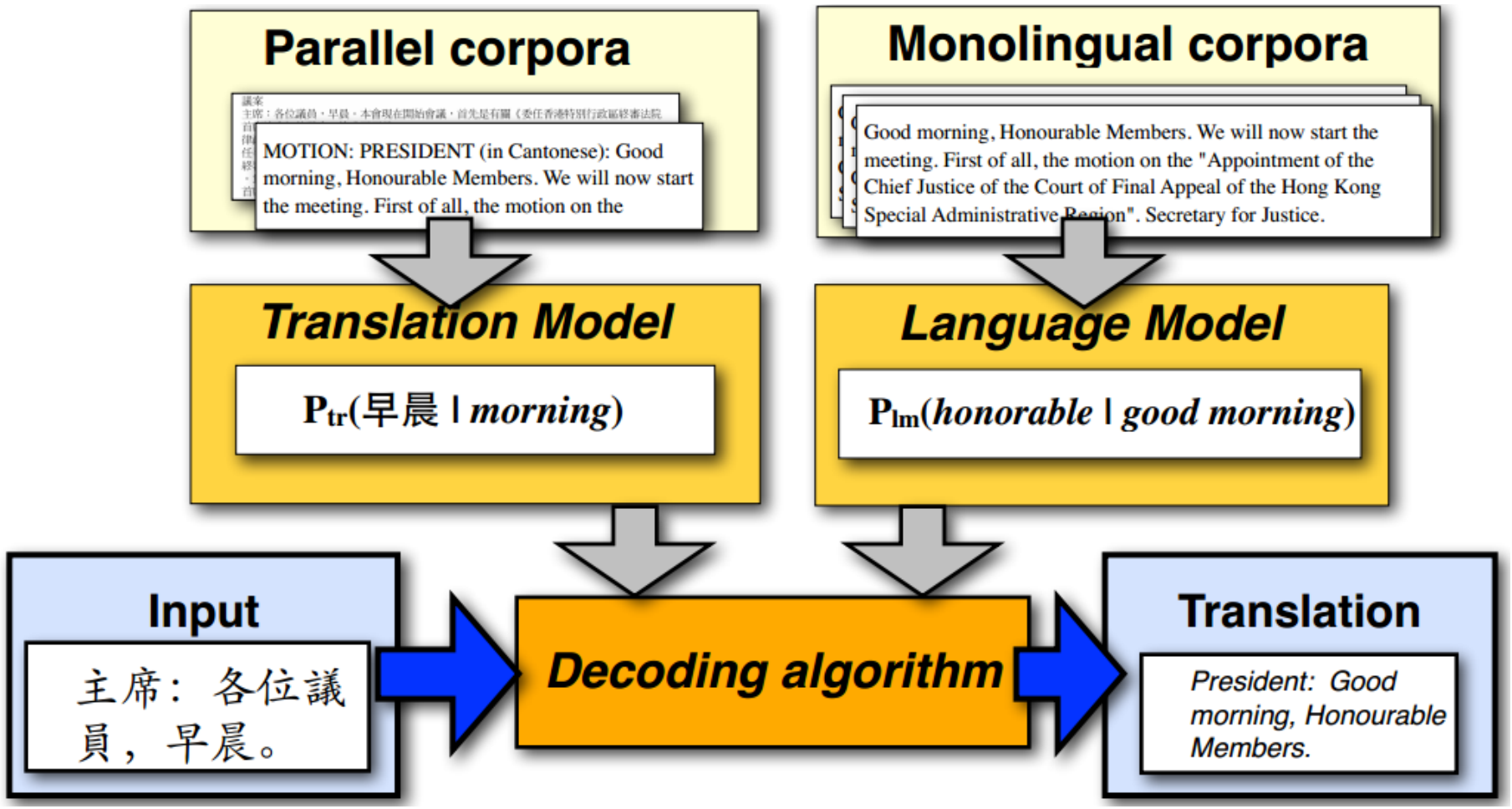
Two view of MT

- Code breaking (the noisy channel, Bayes rule)
  - Easy to use monolingual target language data
  - Search happens under a product of two models (individual models can be simple, product can be powerful)
- Direct modeling (pattern matching)
  - Directly model the process you care about
  - Model must be very powerful

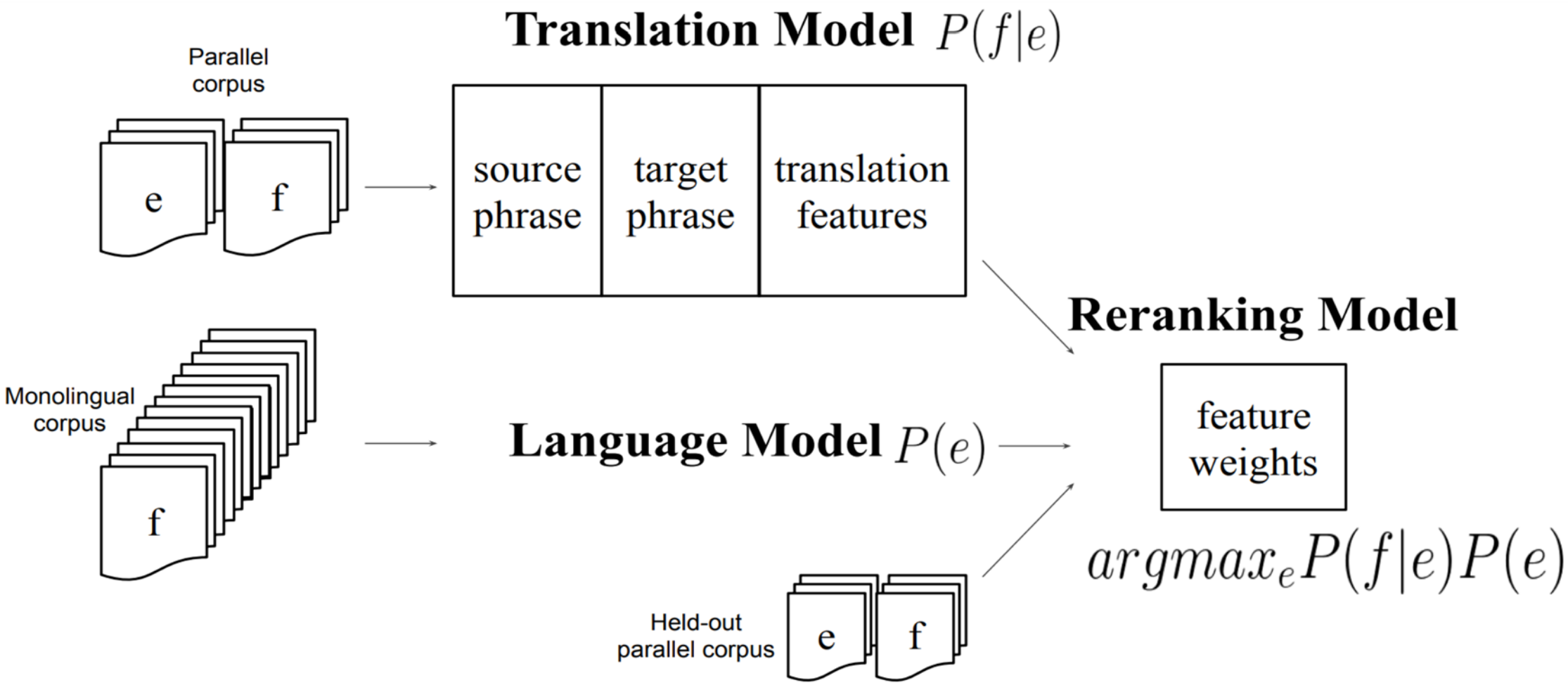
Where are we in 2020?

- Direct modeling is where most of the action is
  - Neural networks are very good at generalizing and conceptually very simple
  - Inference in “product of two models” is hard
- Noisy channel ideas are incredibly important and still play a big role in how we think about translation

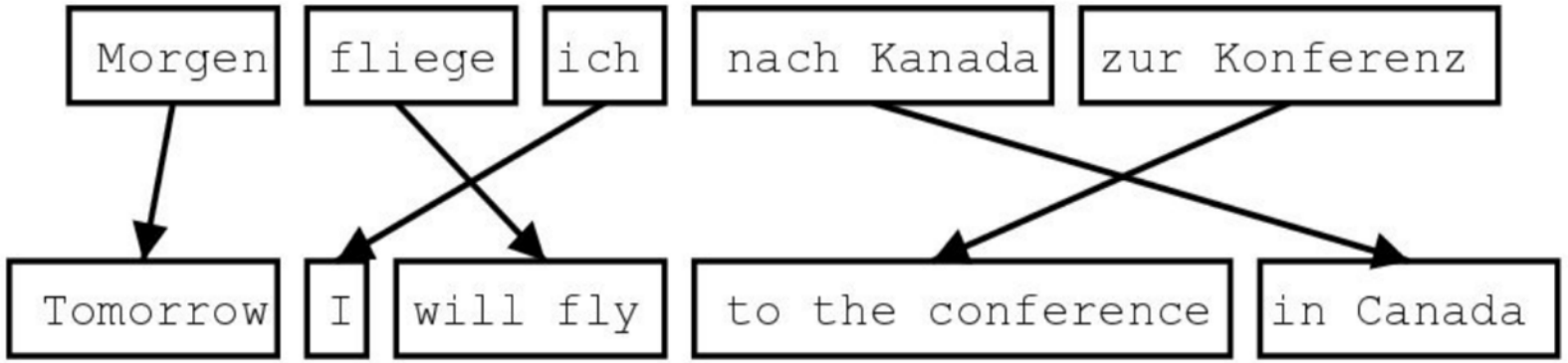
Statistical machine translation

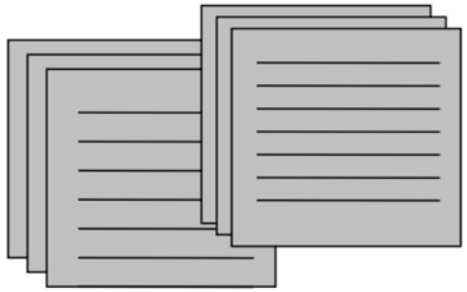


Noisy Channel: Phrase-Based MT

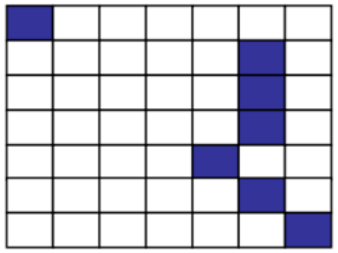


Construction of t-table





Sentence-aligned corpus



Word alignments

cat ||| chat ||| 0.9
the cat ||| le chat ||| 0.8
dog ||| chien ||| 0.8
house ||| maison ||| 0.6
my house ||| ma maison ||| 0.9
language ||| langue ||| 0.9
...

Phrase table (translation model)

Word Alignment Models

- Lexical Translation
  - How do we translate a word? Look it up in the dictionary
    - *Haus – house, building, home, household, shell*
  - Multiple translations
    - Some more frequent than others
    - Different word senses, different registers, different functions
    - *House, home* are common
  - *Shell* is specialized (the Haus of a snail is a shell)

How Common is Each?

- Look at a parallel corpus (German text along with English translation)

Translation of Haus	Count
house	8000
building	1600
home	200
household	150
shell	50

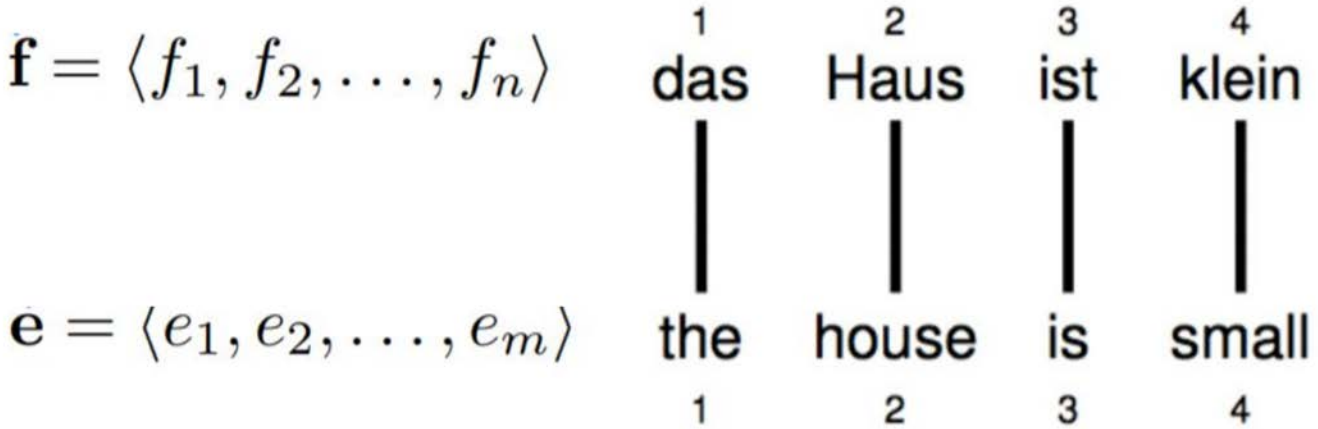
Estimate Translation Probabilities

- Maximum likelihood estimation

$$\hat{p}_{\mathrm{MLE}}(e \mid \texttt{Haus}) = \begin{cases} 0.8 & \text{if } e = \text{house}, \\ 0.16 & \text{if } e = \text{building}, \\ 0.02 & \text{if } e = \text{home}, \\ 0.015 & \text{if } e = \text{household}, \\ 0.005 & \text{if } e = \text{shell}. \end{cases}$$

Word Alignment

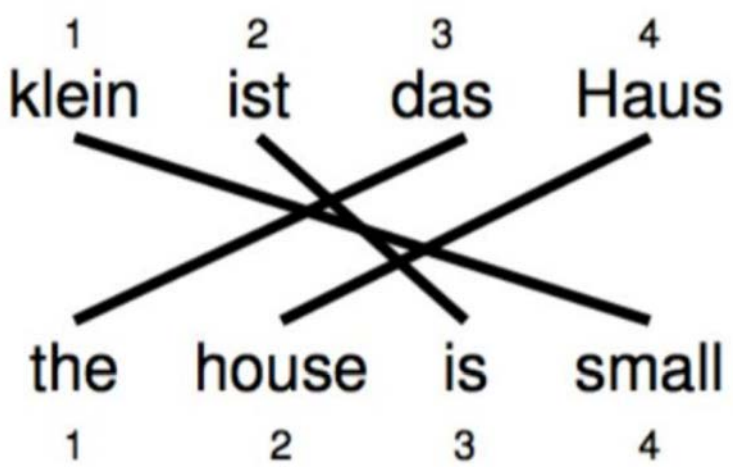
- Alignment can be visualized by drawing links between two sentences, and they are represented as vectors of positions



$$\mathbf{a} = (1, 2, 3, 4)^\top$$

Reordering

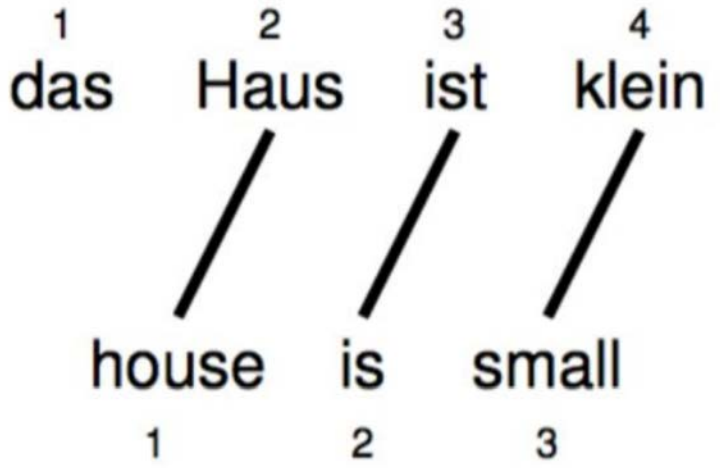
- Words may be reordered during translation



$$\mathbf{a} = (3, 4, 2, 1)^\top$$

Word Dropping

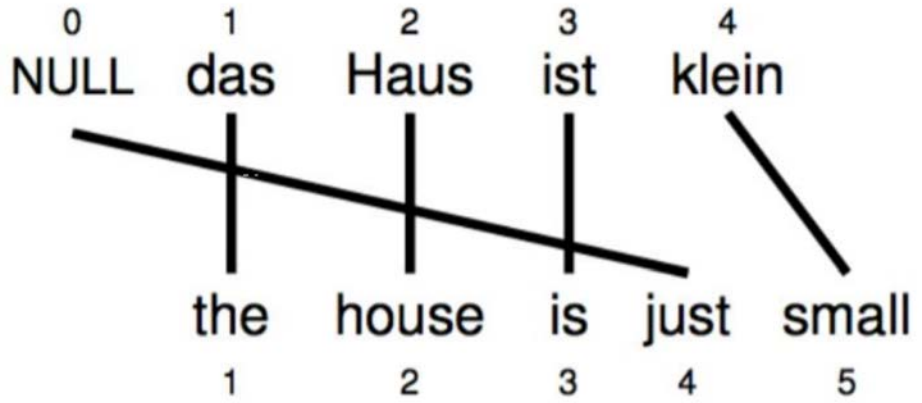
- A source word may not be translated at all



$$\mathbf{a} = (2, 3, 4)^\top$$

Word Insertion

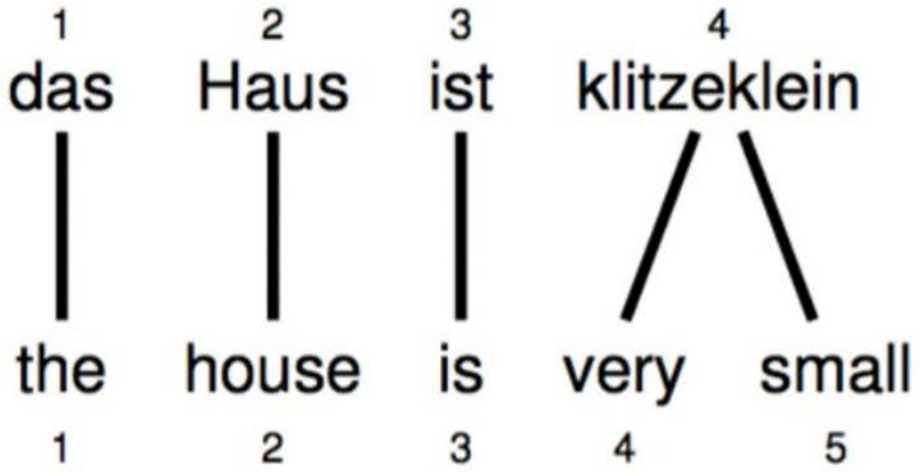
- Words may be inserted during translation
  - English just does not have an equivalent
  - But it must be explained – we typically assume every source sentence contains a NULL token



$$\mathbf{a} = (1, 2, 3, 0, 4)^\top$$

One-to-many Translation

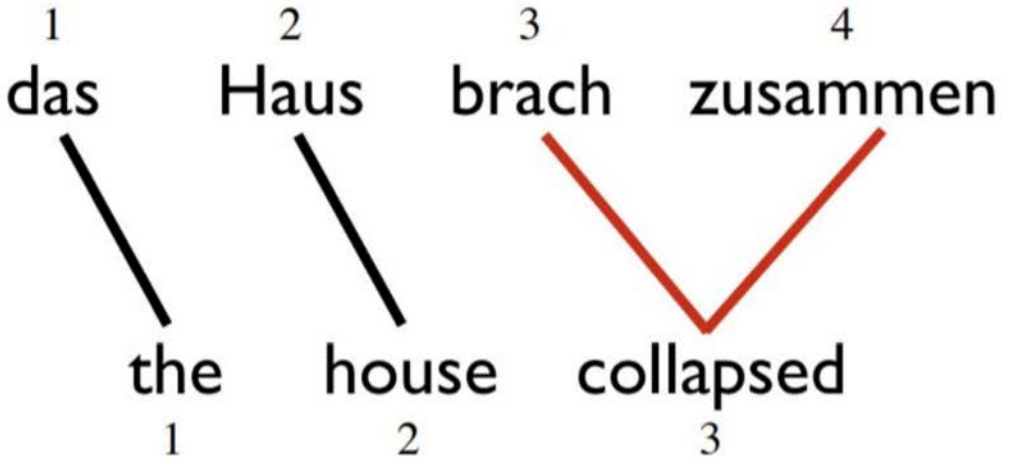
- A source word may translate into more than one target word



$$\mathbf{a} = (1, 2, 3, 4, 4)^\top$$

Many-to-one Translation

- More than one source word may not translate as a unit in lexical translation



$\mathbf{a} = ???$ $\mathbf{a} = (1, 2, (3, 4)^\top)^\top$?

Computing Word Alignments

- Word alignments are the basis for most translation algorithms
- Given two sentences F and E, find a good alignment
- Can learn to align from supervised word alignments, but human-aligned bitexts are rare and expensive to construct.
- Typically use an unsupervised EM-based approach to compute a word alignment from unannotated parallel corpus.

IBM Model 1

- First model proposed in seminal paper by Brown *et al*. in 1993 as part of CANDIDE, the first complete SMT system.
- Generative model: break up translation process into smaller steps.
- Simple lexical translation model
- Additional assumptions
  - All alignment decisions are independent
  - The alignment distribution for each a_i is uniform over all source words and NULL

Lexical Translation

- Goal: A model $p(e|f,m)$

$$e = \langle e_1, e_2, \ldots, e_m \rangle$$
$$f = \langle f_1, f_2, \ldots, f_n \rangle$$

  - Assumptions
    - Each word e_i in e is generated from exactly one word in f
    - Thus, we have an alignment a_i that indicates which word e_i came from. f_{a_i}
    - Given the alignments a, translation decisions are conditionally independent of each other and depend only on the aligned source word f_{a_i}

Lexical Translation

$$p(e|f,m) = \sum_{a \in [0,n]^m} p(a|f,m) \times \prod_{i=1}^{m} p(e_i|f_{a_i})$$

Alignment × Translation | Alignment

IBM Model 1: P(E|F)

- Translation probability
  - For a foreign sentence $f = \left(f_1, \ldots, f_{l_f}\right)$ of length l_f
  - To an English sentence e $= \left(e_1, \ldots, e_{l_e}\right)$ of length l_e
  - With an alignment of each English word e_j to a foreign word f_i according to the alignment function $a: j \rightarrow i$

$$p(e, a|f) = \frac{\epsilon}{\left(l_f + 1\right)^{l_e}} \prod_{j=1}^{l_e} t\left(e_j | f_{a(j)}\right)$$