Statistical Machine Translation

Wei Xu

(many slides from Greg Durrett, Yoav Artzi)

Semester So-far

- Machine Learning Models
  - Linear models: Naive Bayes, Logistic Regression, SVM, Perceptron
  - Neural models: FeedForward Neural Networks, Back-prop, …

- Sequence Models (NER, POS tagging, etc)
  - Hidden Markov Model, Viterbi Algorithm, Conditional Random Fields

- Word Embeddings

- Recurrent NN, Convolutional NN, Neural CRF

Rest of the Semester

- Applications in Natural Language Processing
  - Machine Translation
  - Information Extraction
  - Reading Comprehension
  - Automatic Summarization (if time)
  - Dialog System
  - Speech Recognization
  - etc.

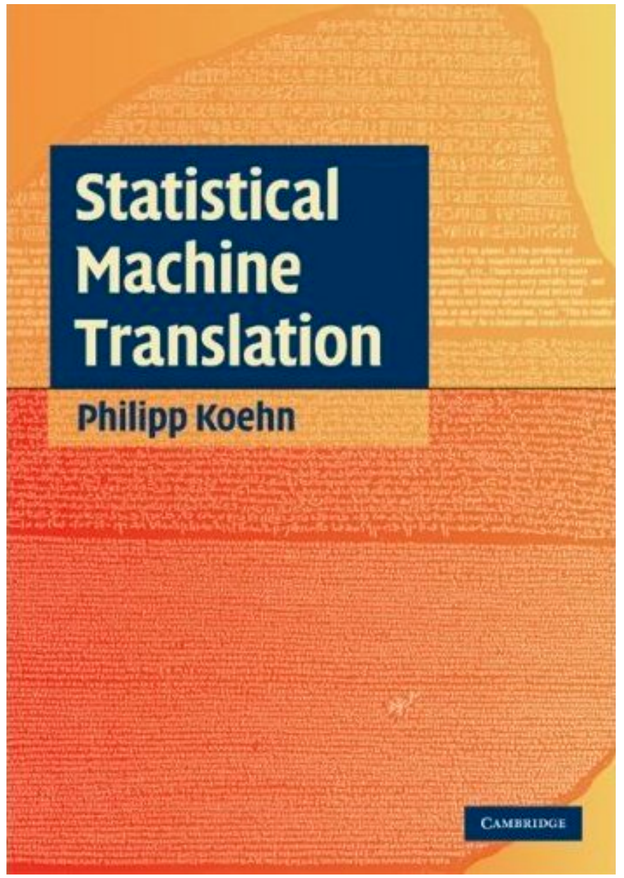
- Some new advances in Natural Language Processing
  - Transformer model (June 2017 —)
  - Contextual Word Embeddings (2018 —)

This Lecture

- MT and evaluation
- Word alignment
- Language models
- Phrase-based decoders
- Syntax-based decoders

Administrivia

- Reading — Eisenstein 18.1, 18.2
- Project 2 due 3/11
- Additional Reading — http://mt-class.org/jhu/
- Midterm (take home)
  - Word2vec, LSTM, machine translation, Transformer …



MT Basics

MT Basics

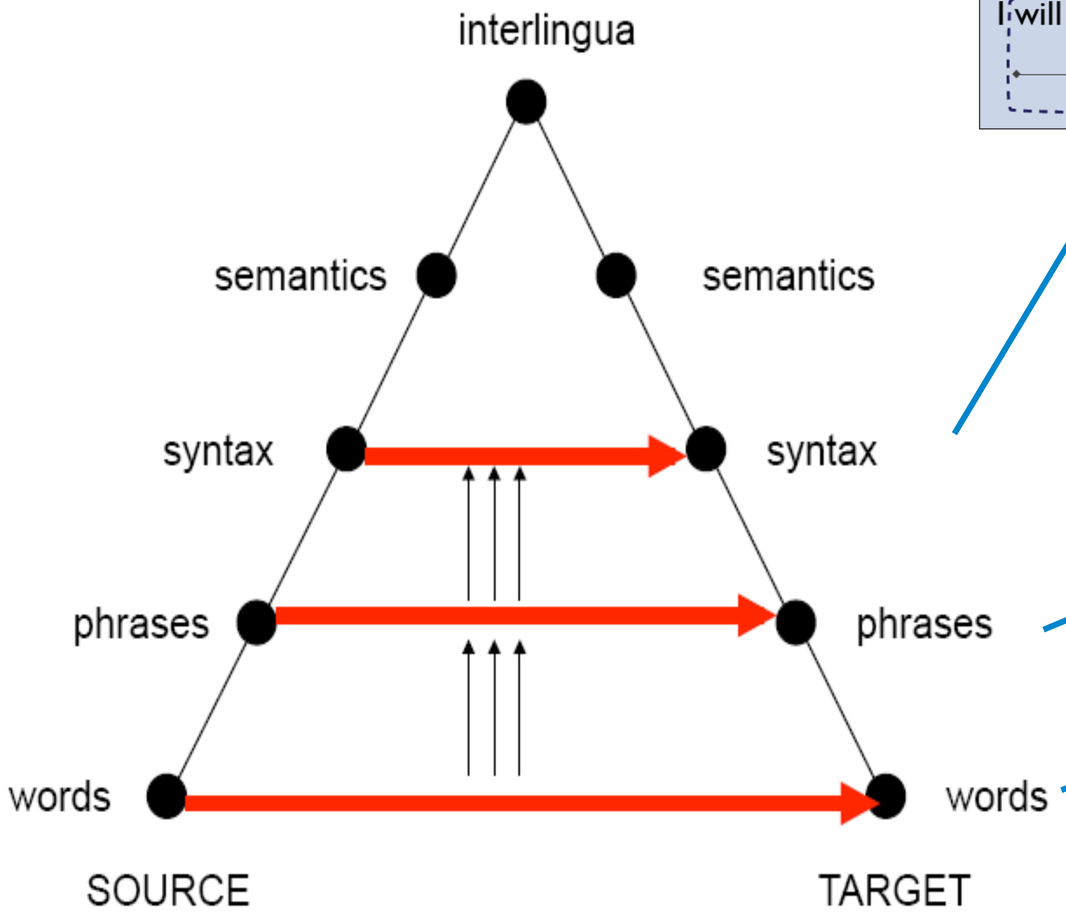
People's Daily, August 30, 2017

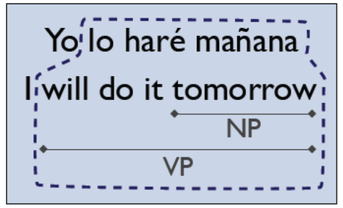
Trump Pope family watch a hundred years a year in the White House balcony

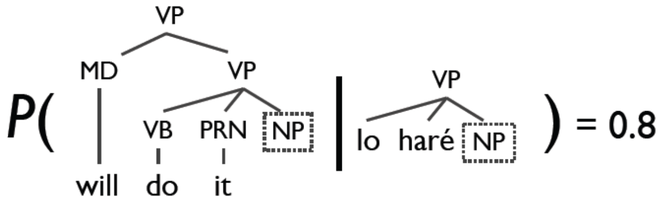
MT Ideally

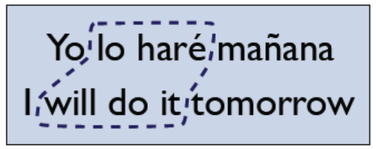
- I have a friend => `∃x friend(x,self)` => J'ai un ami
  J'ai une amie
  - May need information you didn't think about in your representation
  - Hard for semantic representations to cover everything
- Everyone has a friend => `∃x∀y friend(x,y)` / `∀x∃y friend(x,y)` => Tout le monde a un ami
  - Can often get away without doing all disambiguation — same ambiguities may exist in both languages

Levels of Transfer: Vauquois Triangle

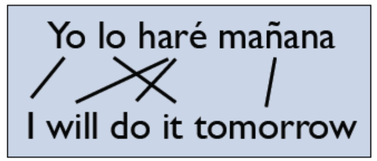








English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2

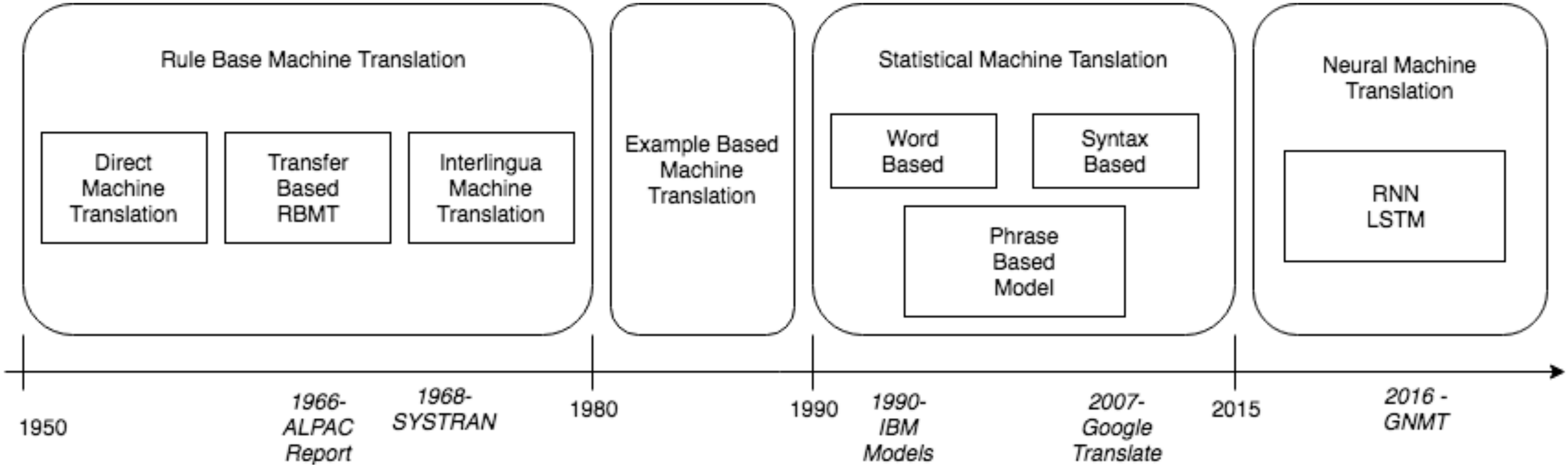


English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

- Today: mostly phrase-based, some syntax

Slide credit: Dan Klein

History of MT



Parallel Training Corpus

		facing with the swelling flow of through traffic zooming past their doors .		retahila de inconvenientes que más y más gente tiene que soportar por el tráfico que pasa por delante_de sus casas , que aumenta a_diario .
5	#77501757	Weekend traffic bans and traffic **jams** are a curse to road transport .	#74765580	Las prohibiciones de conducir los fines de semana y los embotellamientos asolan el transporte por carretera .
6	#79500725	Some people also want to recoup the cost of traffic **jams** from those who get stuck in them , according to the ' polluter pays ' principle .	#76764676	Algunos son partidarios de que incluso los costes ocasionados por los atascos se carguen a el ciudadano que se encuentra atrapado en ellos , de conformidad con el principio de que " quien contamina paga " .
7	#79500765	I think this is an excellent principle and I would like to see it applied in full , but not to traffic **jams** .	#76764713	Me parece un principio acertado y estoy dispuesta a aplicarlo íntegramente , pero no sobre los atascos , ya_que éstos son un claro indicio de el fracaso de la política gubernamental en_materia_de infraestructuras .
8	#79500768	Traffic **jams** are indicative of failed government policy on the infrastructure front , which is why the government itself , certainly in the Netherlands , must be regarded as the polluter .	#76764747	Por eso es preciso subrayar que en estos casos quien contamina es el propio Gobierno , a el menos en los Países_Bajos .
9	#81309716	This would increase traffic **jams** , weaken road safety and increase costs .	#78586130	Esto aumentaría los atascos , mermaría la seguridad vial e incrementaría los costes .
10	#81997391	In the previous legislature , Parliament gave its opinion on the Commission ' s proposals on the simplification of vertical directives on sugar , honey , fruit juices , milk and **jams** .	#79281114	En efecto , durante la precedente legislatura , el Parlamento se manifestó sobre las propuestas de la Comisión relativas a la simplificación de directivas verticales sobre el azúcar , la miel , los zumos de frutas , la leche y las confituras .
11	#81998167	For **jams** , I personally reintroduced an amendment that was not accepted by the Committee on the Environment , Public Health and Consumer Policy , but which I hold to .	#79281936	Para las confituras , yo personalmente volví a introducir una enmienda que no fue aceptada por la Comisión_de_Medio_Ambiente , Salud_Pública y Política_de_el_Consumidor , pero que es importante para mí .
12	#81998209	It concerns not accepting the general use of a chemical flavouring in **jams** and marmalades , that is vanillin .	#79281966	Se trata de no aceptar la utilización generalizada de un aroma químico en las confituras y " marmalades " , a saber , la vainillina .
13	#82800065	This is highlighted particularly in towns where it is necessary to find ways of solving environmental problems and the difficulties caused by traffic **jams** .	#80085988	Esto se pone_de_relieve aún más en las ciudades , en las que hay que encontrar medios para eliminar los inconvenientes derivados de los problemas medioambientales y de la congestión de el tráfico .

Evaluating MT

- Fluency: does it sound good in the target language?
- Fidelity/adequacy: does it capture the meaning of the original?
- BLEU score: geometric mean of 1-, 2-, 3-, and 4-gram precision vs. a reference, multiplied by brevity penalty

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

		1-gram	2-gram	3-gram
hypothesis 1	I am exhausted	3/3	1/2	0/1
hypothesis 2	Tired is I	1/3	0/2	0/1
hypothesis 3	I I I	1/3	0/2	0/1
reference 1	I am tired			
reference 2	I am ready to sleep now and so exhausted			

Evaluating MT

- Fluency: does it sound good in the target language?
- Fidelity/adequacy: does it capture the meaning of the original?
- BLEU score: geometric mean of 1-, 2-, 3-, and 4-gram precision vs. a reference, multiplied by brevity penalty

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

- Typically $n = 4$, $w_i = 1/4$

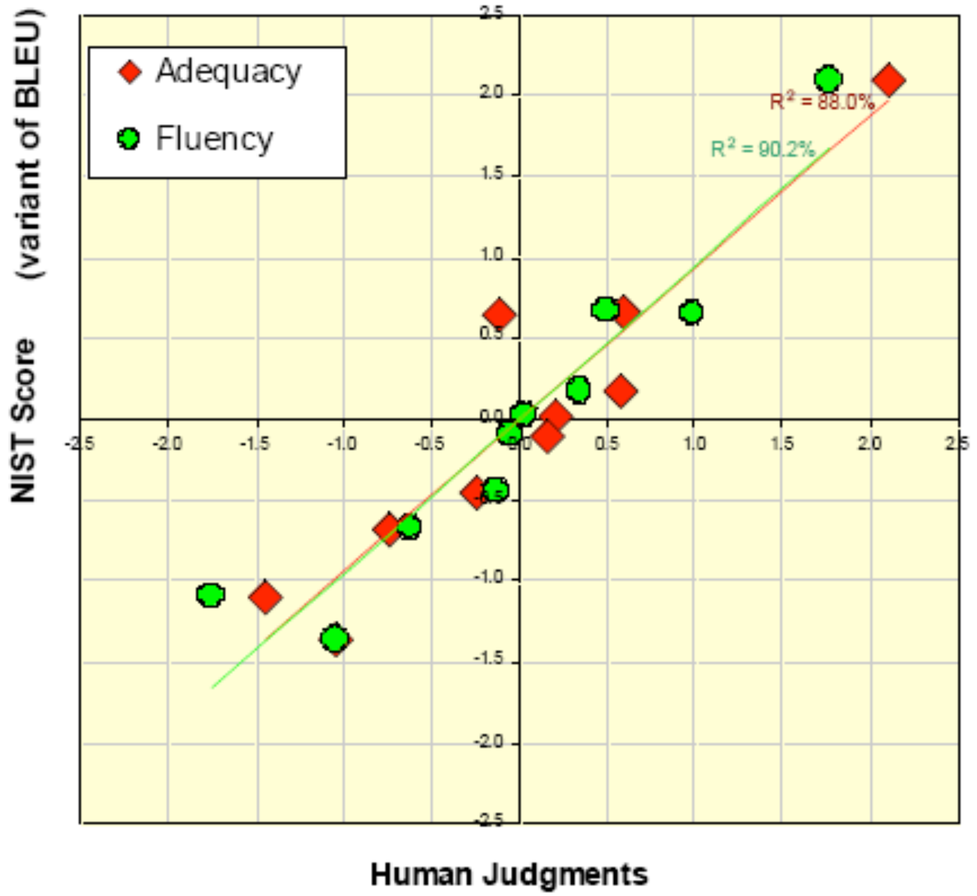
$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

- r = length of reference
  c = length of system output

- Does this capture fluency and adequacy?

BLEU Score

- Better methods with human-in-the-loop
- If you're building real MT systems, you do user studies. In academia, you mostly use BLEU.



slide from G. Doddington (NIST)

Appraise - Human Evaluation Interface

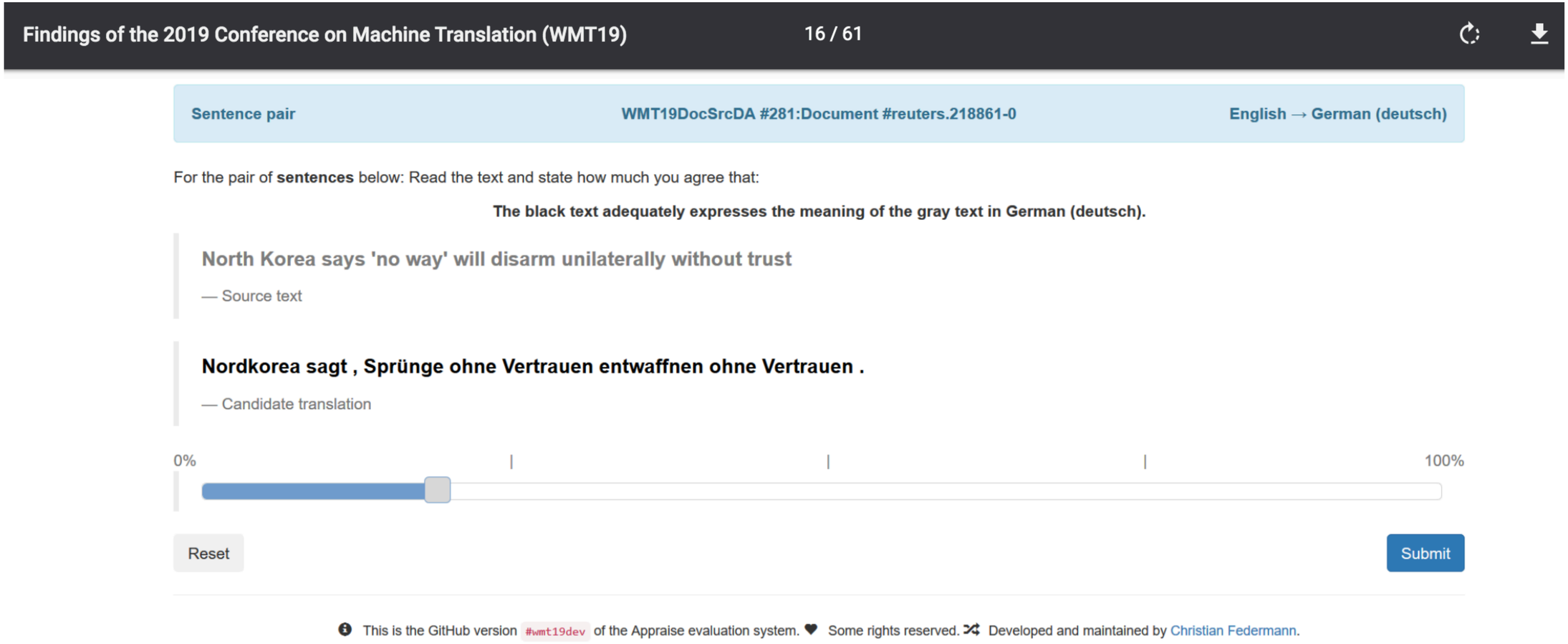


Figure 3: Screen shot of segment-rating portion of document-level direct assessment in the Appraise interface for an example English to German assessment from the human evaluation campaign. The annotator is presented with the machine translation output segment randomly selected from competing systems (anonymized) and is asked to rate the translation on a sliding scale.

https://www.aclweb.org/anthology/W19-5301.pdf

Other MT Evaluation Metrics

- BLEU (2002): n-gram overlap
- METEOR (2005): also take into consideration of synonyms
- HTER (2009): human-assisted translation error rate
- BERTScore (2019): embedding-based
- BLEURT (2020) and COMET (2020): trained neural network model using human evaluation data
- and many more …

Phrase-Based MT

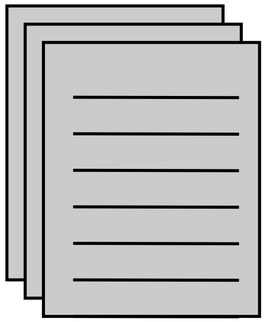
- Key idea: translation works better the bigger chunks you use
- Remember phrases from training data, translate piece-by-piece and stitch those pieces together to translate
  - How to identify phrases? Word alignment over source-target bitext
  - How to stitch together? Language model over target language
  - Decoder takes phrases and a language model and searches over possible translations
- NOT like standard discriminative models (take a bunch of translation pairs, learn a ton of parameters in an end-to-end way)

Phrase-Based MT

- Goal: translate from Foreign language to English

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 $P(f|e)$



Unlabeled English data

Language model $P(e)$

$$P(e|f) \propto P(f|e)P(e)$$

Noisy channel model: combine scores from translation model + language model to translate foreign to English

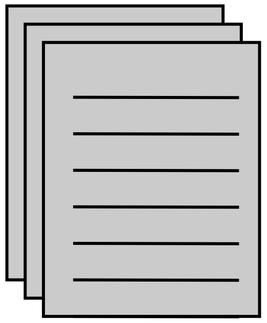
"Translate faithfully but make fluent English"

Phrase-Based MT

- Goal: translate from Foreign language to English

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 $P(f|e)$



Unlabeled English data

Language model $P(e)$

$$P(e|f) \propto P(f|e)P(e)$$

Noisy channel model: combine scores from translation model + language model to translate foreign to English

"Translate faithfully but make fluent English"

Word Alignment

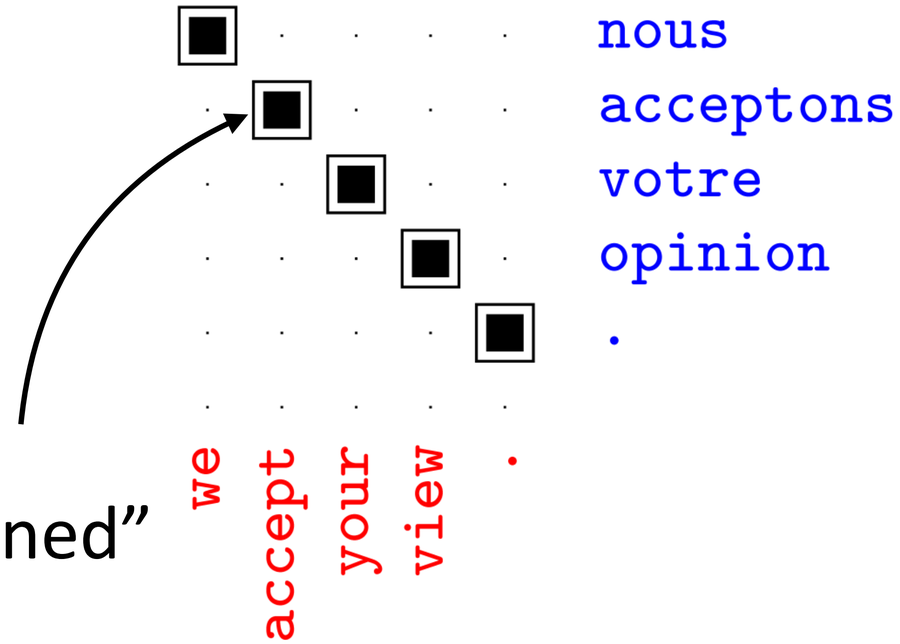
Word Alignment

- Input: a bitext corpus, pairs of translated sentences

  nous acceptons votre opinion . ||| we accept your view

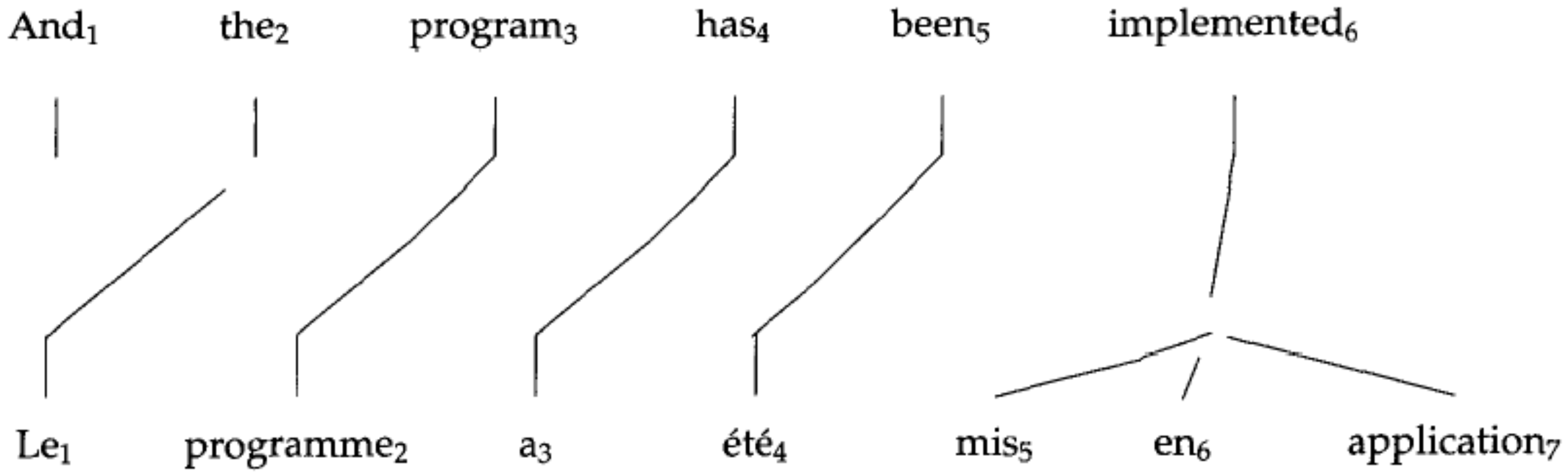
  nous allons changer d'avis ||| we are going to change our minds

- Output: alignments between words in each sentence
  - We will see how to turn these into phrases



"accept and acceptons are aligned"

1-to-Many Alignments



- An alignment $\mathbf{a}$ identifies which English word each French word originated from.

Word Alignment

- Models P(**f**|**e**): probability of “Foreign” sentence being generated from “English” sentence according to a model

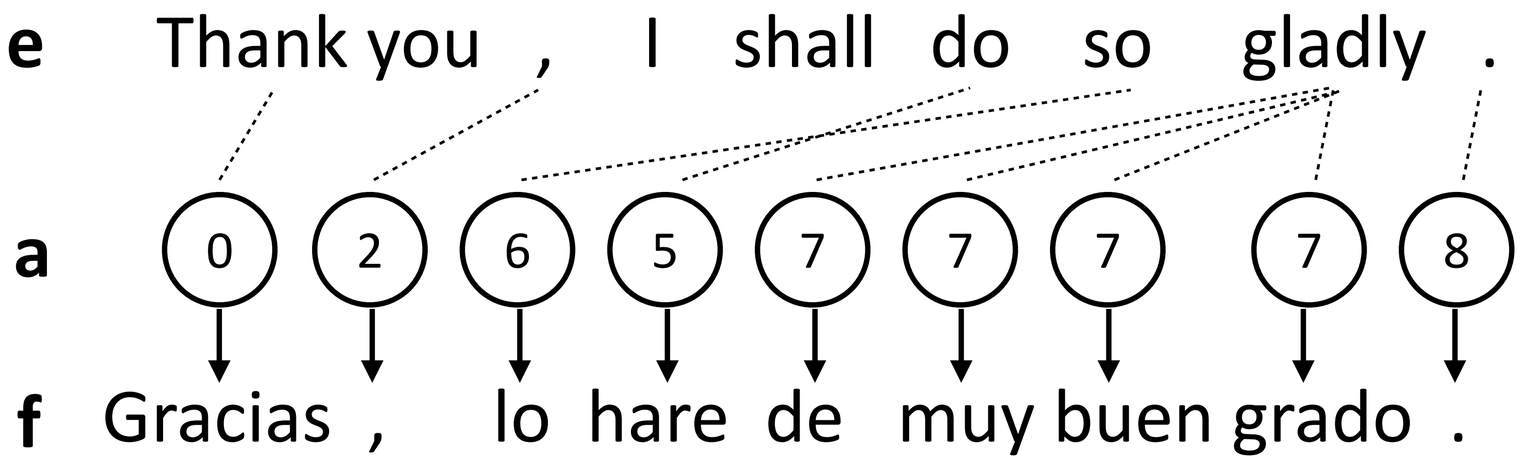
- Latent variable model: $P(\mathbf{f}|\mathbf{e}) = \sum_{\mathbf{a}} P(\mathbf{f}, \mathbf{a}|\mathbf{e}) = \sum_{\mathbf{a}} P(\mathbf{f}|\mathbf{a}, \mathbf{e}) P(\mathbf{a})$

sum over all possible alignments

IBM Model 1

- Each “Foreign” word is aligned to *at most* one English word

$$P(\mathbf{f}, \mathbf{a}|\mathbf{e}) = \prod_{i=1}^{n} P(f_i|e_{a_i})P(a_i)$$



- Set P(a) uniformly (no prior over good alignments) = 1 / (#words in e)
- $P(f_i|e_{a_i})$: word translation probability. Learn with EM (Eisenstein ch 18.2.2)

Brown et al. (1993)

IBM Model 1

- Sketch of the expectation-maximization (EM) algorithm for Model 1

```
# Accumulation (over corpus)
For each sentence pair
  For each source position j
    Sum = 0.0
    For each target position i
      Sum += p(f_j|e_i)
    For each target position i
      Count(f_j,e_i) += p(f_j|e_i)/Sum


# Re-estimate probabilities (over count table)
For each target word e
  Sum = 0.0
  For each source word f
    Sum += Count(f,e)
  For each source word f
    p(f|e) = Count(f,e)/Sum

# Repeat for several iterations
```

Slide Credit: Sanjika Hewavitharana

IBM Model 1

p(fle)	And	the	program	has	been	implemented
Le	0.2	0.6	0.1	0.025	0.05	0.025
programme	0.05	0.2	0.45	0.1	0.1	0.1
a	0.1	0.1	0.15	0.2	0.15	0.3
ete	0.05	0.05	0.05	0.05	0.7	0.1
mis	0.2	0.05	0.05	0.05	0.25	0.4
en	0.25	0.1	0.25	0.25	0.1	0.05
application	0.01	0.03	0.01	0.02	0.03	0.9

- $P(f_i|e_{a_i})$: word translation probability table

Slide Credit: Yoav Artiz

IBM Models

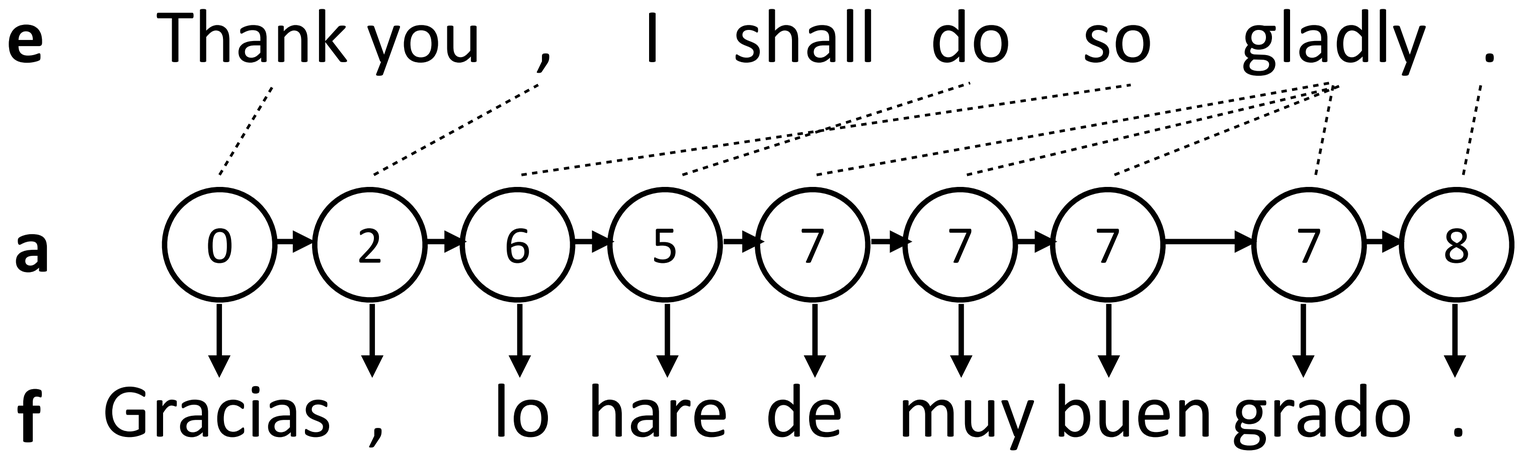
- IBM1 – lexical probabilities only
- IBM2 – lexicon plus absolut position
- IBM3 – plus fertilities
- IBM4 – inverted relative position alignment
- IBM5 – non-deficient version of model 4
- HMM – lexicon plus relative position
- BiBr – Bilingual Bracketing, lexical probabilites plus reordering via parallel segmentation
- Syntactical alignment models

[Brown et.al. 1993, Vogel et.al. 1996, Och et al 1999, Wu 1997, Yamada et al. 2003]

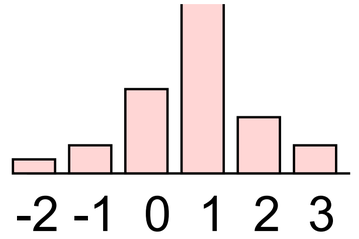
HMM for Alignment

- Sequential dependence between a's to capture monotonicity

$$P(\mathbf{f}, \mathbf{a}|\mathbf{e}) = \prod_{i=1}^{n} P(f_i|e_{a_i})P(a_i|a_{i-1})$$



- Alignment dist parameterized by jump size: $P(a_j - a_{j-1})$

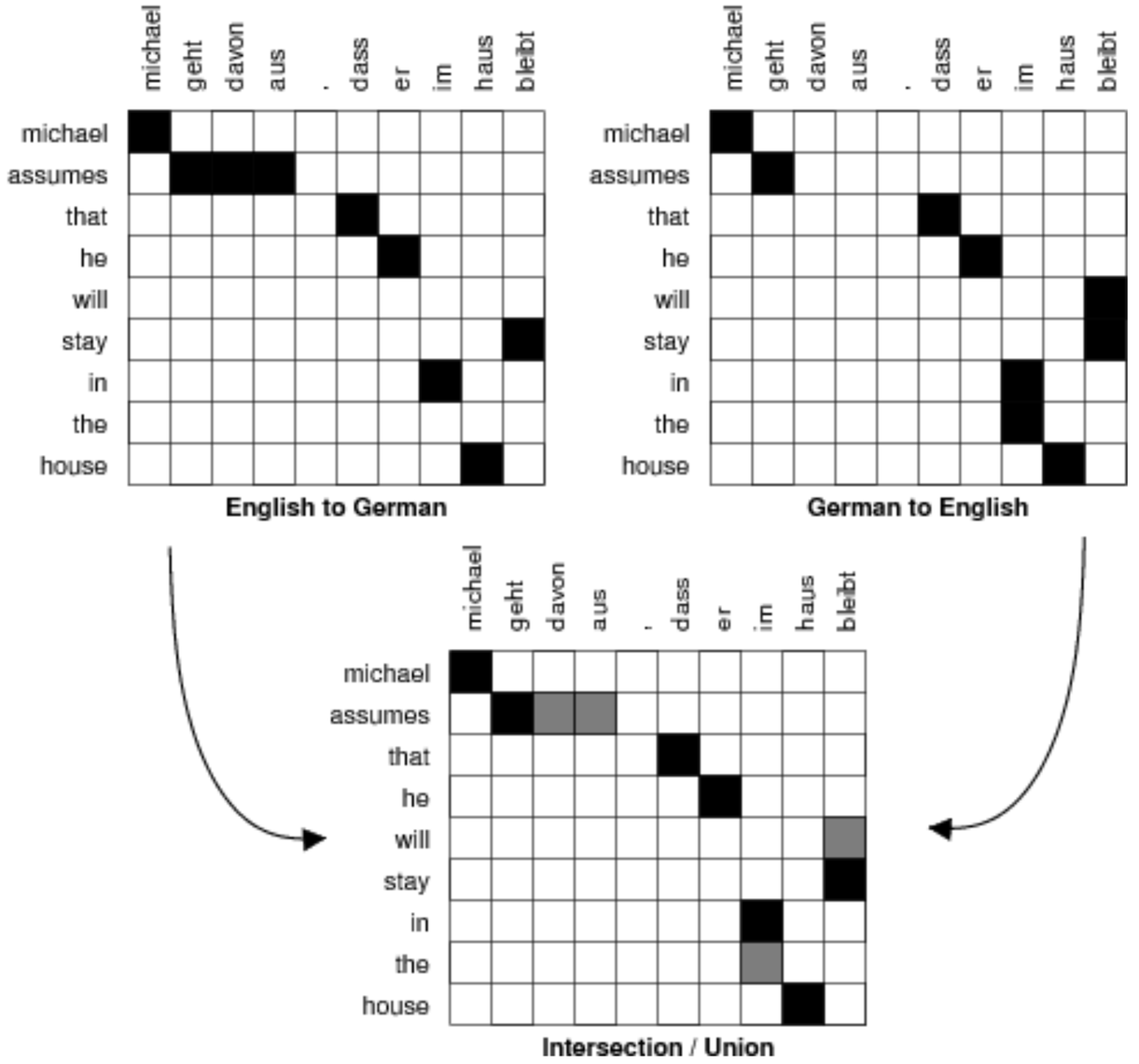


- $P(f_i|e_{a_i})$: same as before

Brown et al. (1993)

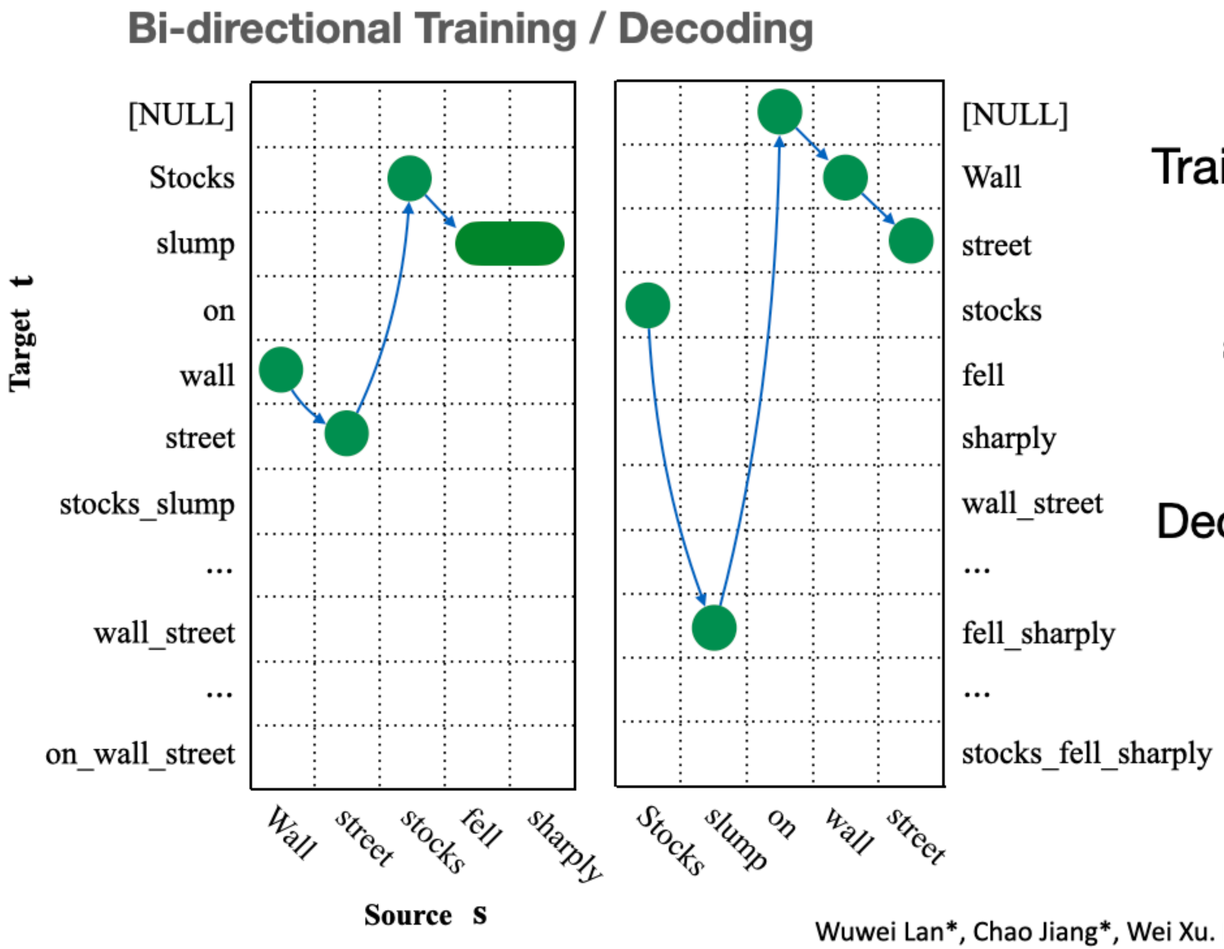
HMM Model

- Run in both directions - Q: Why?
- Alignments are generally monotonic (along diagonal)
- Some mistakes, especially when you have rare words (*garbage collection*)
- GIZA++ Toolkit (Och & Ney, 2003)



Supervised Methods also Exist

Semi-CRF Word Alignment Model



Training objective:

$$\sum_{\mathbf{s},\mathbf{t},\mathbf{a}} -logP(\mathbf{a}_{s2t}\,|\,\mathbf{s},\mathbf{t}) - logP(\mathbf{a}_{t2s}\,|\,\mathbf{t},\mathbf{s})$$

Source-to-target **Target-to-source**

Decoding:

Viterbi-like Algorithm + Intersect + Expand

Wuwei Lan*, Chao Jiang*, Wei Xu. Neural Semi-Markov CRF for Monolingual Word Alignment (ACL 2021)

Phrase Extraction

- Find contiguous sets of aligned words in the two languages that don't have alignments to other words

de assister à la runion et ||| to attend the meeting and

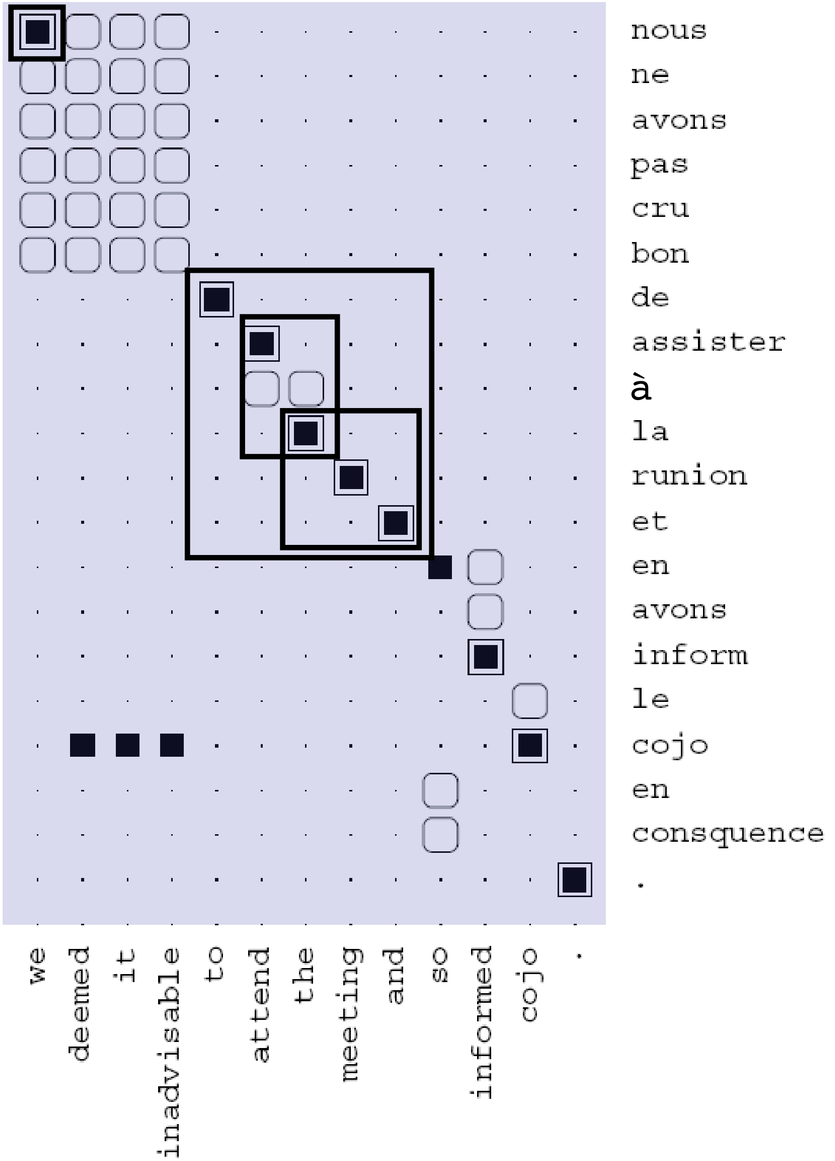
assister à la runion ||| attend the meeting

la runion and ||| the meeting and

nous ||| we

…

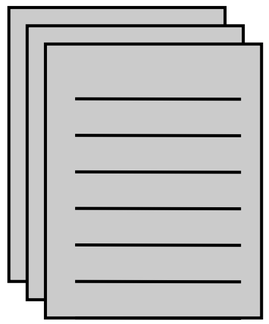
- Lots of phrases possible, count across all sentences and score by frequency



Phrase-Based MT

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 P(f|e)



Language model P(e)

Unlabeled English data

$$P(e|f) \propto P(f|e)P(e)$$

Noisy channel model: combine scores from translation model + language model to translate foreign to English

"Translate faithfully but make fluent English"

Language Modeling

N-gram Language Models

I visited San _____ put a distribution over the next word

- Simple generative model: distribution of next word is a multinomial distribution conditioned on previous n-1 words

$$P(x|\text{visited San}) = \frac{\text{count}(\text{visited San}, x)}{\text{count}(\text{visited San})}$$

Maximum likelihood estimate of this probability from a corpus

- Just relies on counts, even in 2008 (Google n-gram corpus) could scale up to 1.3M word types, 4B n-grams (all 5-grams occurring >40 times on the Web)

Smoothing N-gram Language Models

I visited San _____ put a distribution over the next word!

- Smoothing is very important, particularly when using 4+ gram models

$$P(x|\text{visited San}) = (1-\lambda)\frac{\text{count}(\text{visited San}, x)}{\text{count}(\text{visited San})} + \lambda\frac{\text{count}(\text{San}, x)}{\text{count}(\text{San})}$$

smooth this too!

- One technique is "absolute discounting:" subtract off constant *k* from numerator, set lambda to make this normalize (*k*=1 is like leave-one-out)

$$P(x|\text{visited San}) = \frac{\text{count}(\text{visited San}, x) - k}{\text{count}(\text{visited San})} + \lambda\frac{\text{count}(\text{San}, x)}{\text{count}(\text{San})}$$

- Kneser-Ney smoothing: this trick, plus low-order distributions modified to capture fertilities (how many distinct words appear in a context)

Engineering N-gram Models

- For 5+-gram models, need to store between 100M and 10B context-word-count triples

(a) Context-Encoding

w	c	val
1933	15176585	3
1933	15176587	2
1933	15176593	1
1933	15176613	8
1933	15179801	1
1935	15176585	298
1935	15176589	1

(b) Context Deltas

Δw	Δc	val
1933	15176585	3
+0	+2	1
+0	+5	1
+0	+40	8
+0	+188	1
+2	15176585	298
+0	+4	1

(c) Bits Required

$\lvert\Delta w\rvert$	$\lvert\Delta c\rvert$	$\lvert val\rvert$
24	40	3
2	3	3
2	3	3
2	9	6
2	12	3
4	36	15
2	6	3

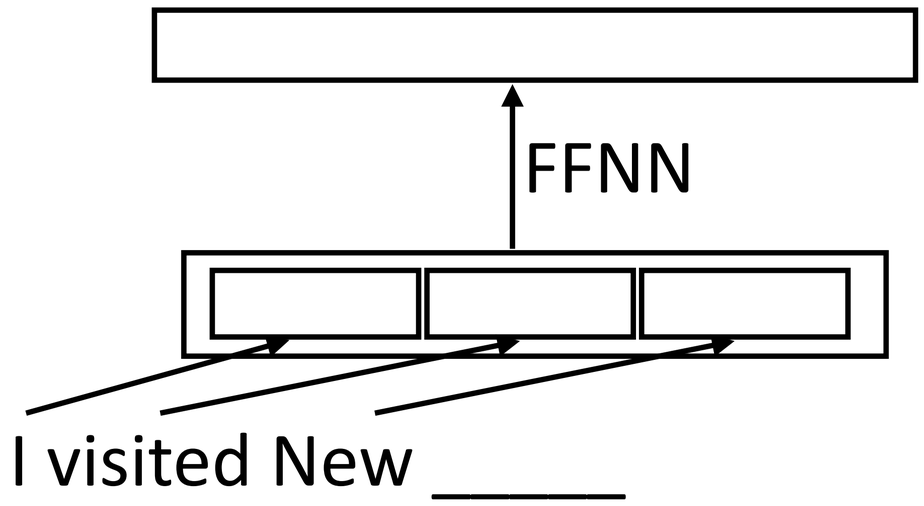
- Make it fit in memory by *delta encoding* scheme: store deltas instead of values and use variable-length encoding

- KenLM and BerkeleyLM toolkit

Heafield (2011), Pauls and Klein (2011)

Neural Language Models

- Early work: feedforward neural networks looking at context

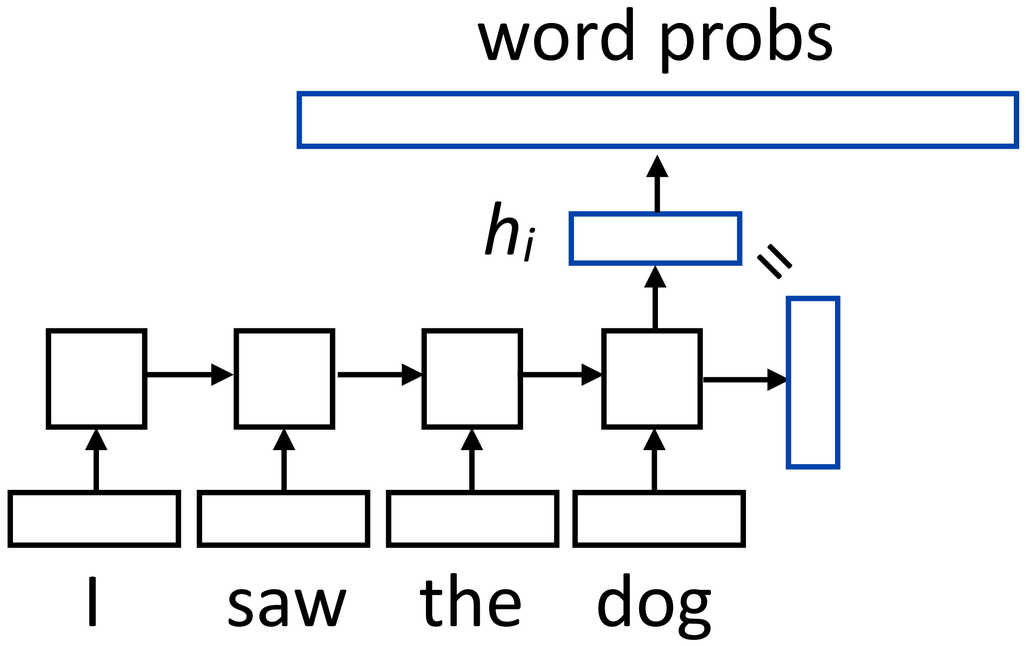


$$P(w_i|w_{i-n},\ldots,w_{i-1})$$

- Slow to train over lots of data!
- Still only look at a fixed window of context … can we use more?

Mnih and Hinton (2003)

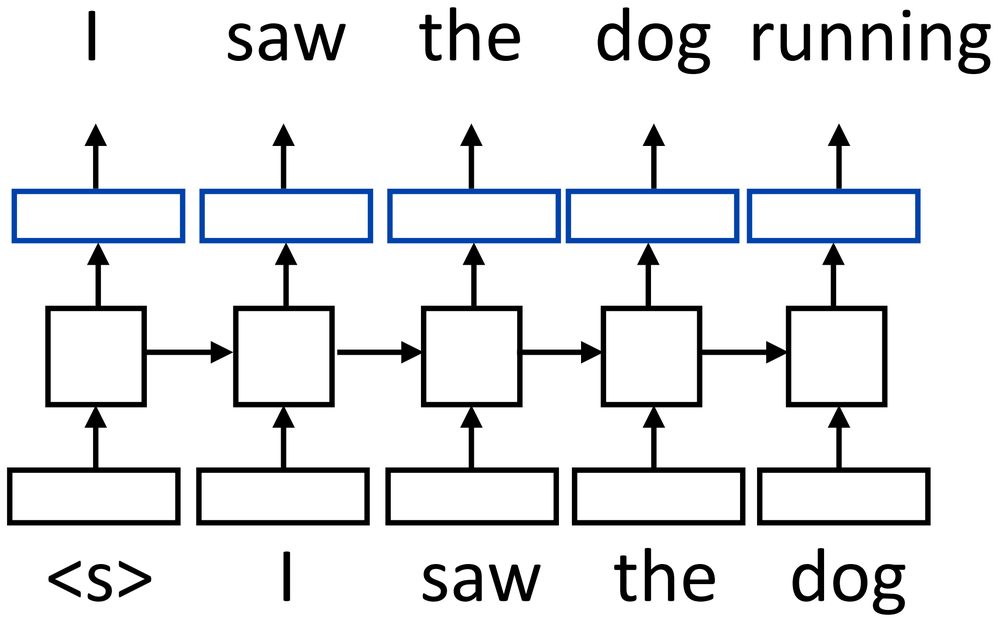
RNN Language Modeling



$$P(w|\text{context}) = \text{softmax}(W\mathbf{h}_i)$$

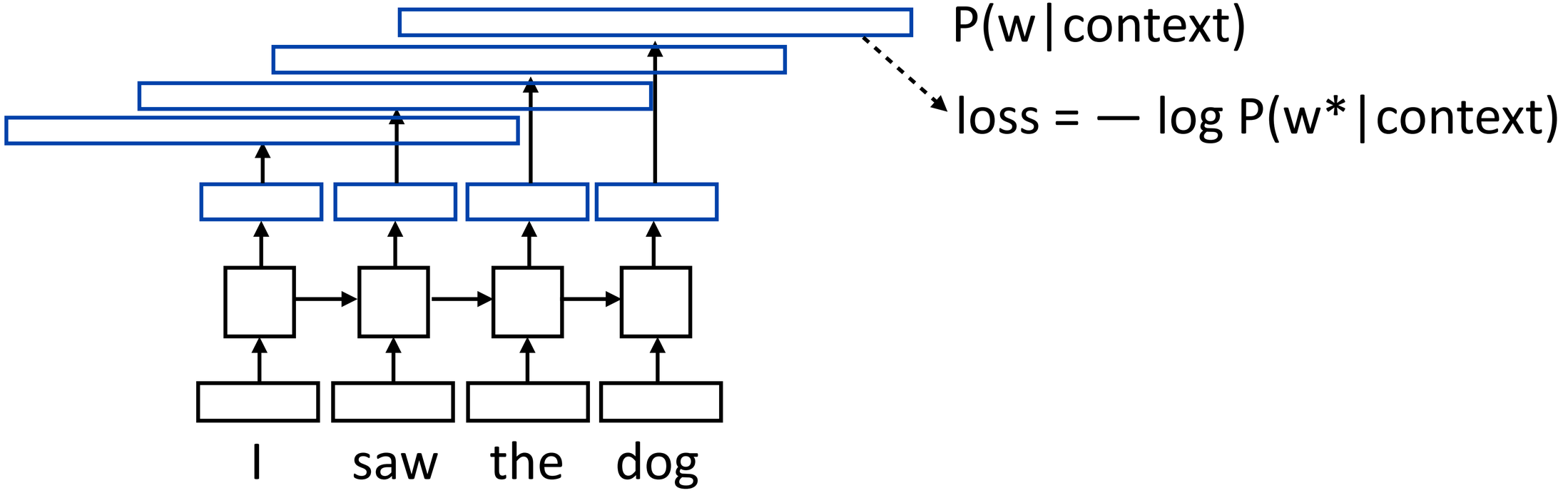
- W is a (vocab size) x (hidden size) matrix

Training RNNLMs



- Input is a sequence of words, output is those words shifted by one,
- Allows us to efficiently batch up training across time (one run of the RNN)

Training RNNLMs



- Total loss = sum of negative log likelihoods at each position
- Backpropagate through the network to simultaneously learn to predict next word given previous words at all positions

LM Evaluation

- Accuracy doesn't make sense — predicting the next word is generally impossible so accuracy values would be very low
- Evaluate LMs on the likelihood of held-out data (averaged to normalize for length)

$$\frac{1}{n}\sum_{i=1}^{n}\log P(w_i|w_1,\ldots,w_{i-1})$$

- Perplexity: exp(average negative log likelihood). Lower is better
  - Suppose we have probs 1/4, 1/3, 1/4, 1/3 for 4 predictions
  - Avg NLL (base e) = 1.242 Perplexity = 3.464

Results

- Evaluate on Penn Treebank: small dataset (1M words) compared to what's used in MT, but common benchmark
- Kneser-Ney 5-gram model with cache: PPL = 125.7
- LSTM: PPL ~ 60-80 (depending on how much you optimize it)
- Melis et al.: many neural LM improvements from 2014-2017 are subsumed by just using the right regularization (right dropout settings). So LSTMs are pretty good

Merity et al. (2017), Melis et al. (2017)

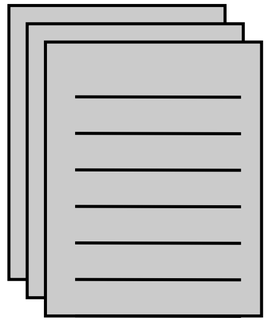
Applications of Language Modeling

- Not limited to machine translations!
- All generation tasks: dialogue, text simplification, paraphrasing, story generation, etc.
- Grammatical error correction
- Pretraining! (more later in the course)
  - Language modeling involves predicting words given context.
  - Learning a neural network to do this induces useful representations for other tasks, similar to word2vec/GloVe.
  - ELMo, BERT, RoBERTa, GPT-2, GPT-3 …

Phrase-Based MT

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 P(f|e)



Language model P(e)

Unlabeled English data

$$P(e|f) \propto P(f|e)P(e)$$

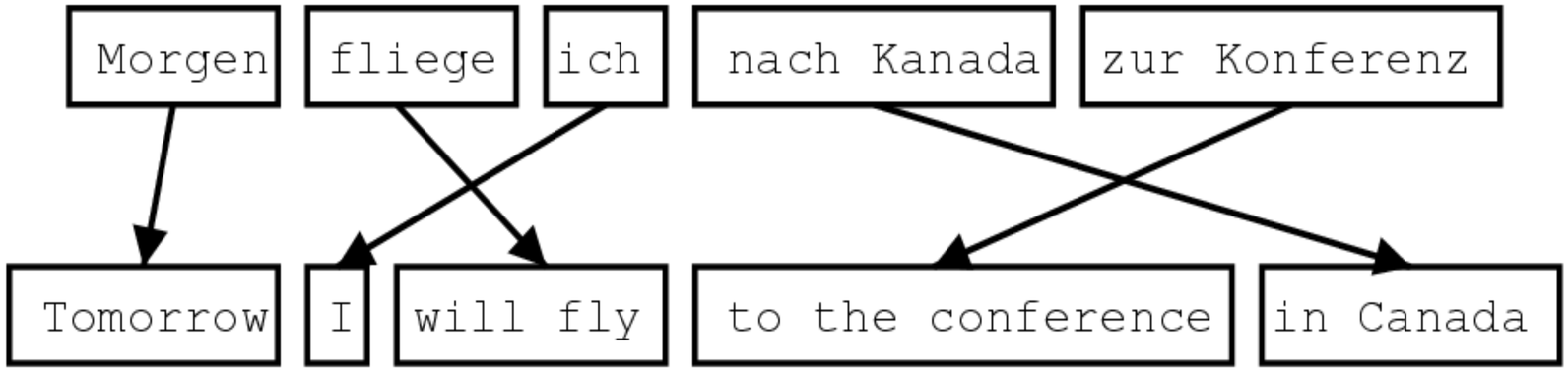
Noisy channel model: combine scores from translation model + language model to translate foreign to English

"Translate faithfully but make fluent English"

Decoding

Phrase-Based Decoding

- Inputs:
  - Language model that scores $P(e_i|e_1,\ldots,e_{i-1}) \approx P(e_i|e_{i-n-1},\ldots,e_{i-1})$
  - Phrase table: set of phrase pairs (**e**, **f**) with probabilities P(**f**|**e**)
- What we want to find: **e** produced by a series of phrase-by-phrase translations from an input **f**, possibly with reordering:



Phrase lattices are big!

这	7人	中包括	来自	法国	和	俄罗斯	的	宇航	员	.
the	7 people	including	by some		**and**	the russian	**the**	the astronauts		,
it	7 people included		by france		and the	the russian		international astronautical	of rapporteur .	
this	7 out	including the	**from**	the french	and the russian		the fifth		.	
these	7 among	including from		the french and		of the russian	of	space	members	.
that	7 persons	including from the		of france	and to	russian	of the	aerospace	members .	
	7 include		from the	of france and		russian		**astronauts**		. the
	7 numbers include		**from france**		and russian		of astronauts who			. ”
	7 populations include		those from france		and russian			astronauts .		
	7 deportees included		come from	**france**	**and russia**		in	astronautical	personnel	;
	7 philtrum	including those from		**france and**		**russia**	a space		**member**	
		including representatives from		france and the		**russia**		astronaut		
		include	came from	**france and russia**			by cosmonauts			
		include representatives from		french	**and russia**			cosmonauts		
		include	came from france		and russia 's			cosmonauts .		
		includes	coming from	french and		russia 's		cosmonaut		
				french and russian			's	astronavigation	member .	
				french	**and russia**		**astronauts**			
					and russia 's				special rapporteur	
					, and	**russia**			rapporteur	
					, and russia				rapporteur .	
					, and russia					
					or	russia 's				

Slide credit: Dan Klein

Phrase-Based Decoding

- Input
- Translations

lo haré | rápidamente | .

I'll do it | quickly | .

quickly | I'll do it | .

The decoder...

tries different segmentations,

translates phrase by phrase,

and considers reorderings.

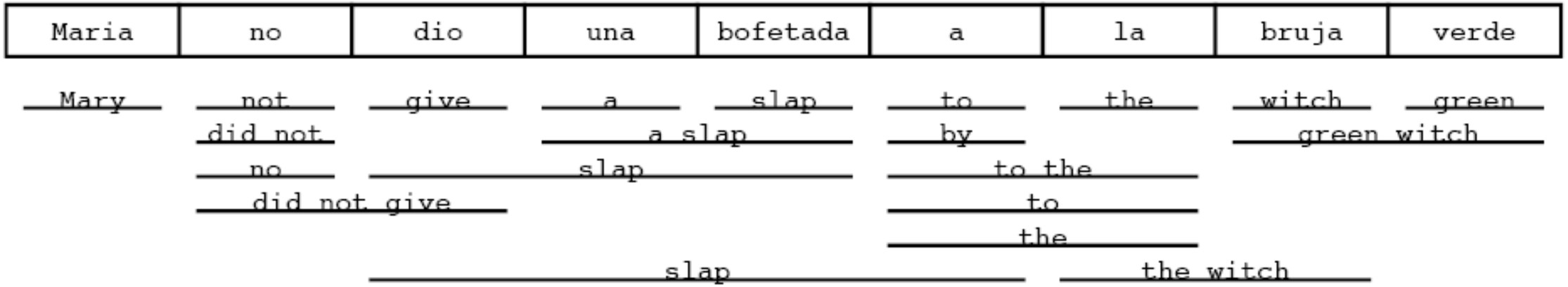
$$\arg\max_{\mathbf{e}} [P(\mathbf{f}|\mathbf{e}) \cdot P(\mathbf{e})]$$

- Decoding objective (for 3-gram LM)

$$\arg\max_{\mathbf{e}} \left[\prod_{\langle \bar{e}, \bar{f} \rangle} P(\bar{f}|\bar{e}) \cdot \prod_{i=1}^{|\mathbf{e}|} P(e_i|e_{i-1}, e_{i-2}) \right]$$

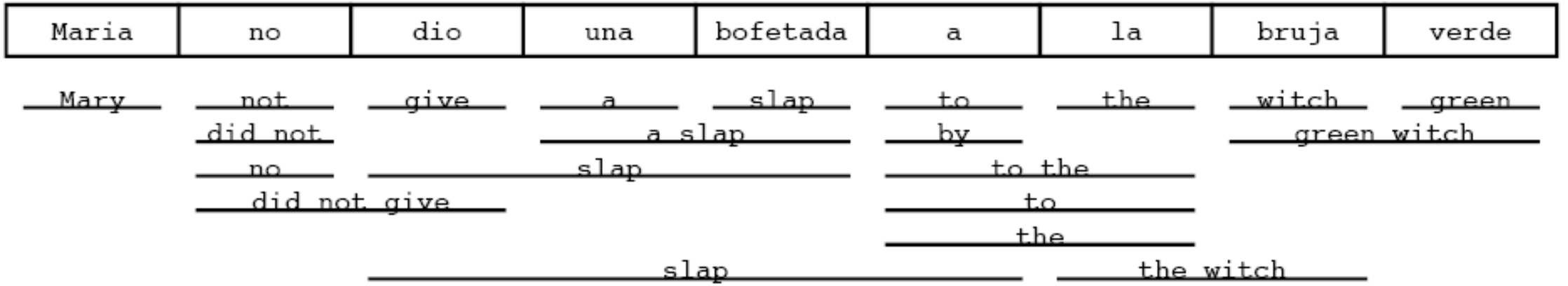
Slide credit: Dan Klein

Beam Search



- An efficient search algorithm which aims to find high-probability sequences (not necessarily the optimal sequence, though)
- Core idea:
  - On each step of decoder, keep track of the *k* most probable partial sequences (which are called hypotheses).
  - *k* is the beam size.

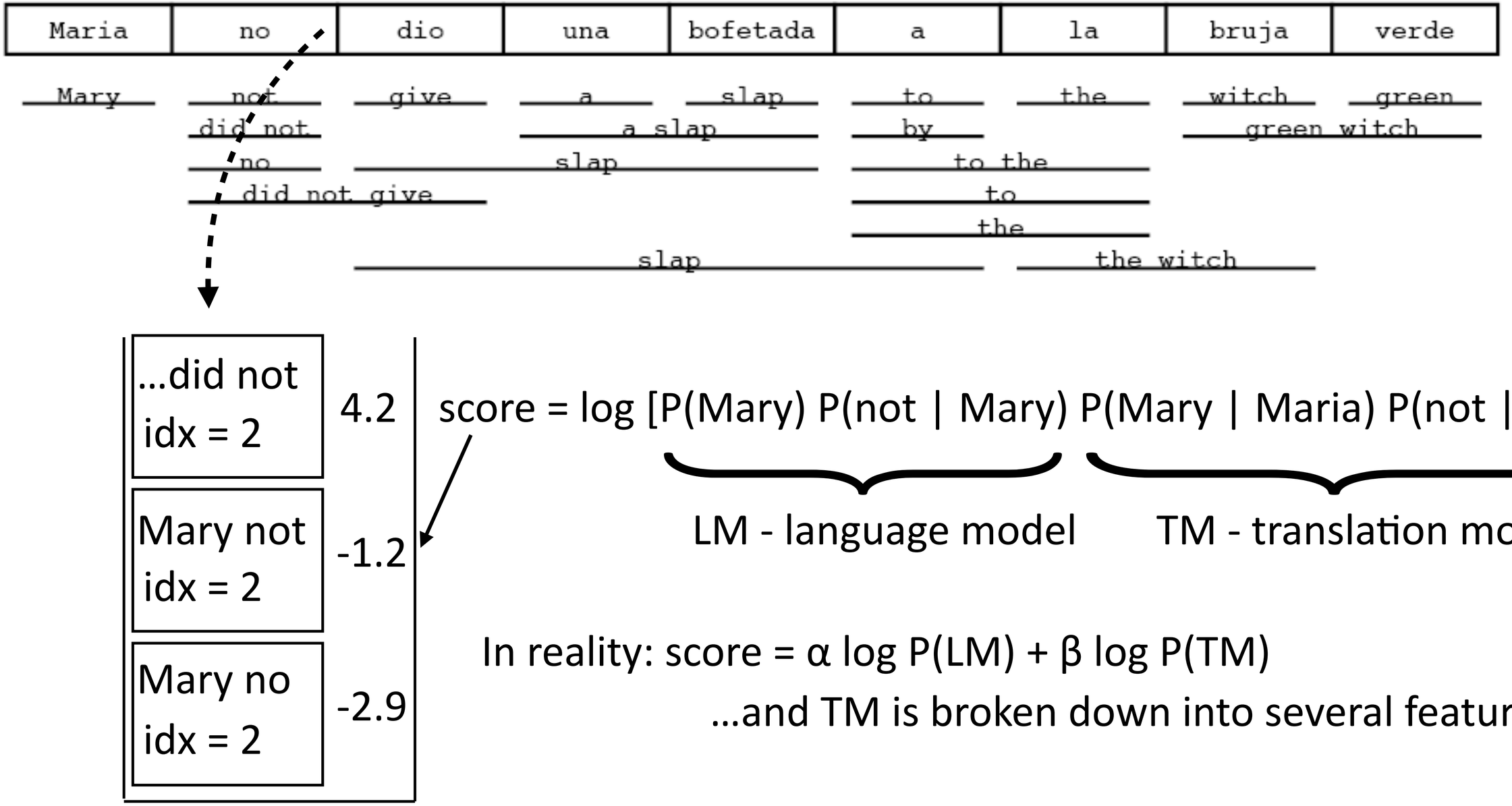
Monotonic Translation



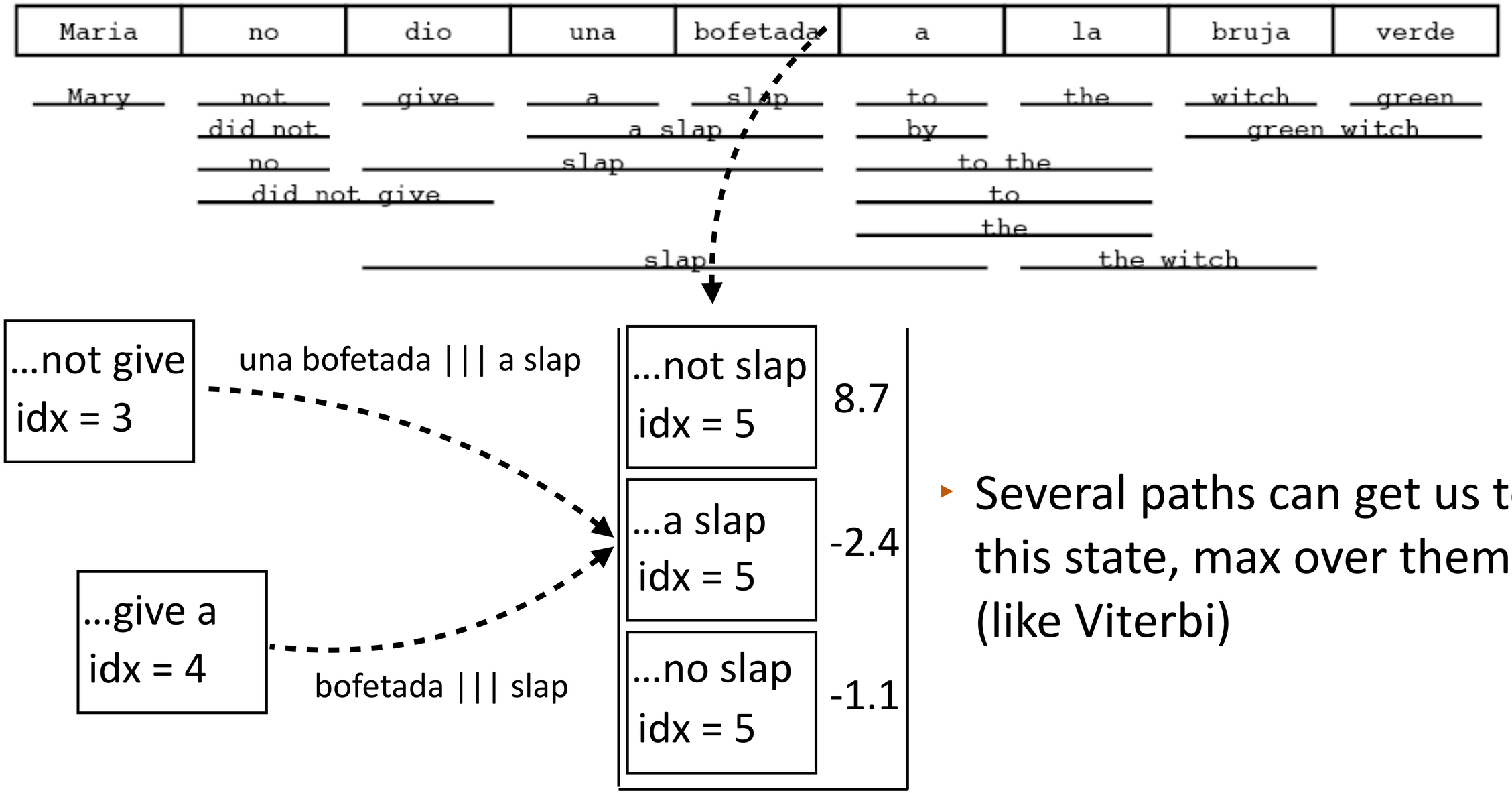
- If we translate with beam search, what state do we need to keep in the beam?
  - What have we translated so far?
  - What words have we produced so far?
  - When using a 3-gram LM, only need to remember the last 2 words!

$$\arg\max_{\mathbf{e}} \left[\prod_{\langle \bar{e}, \bar{f} \rangle} P(\bar{f}|\bar{e}) \cdot \prod_{i=1}^{|\mathbf{e}|} P(e_i|e_{i-1}, e_{i-2}) \right]$$

Monotonic Translation

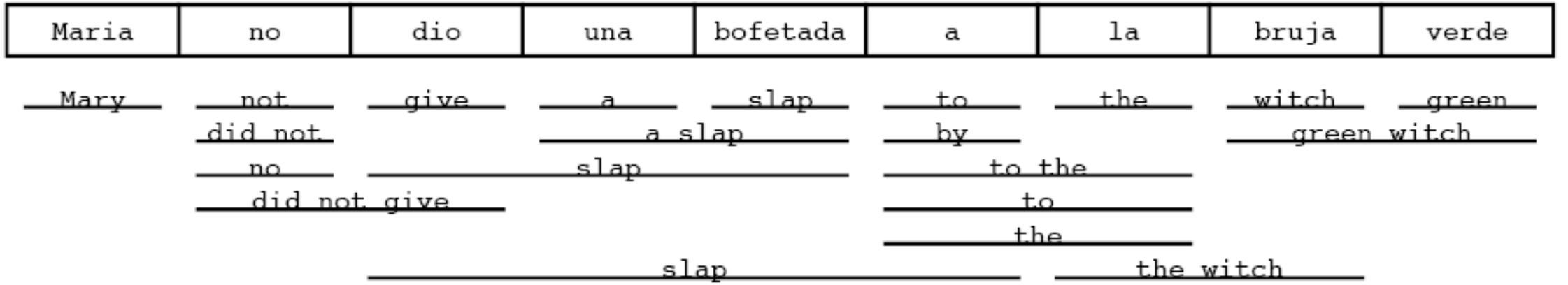


Monotonic Translation

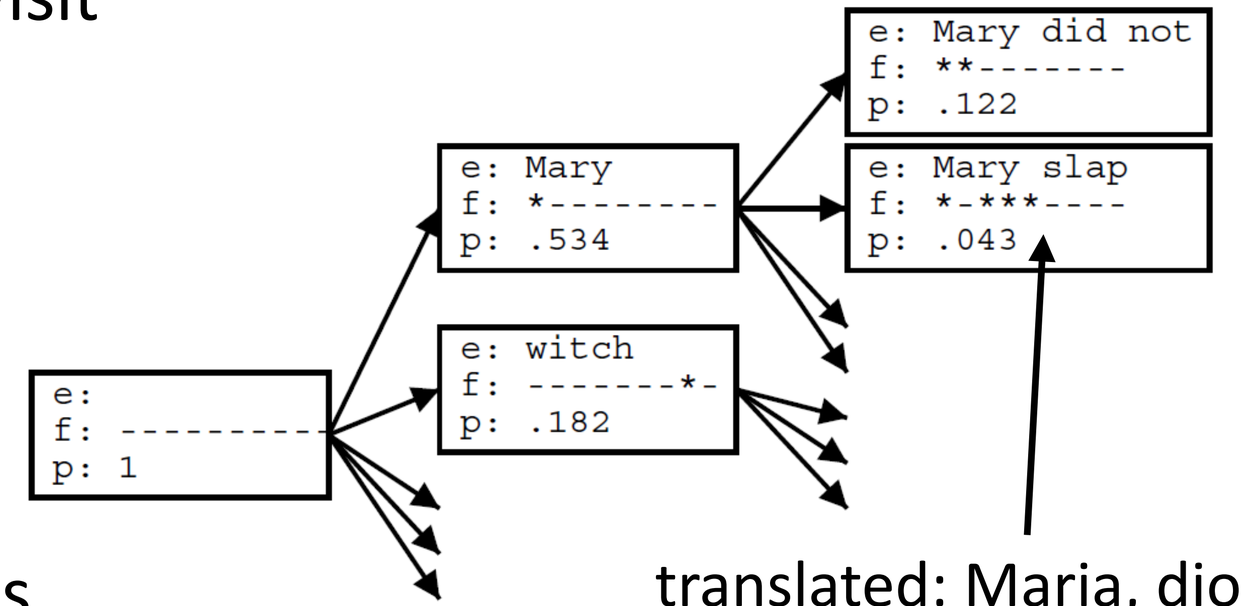


- Several paths can get us to this state, max over them (like Viterbi)

Non-Monotonic Translation



- Non-monotonic translation: can visit source sentence “out of order”
- State needs to describe which words have been translated and which haven’t
- Big enough phrases already capture lots of reorderings, so this isn’t as important as you think



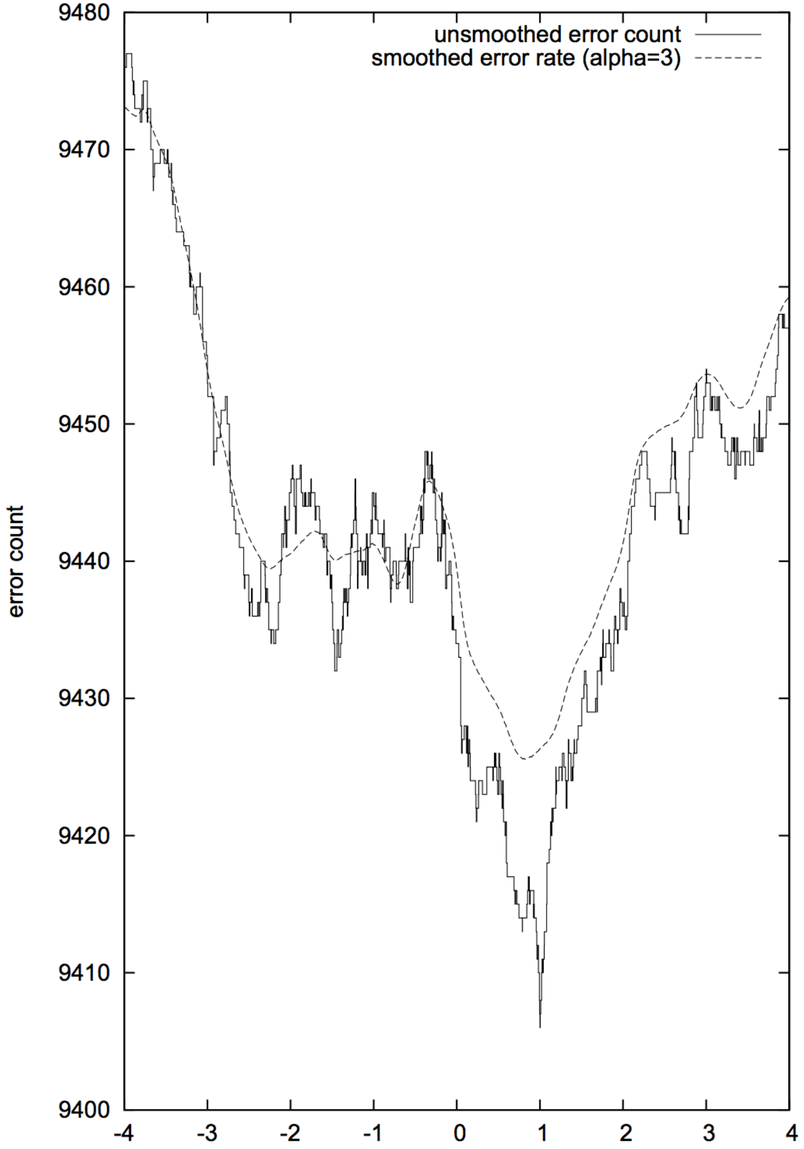
translated: Maria, dio, una, bofetada

Training Decoders

score = α log P(LM) + β log P(TM)

…and TM is broken down into several feature

- Usually 5-20 feature weights to set, want to optimize for BLEU score which is not differentiable
- MERT (Och 2003): decode to get 1000-best translations for each sentence in a small training set (<1000 sentences), do line search on parameters to directly optimize for BLEU



Moses

- Toolkit for machine translation due to Philipp Koehn + Hieu Hoang
  - Pharaoh (Koehn, 2004) is the decoder from Koehn's thesis
- Moses implements word alignment, language models, and this decoder, plus *a ton* more stuff
  - Highly optimized and heavily engineered, could more or less build SOTA translation systems with this from 2007-2013

http://www.statmt.org/moses/

Syntax

Syntactic MT

- Rather than use phrases, use a *synchronous context-free grammar*

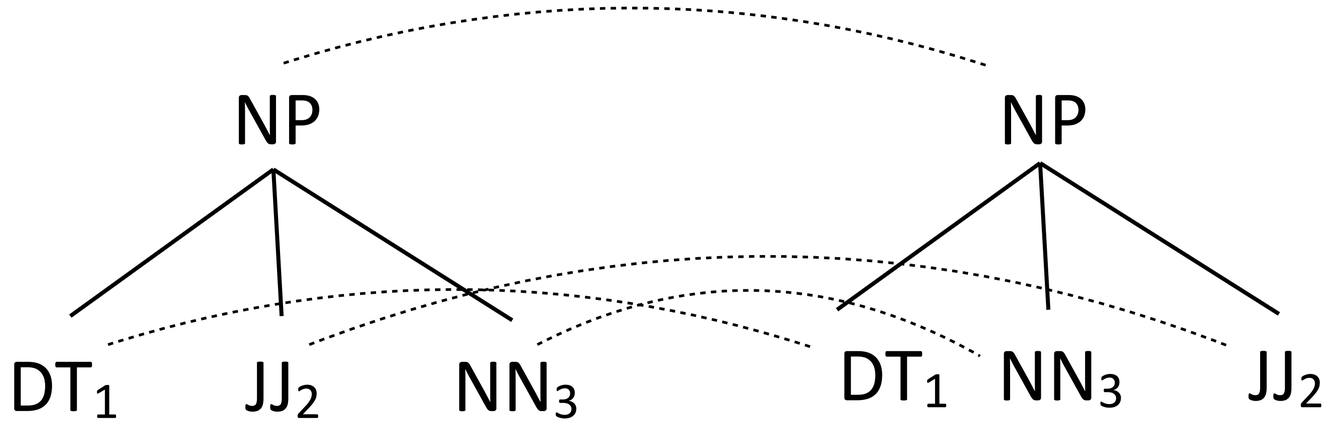
$NP \rightarrow [DT_1\ JJ_2\ NN_3;\ DT_1\ NN_3\ JJ_2]$

$DT \rightarrow$ [the, la]

$DT \rightarrow$ [the, le]

$NN \rightarrow$ [car, voiture]

$JJ \rightarrow$ [yellow, jaune]

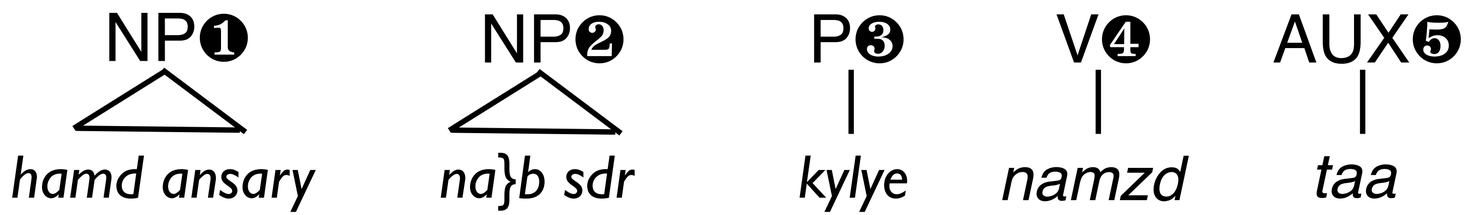


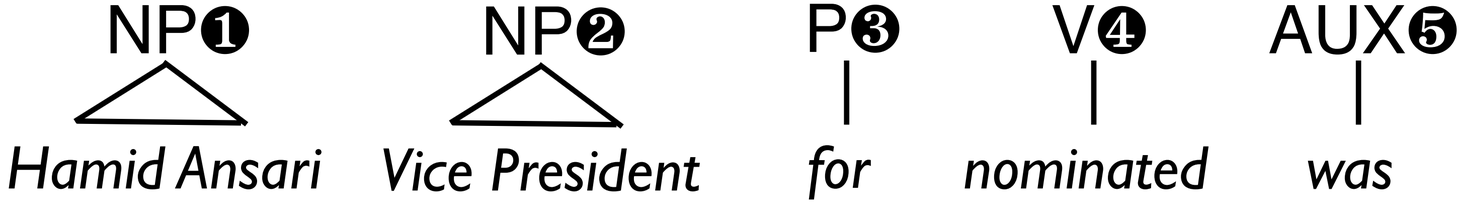
- Translation = parse the input with "half" of the grammar, read off the other half
- Assumes parallel syntax up to reordering

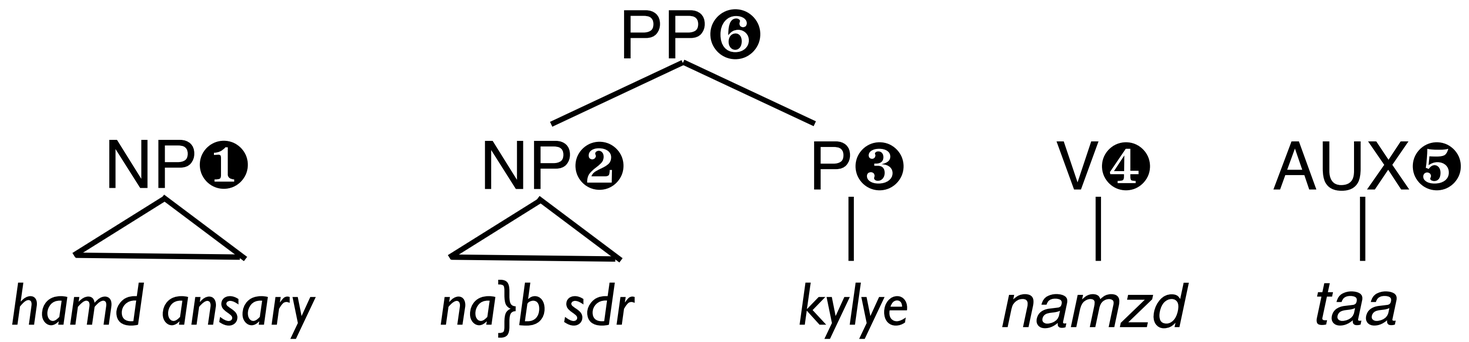
Syntactic MT

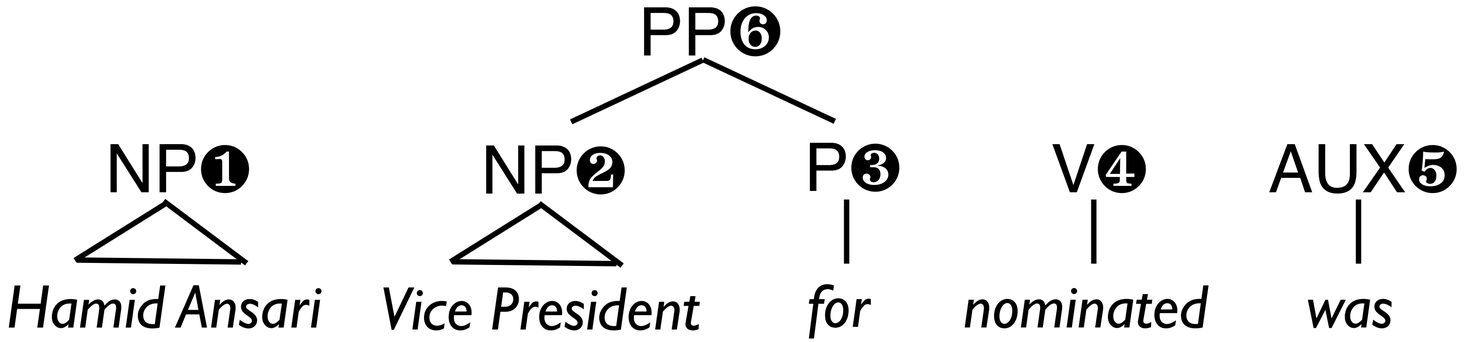
- Rather than use phrases, use a *synchronous context-free grammar*

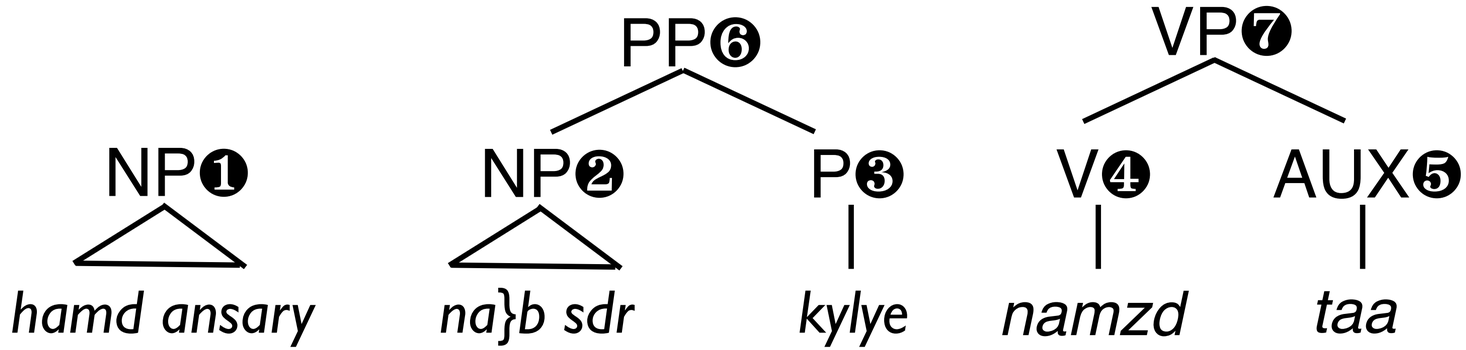
	Urdu	English
S →	NP① VP②	NP① VP②
VP→	PP① VP②	VP② PP①
VP→	V① AUX②	AUX② V①
PP →	NP① P②	P② NP①
NP →	*hamd ansary*	*Hamid Ansari*
NP →	*na}b sdr*	*Vice President*
V →	*namzd*	*nominated*
P →	*kylye*	*for*
AUX →	*taa*	*was*

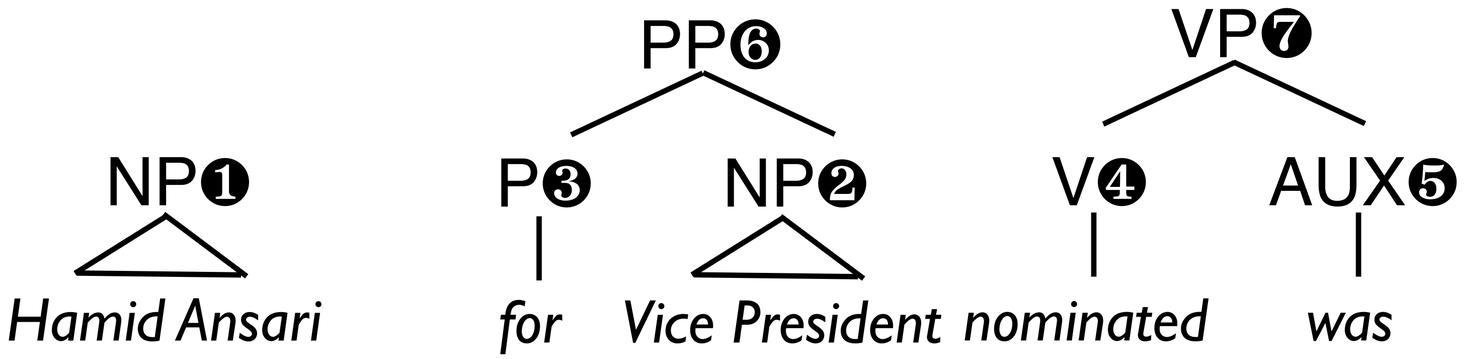

NP❶
NP❷
P❸
V❹
AUX❺
hamd ansary
na}b sdr
kylye
namzd
taa

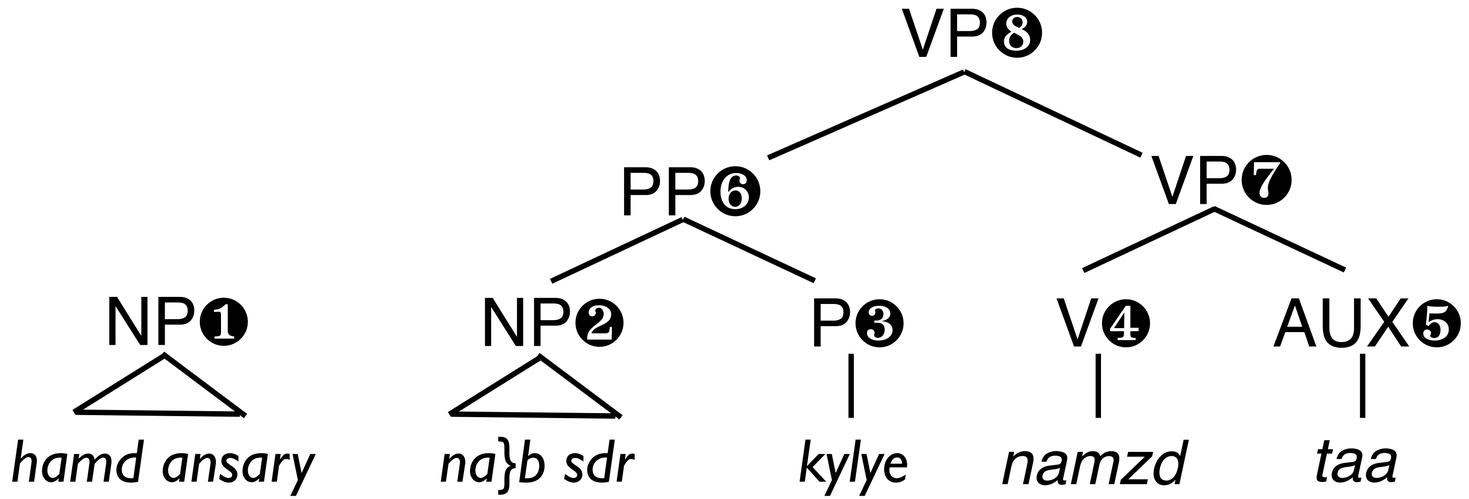

NP❶
NP❷
P❸
V❹
AUX❺
Hamid Ansari
Vice President
for
nominated
was

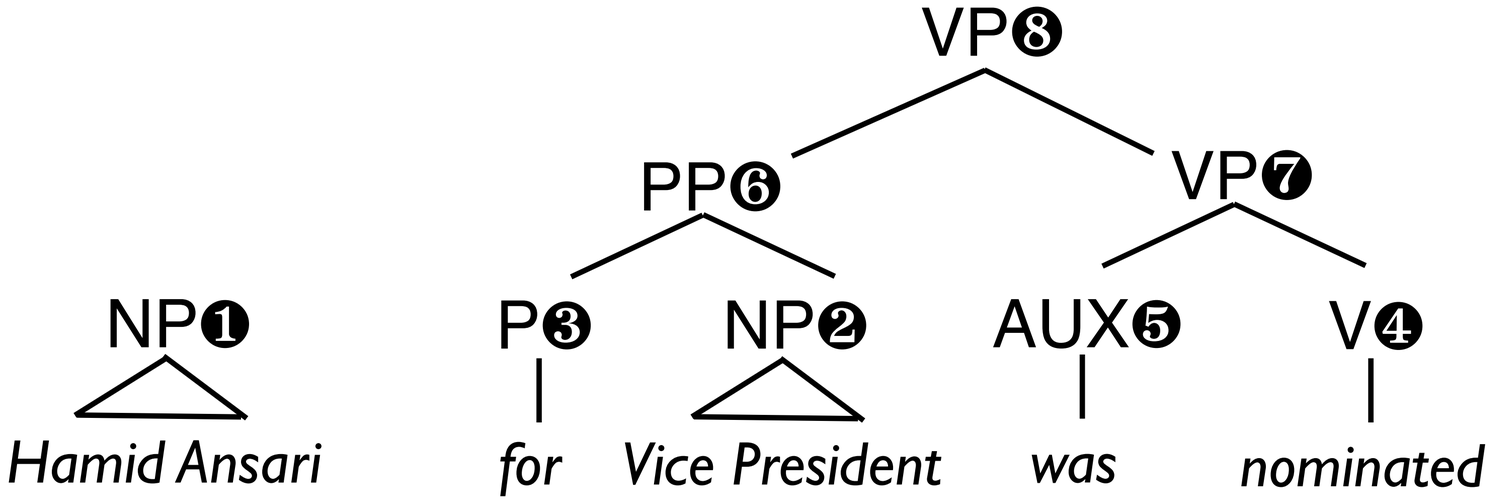

PP❻
NP❶
NP❷
P❸
V❹
AUX❺
hamd ansary
na}b sdr
kylye
namzd
taa

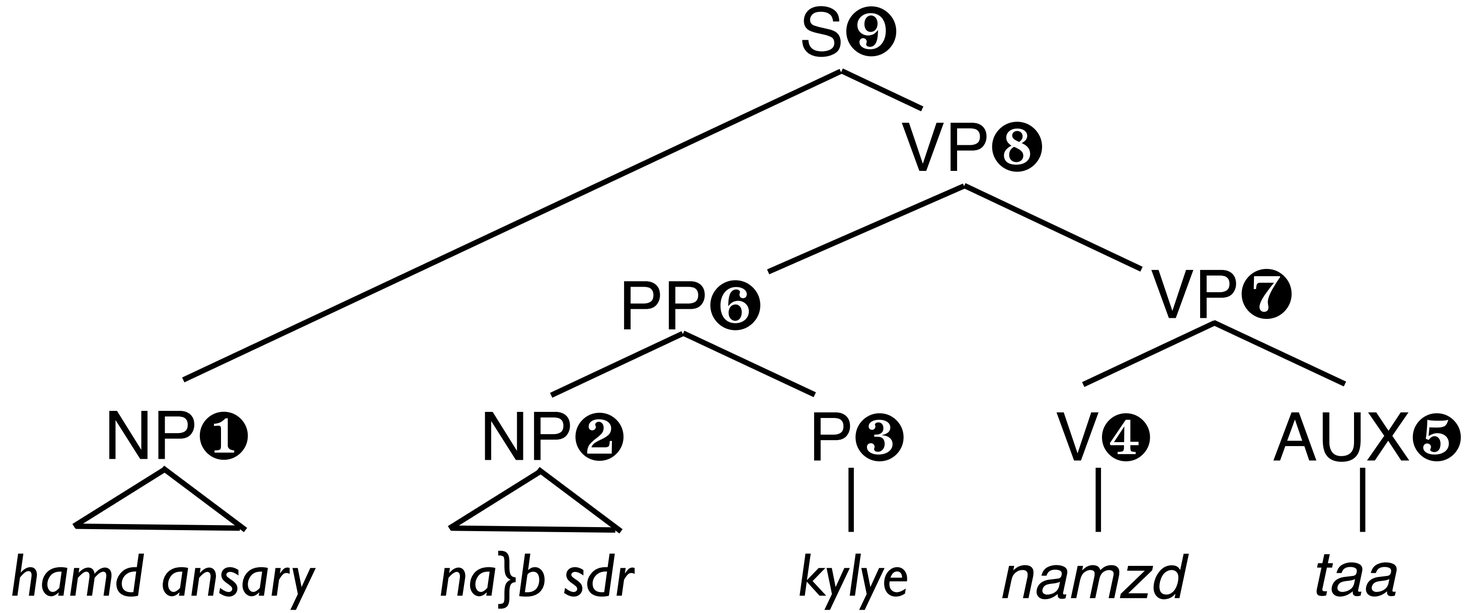

PP❻
NP❶
NP❷
P❸
V❹
AUX❺
Hamid Ansari
Vice President
for
nominated
was

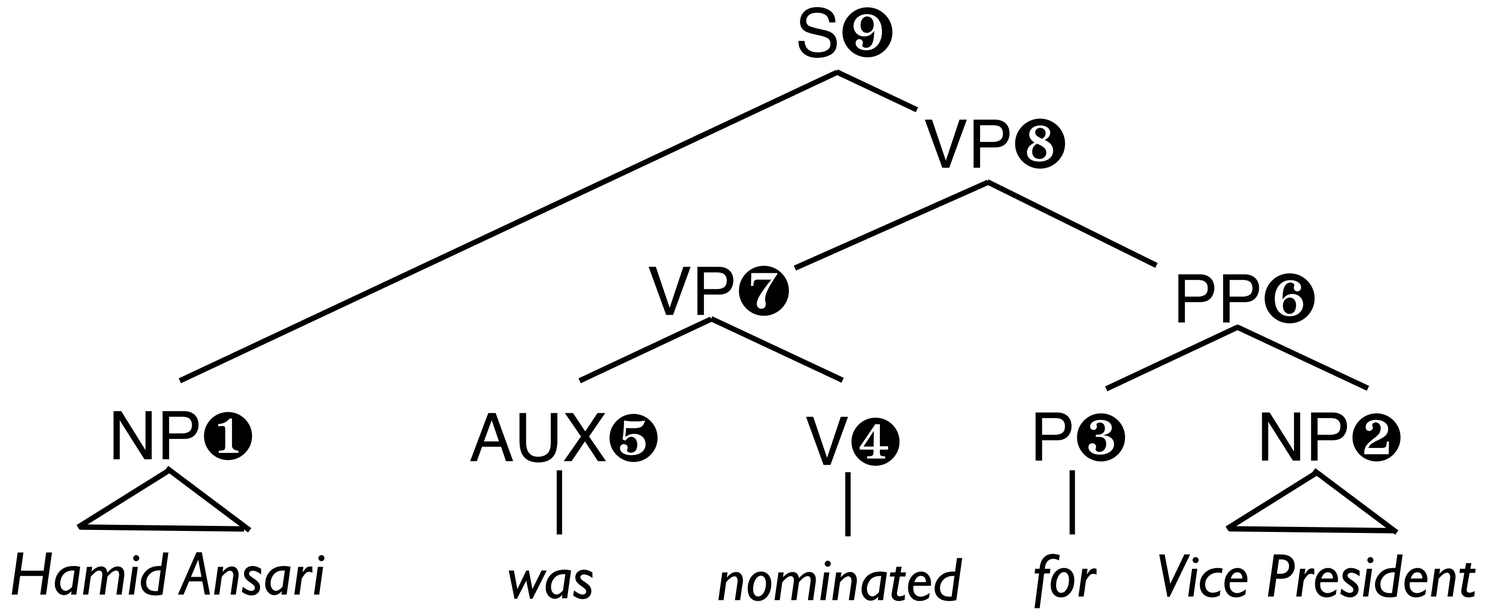

PP❻
VP❼
NP❶
NP❷
P❸
V❹
AUX❺
hamd ansary
na}b sdr
kylye
namzd
taa


PP❻
VP❼
NP❶
P❸
NP❷
V❹
AUX❺
Hamid Ansari
for
Vice President
nominated
was


VP❽
PP❻
VP❼
NP❶
NP❷
P❸
V❹
AUX❺
hamd ansary
na}b sdr
kylye
namzd
taa


VP❽
PP❻
VP❼
NP❶
P❸
NP❷
AUX❺
V❹
Hamid Ansari
for
Vice President
was
nominated


S❾
VP❽
PP❻
VP❼
NP❶
NP❷
P❸
V❹
AUX❺
hamd ansary
na}b sdr
kylye
namzd
taa


S❾
VP❽
VP❼
PP❻
NP❶
AUX❺
V❹
P❸
NP❷
Hamid Ansari
was
nominated
for
Vice President

Joshua

- Toolkit for syntactic machine translation due to many researchers at JHU (Weese, Ganitkevitch, Callison-Burch, Post, Lopez, …)

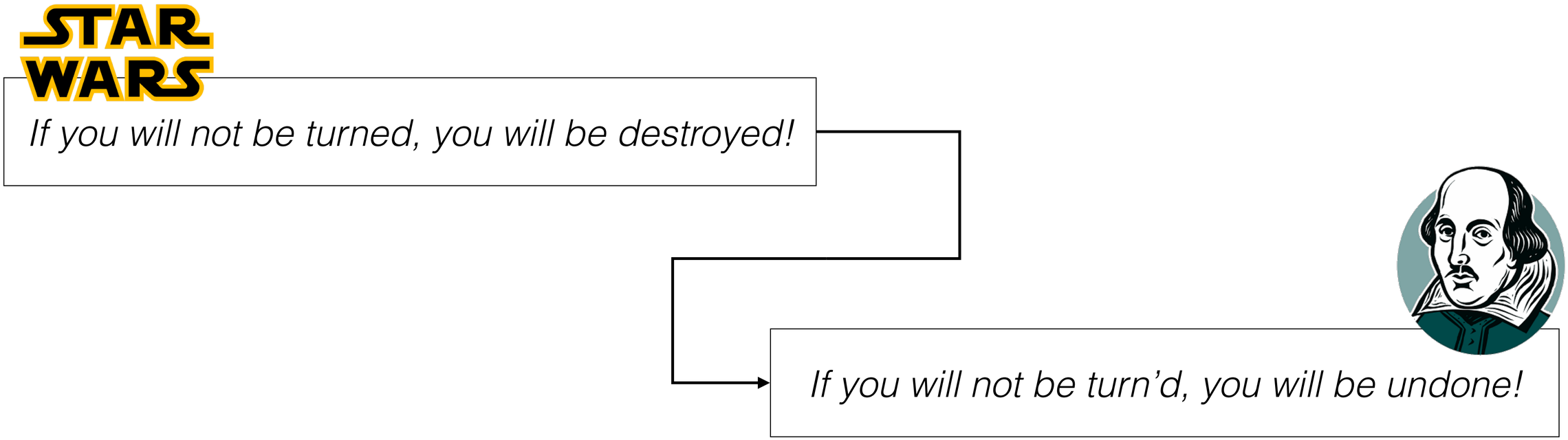
- Joshua implements synchronized grammar extraction (Thrax!), parsing, language modeling, pruning , plus *a ton* more stuff

- Joshua uses two types of SCFG: Hiero grammar (Chiang, 2007), SAMT grammar (Zollmann & Venugopal, 2007)

https://cwiki.apache.org/confluence/display/JOSHUA/

Case Studies: Monolingual MT

Style Transfer



- Applied phrase-based MT (Moses Toolkit) to Shakespearean bitext

Wei Xu, Alan Ritter, Bill Dolan, Ralph Grishman, Colin Cherry. "Paraphrasing for Style" in COLING (2012)

Text Simplification

Slightly more fourth-graders nationwide are reading proficiently compared with a decade ago, but only a third of them are now reading well, according to a new report.

transform

Most fourth-graders are better readers than they were 10 years ago.
But few of them can actually read well.

Wei Xu, Courtney Napoles, Ellie Pavlick, Chris Callison-Burch. “Optimizing Statistical Machine Translation for Simplification” in TACL (2016)

Text Simplification

Slightly more fourth-graders ~~nationwide~~ are reading proficiently compared with a decade ago, but only a third of them are now reading well, ~~according to a new report.~~

transform

Most fourth-graders are better readers than they were 10 years ago.
But few of them can actually read well.

Wei Xu, Courtney Napoles, Ellie Pavlick, Chris Callison-Burch. “Optimizing Statistical Machine Translation for Simplification” in TACL (2016)

Text Simplification

Feature Functions
(readability, language modeling, etc.)

Large-scale Paraphrases
(lexical, phrasal, syntactic)

Tuning Data
(crowdsourced multi-references)

amazon mechanical turk™
Artificial Artificial Intelligence

Objective Function
(Xu et al., 2016)

$$SARI = d_1 F_{add} + d_2 F_{keep} + d_3 P_{del}$$

$$p_{add}(n) = \frac{\sum_{g \in O} \min\left(\#_g(O \cap \bar{I}), \#_g(R)\right)}{\sum_{g \in O} \#_g(O \cap \bar{I})}$$

$$r_{add}(n) = \frac{\sum_{g \in O} \min\left(\#_g(O \cap \bar{I}), \#_g(R)\right)}{\sum_{g \in O} \#_g(R \cap \bar{I})}$$

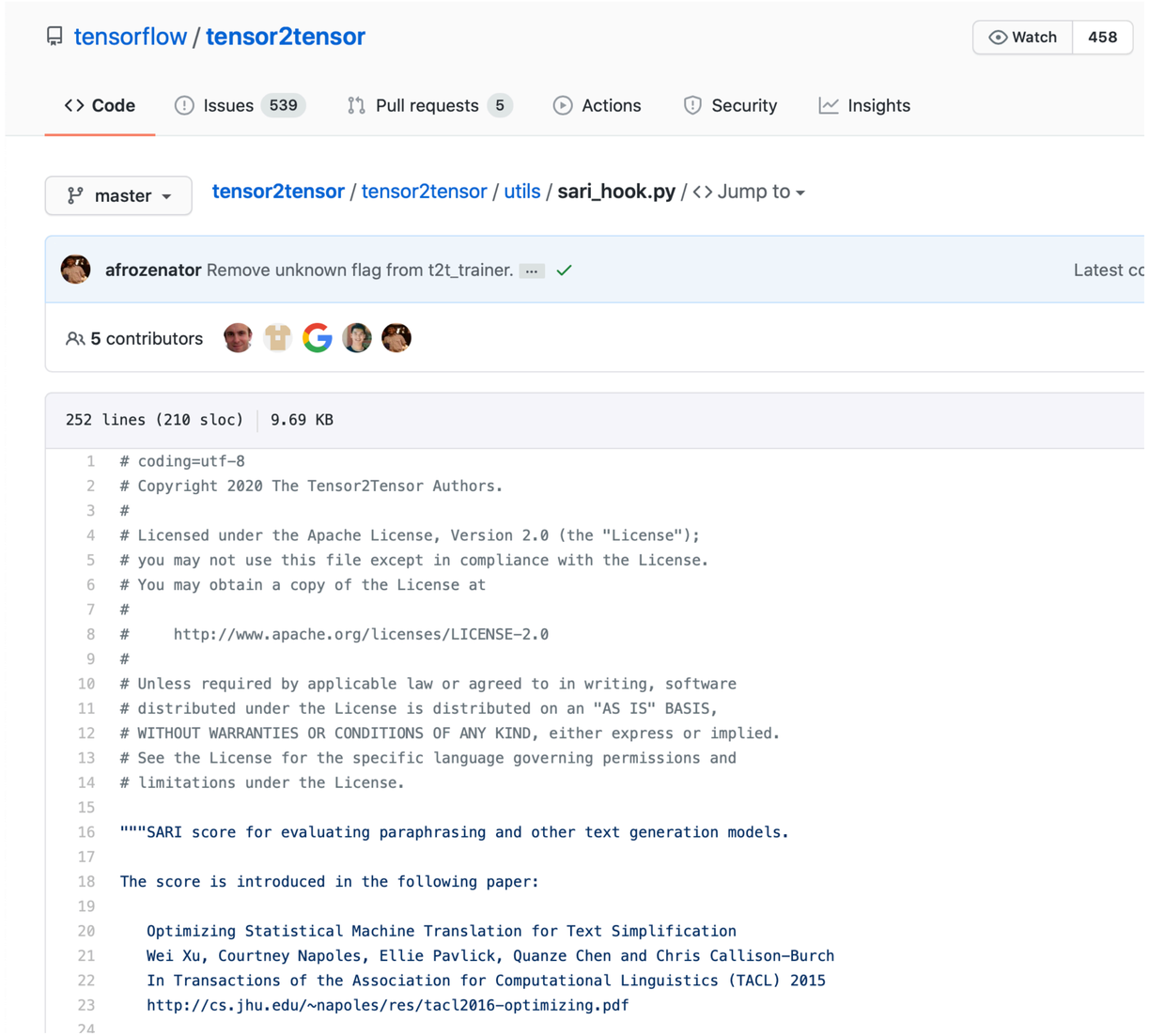
Pairwise Ranking Optimization

$$\begin{aligned} g(i,j) > g(i,j') &\Leftrightarrow h_{\mathbf{w}}(i,j) > h_{\mathbf{w}}(i,j') \\ &\Leftrightarrow h_{\mathbf{w}}(i,j) - h_{\mathbf{w}}(i,j') > 0 \\ &\Leftrightarrow \mathbf{w} \cdot \mathbf{x}(i,j) - \mathbf{w} \cdot \mathbf{x}(i,j') > 0 \\ &\Leftrightarrow \mathbf{w} \cdot (\mathbf{x}(i,j) - \mathbf{x}(i,j')) > 0 \end{aligned}$$

- Implemented by modifying 4 major components of syntax-based MT (Joshua Toolkit); SARI is now part of tensor2tensor library.

Wei Xu, Courtney Napoles, Ellie Pavlick, Chris Callison-Burch. “Optimizing Statistical Machine Translation for Simplification” in TACL (2016)

SARI for Text-to-Text Generation



tensorflow / tensor2tensor

Watch 458

Code | Issues 539 | Pull requests 5 | Actions | Security | Insights

master | tensor2tensor / tensor2tensor / utils / sari_hook.py / Jump to

afrozenator Remove unknown flag from t2t_trainer. ... ✓ Latest cc

5 contributors

252 lines (210 sloc) | 9.69 KB

```
1  # coding=utf-8
2  # Copyright 2020 The Tensor2Tensor Authors.
3  #
4  # Licensed under the Apache License, Version 2.0 (the "License");
5  # you may not use this file except in compliance with the License.
6  # You may obtain a copy of the License at
7  #
8  #     http://www.apache.org/licenses/LICENSE-2.0
9  #
10 # Unless required by applicable law or agreed to in writing, software
11 # distributed under the License is distributed on an "AS IS" BASIS,
12 # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 # See the License for the specific language governing permissions and
14 # limitations under the License.
15
16 """SARI score for evaluating paraphrasing and other text generation models.
17
18 The score is introduced in the following paper:
19
20    Optimizing Statistical Machine Translation for Text Simplification
21    Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen and Chris Callison-Burch
22    In Transactions of the Association for Computational Linguistics (TACL) 2015
23    http://cs.jhu.edu/~napoles/res/tacl2016-optimizing.pdf
24
```

- SARI metric — compares **s**ystem output **a**gainst **r**eferences and against the **i**nput sentence

SARI is added to TensorFlow by Google AI group in Feb 2019.

Google TensorFlow

Now, also in HUGGING FACE

SARI for Text-to-Text Generation

"Leveraging Pre-trained Checkpoints for Sequence Generation Tasks"

[Sascha Rothe, Shashi Narayan, Aliaksei Severyn - TACL 2020]

← using SARI for sentence splitting and fusion

"Decontextualization: Making Sentences Stand-Alone"

[Eunsol Choi, Jennimaria Palomaki, Matthew Lamm, Tom Kwiatkowski, Dipanjan Das, Michael Collins - TACL 2021]

← using SARI for sentence decontextualization: taking a sentence together with its context and rewriting it to be interpretable out of context, while preserving its meaning

"Evidence-based Factual Error Correction"

[James Thorne, Andreas Vlachos - ACL 2021]

← using SARI for revising claims based on facts correlates well with human judgements!

Takeaways

- Phrase-based systems consist of 3 pieces: aligner, language model, decoder
  - HMMs work well for alignment
  - N-gram language models are scalable and historically worked well
  - Decoder requires searching through a complex state space
- Lots of system variants incorporating syntax
- Next: neural MT