Neural Network models and Google Translate

Keith Stevens, Software Engineer - Google Translate
kstevens@google.com

Our Mission

High quality translation when and where you need it.

CONSUME

Find information and understand it

COMMUNICATE

Share with others in real life

EXPLORE

Learn a new language

1.5B
queries/day

more than 1 million books per day

225M
7-day actives

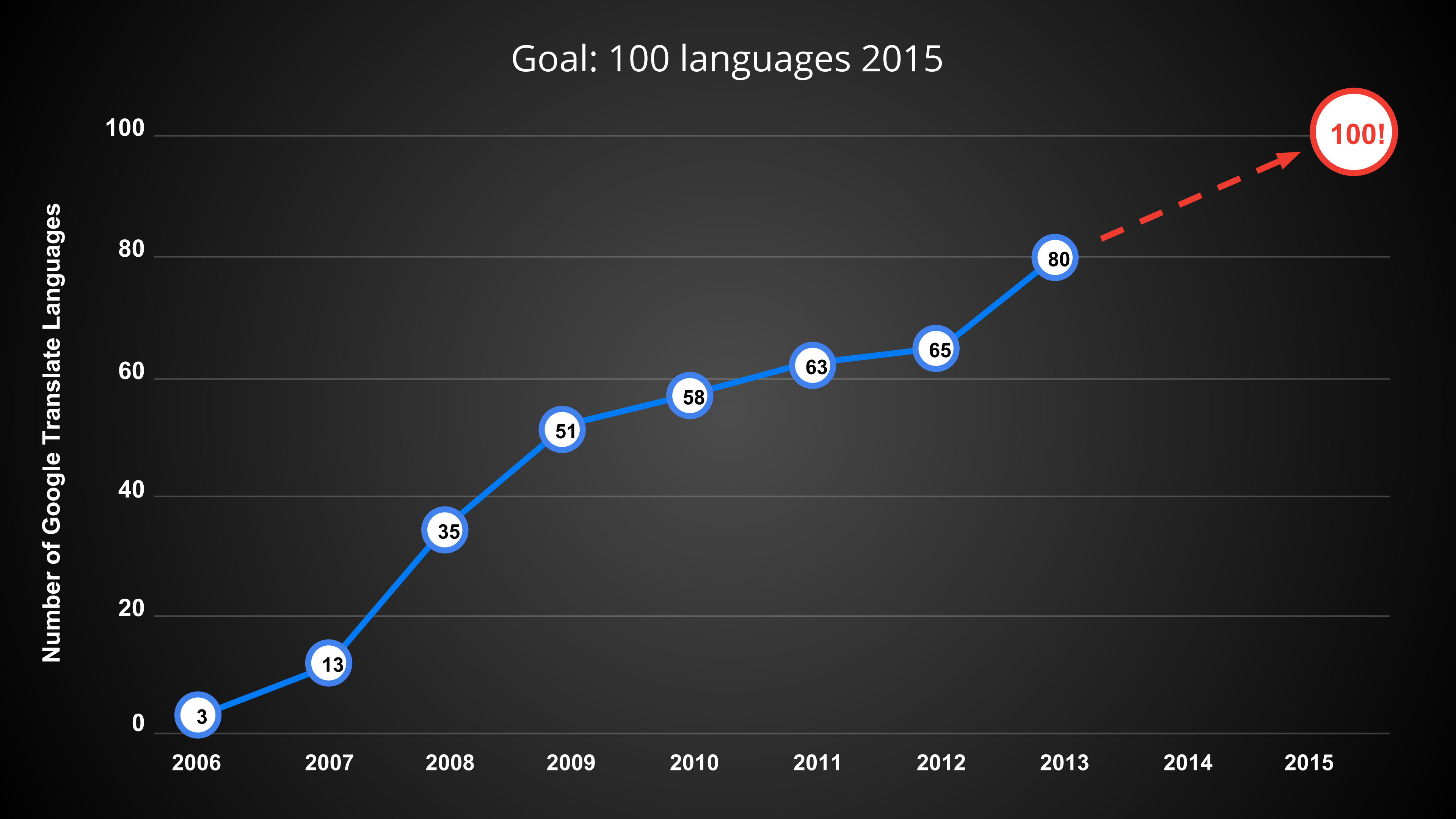
users come back for more

250M
mobile app downloads

Android + iOS

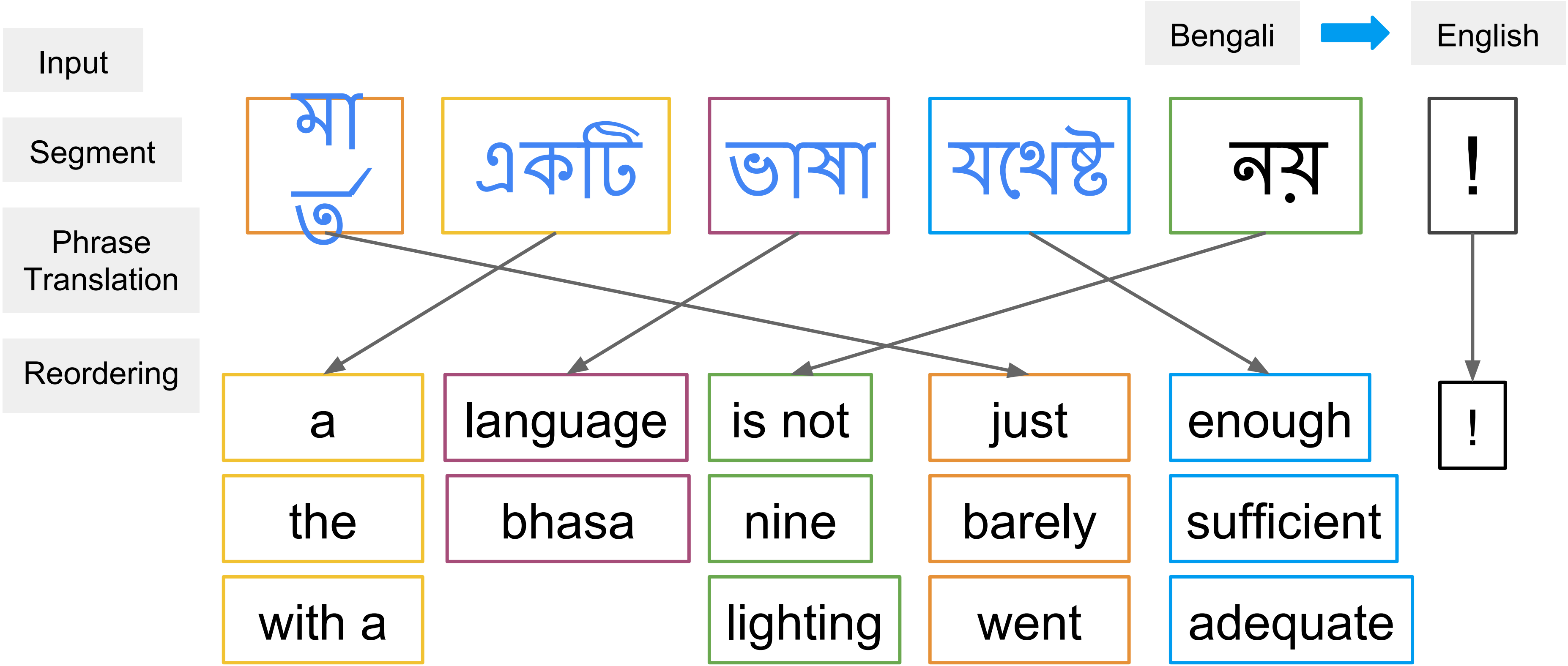
4.4
Play Store rating

Highest among popular Google apps



Goal: 100 languages 2015
Number of Google Translate Languages
100
80
60
40
20
0
3
13
35
51
58
63
65
80
100!
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015

The Model Until Today



The Model Until Today

Bengali → English

Input	মাতৃ	একটি	ভাষা	যথেষ্ট	নয়	!

Segment

Phrase Translation

Reordering

Search

Output

A language is not just enough!

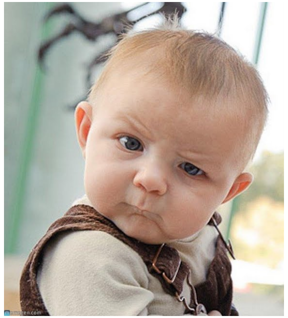
Some Fundamental Limits

I love hunting, I even bought a new bow

Je aime la chasse, je ai même acheté un nouvel arc

I don’t wear ties . I don’t know how to make a bow

Je ne porte pas de liens . Je ne sais pas comment faire un arc



Some Fundamental Limits

Chancellor Merkel exited the summit . The chancellor said that ...

La chancelière Merkel a quitté le sommet . Le chancelier a déclaré que

Neural MT [Badhanau et al. 2014]

La chancelière Merkel a quitté le sommet et le chancelier a déclaré :

Some Fundamental Limits

They near the hulking Saddam Mosque.

Ellos cerca de la hulking Mezquita Saddam.

Some Fundamental Limits

Can pure Phrase based models ever model this?

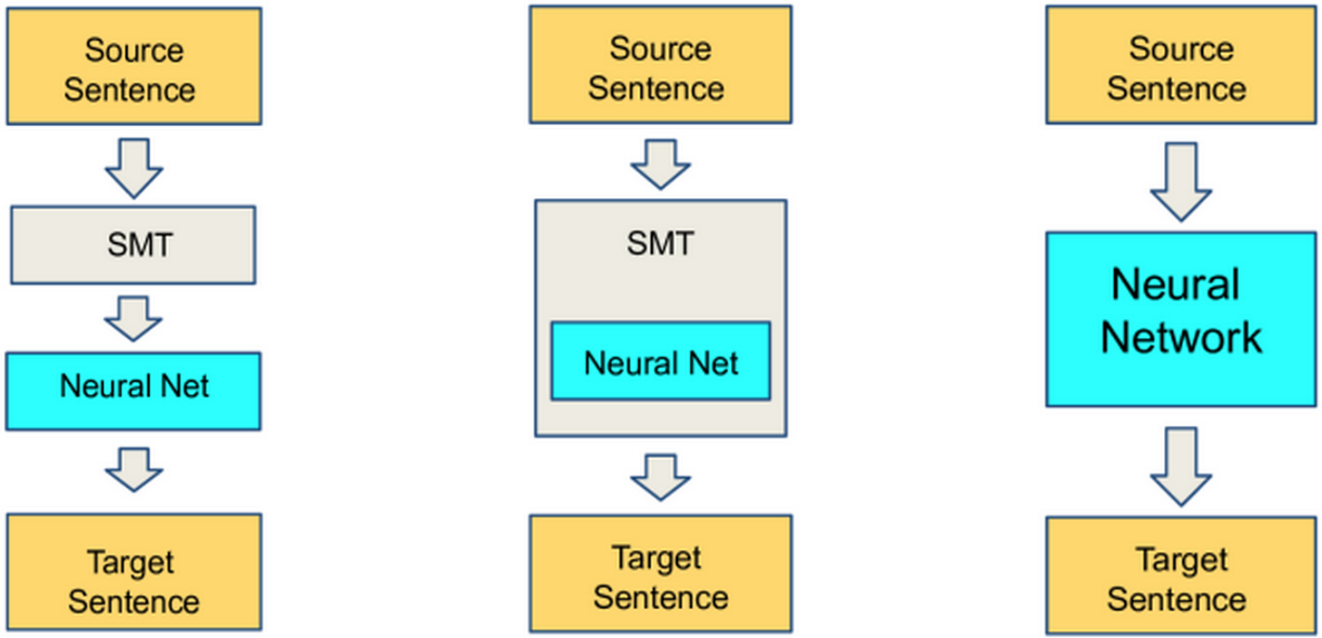
$$p(t_1 \ldots t_T | s, a) = \prod_{i=1}^{T} p(t_i | t_1 \ldots t_{i-1}, s, a)$$

How can we deal with complex word forms with few observations?

Avrupalılaştıramadıklarımızdanmışsınızcasına

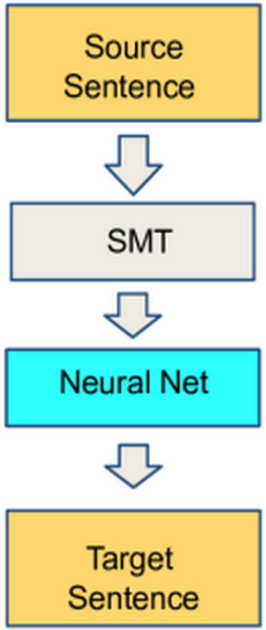
as if you are reportedly of those of ours that we were unable to Europeanize

New Neural Approaches to Machine Translation



(Schwenk, CSL 2007) (Devlin et al, ACL 2014) (Sutskever et al, NIPS 2014)

Neural Networks for scoring



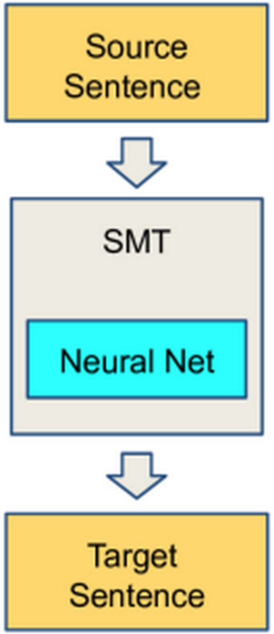
(Schwenk, CSL 2007)

Neural Language Models*

Long Short-Term Memories

* Not covered in this talk

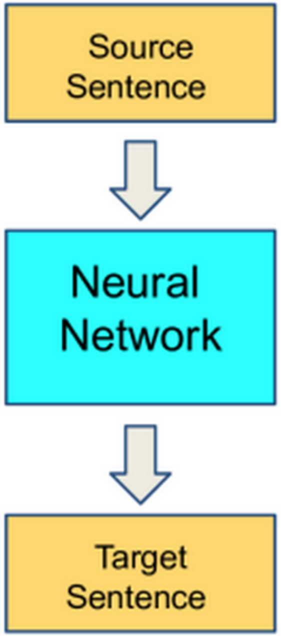
Neural Networks as Phrase Based Features



(Devlin et al, ACL 2014)

{
Morphological Embeddings
Neural Network Joint Model
Neural Network Lexical Translation Model

Neural Networks for all the translations

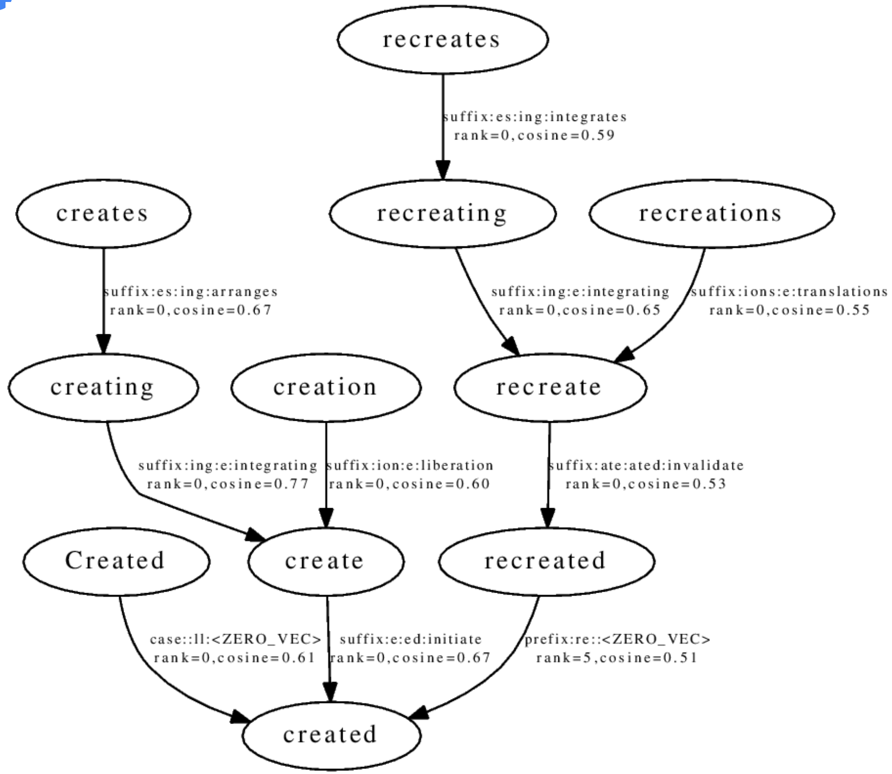


Long Short-Term Memories

(Sutskever et al, NIPS 2014)

Morphological Embeddings

1. Project words in embedding space
2. Compute Prefix-Suffix transforms
3. Compute meaning preserving graph of related forms



Using Skip-Gram Embeddings

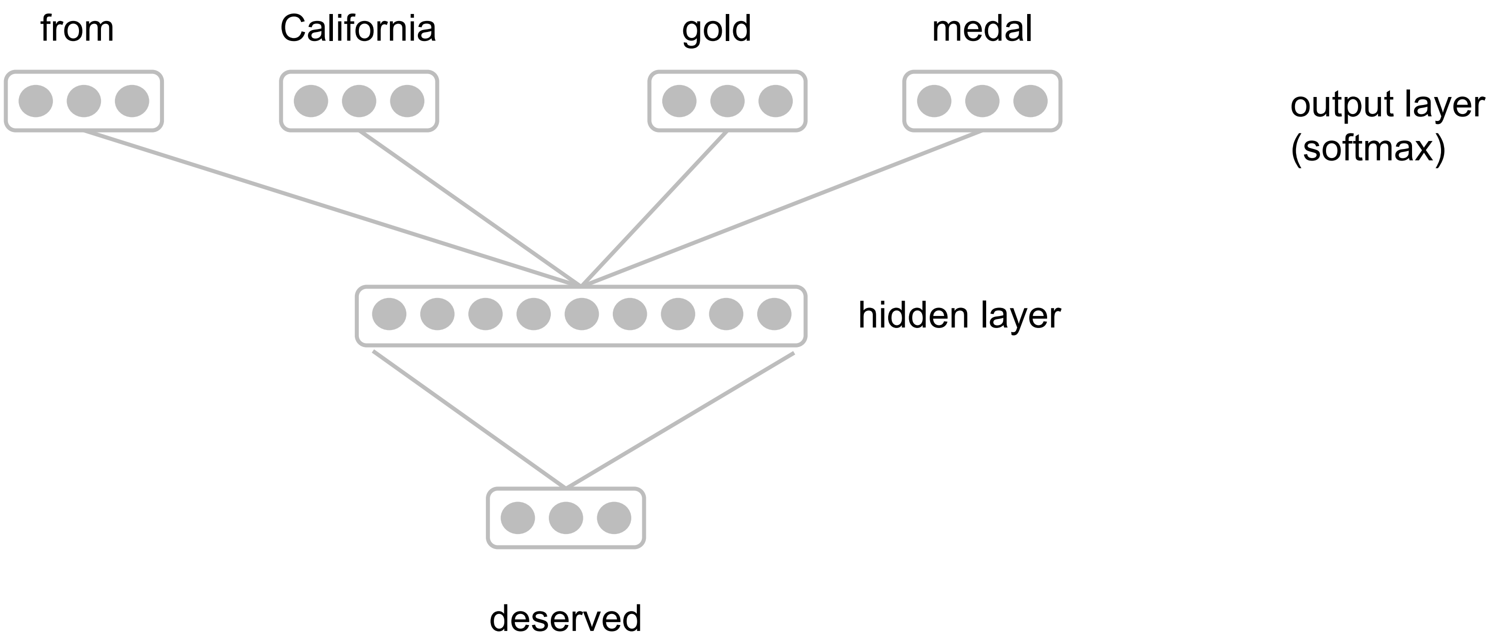
The cute dog from California <u>deserved</u> the gold medal .

D D

$$\arg\max_{v_w,v_c}\Big(\sum_{(w,c)\in D}\log\sigma(v_w\cdot v_c)+\sum_{(w,c)\in\overline{D}}\log\sigma(-v_w\cdot v_c)\Big)$$

$$\sigma(x)=\frac{1}{1+e^{x}}$$

Using Skip-Gram Embeddings



Learning Transformation Rules

Semantic Similarity

v(car) ~= v(automobile)   v(car) != v(seahorse)

Syntactic Transforms

v(anti+) = v(anticorruption) - v(corruption)

Compositionality

v(unfortunate) + v(+ly) ~= v(unfortunately)

v(king) - v(man) + v(woman) ~= v(queen)

Learning Transformation Rules

Rule	Hit Rate	Support
suffix:er:o	0.8	vote, voto

Learning Transformation Rules

Rule	Hit Rate	Support
suffix:er:o	0.8	vote, voto
prefix:ε:in	28.6	competent, incompetent

Learning Transformation Rules

Rule	Hit Rate	Support
suffix:er:o	0.8	vote, voto
prefix:ε:in	28.6	competent, incompetent
suffixed:ted:te	54.9	imitated, imitate

Learning Transformation Rules

Rule	Hit Rate	Support
suffix:er:o	0.8	vote, voto
prefix:ε:in	28.6	competent, incompetent
suffixed:ted:te	54.9	imitated, imitate
suffix:y:ies	69.6	foundry, foundries

Learning Transformation Rules

Rule	Hit Rate	Support
suffix:er:o	0.8	vote, voto
p[illegible]	[illegible]	[illegible]
s[illegible]	[illegible]	[illegible]
suffix:y:ies	69.6	foundry, foundries

$$\cos(v(\text{foundry}) - v(+y) + v(+ies), v(\text{foundries})) > t$$

Lexicalizing Transformations

suffix:ly:ε	32.1	v(only) - v(+ly) ~= v(on)	
		v(completely) - v(+ly) ~= v(complete)	

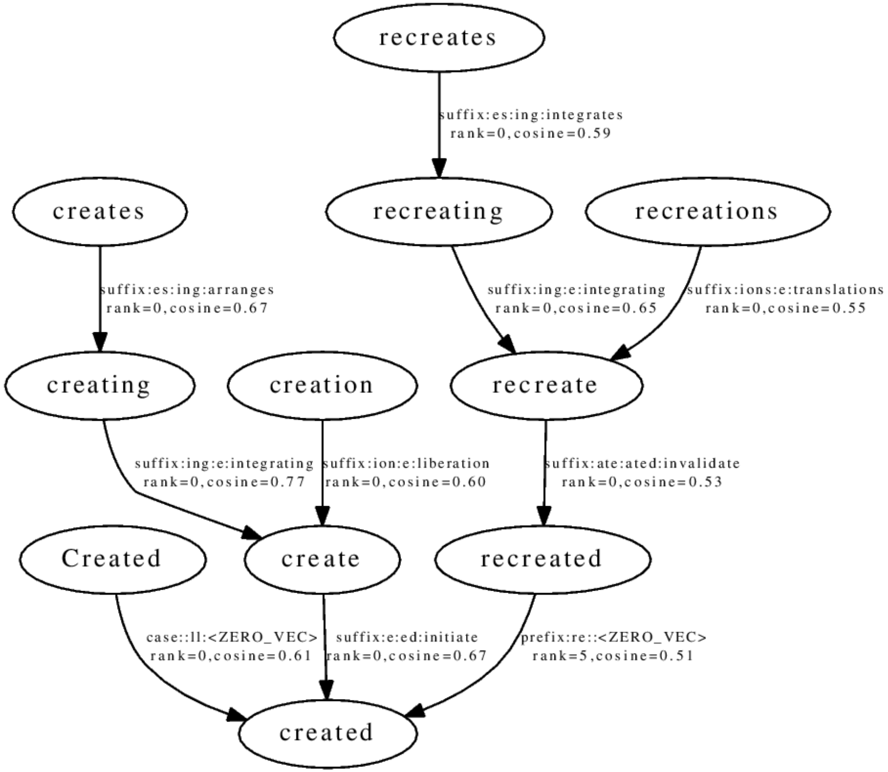
Lexicalizing Transformations

suffix:ly:ε 32.1 v(only) - ... ~= v(on)

...)

cos(v(w) - v(+ly) + v(+ε), v(w’)) > t

Lexicalizing Transformations

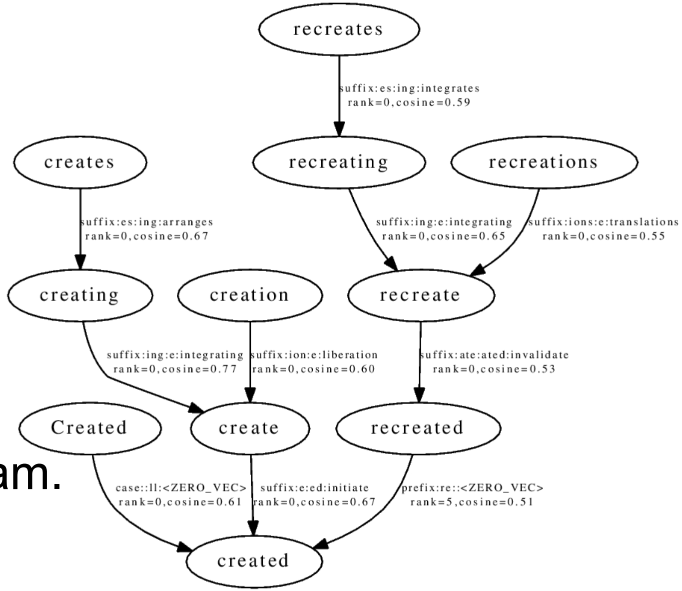


Lexicalizing Transformations

They near the hulking Saddam Mosque.

Ellos cerca de la hulking Mezquita Saddam.

Ellos cerca de la imponente Mezquita Saddam.



Using Neural Networks as Features

Ideally, use a model that jointly captures source & target information:

$$p(t_1 \dots t_T | s, a) = \prod_{i=1}^{T} p(t_i | t_1 \dots t_{i-1}, s, a)$$

But retaining entire target history explodes decoder's search space, so condition only on *n* previous words, like in *n+1*-gram LMs:

$$p(t_1 \dots t_T | s, a) \approx \prod_{i=1}^{T} p(t_i | t_{i-n} \dots t_{i-1}, s, a)$$

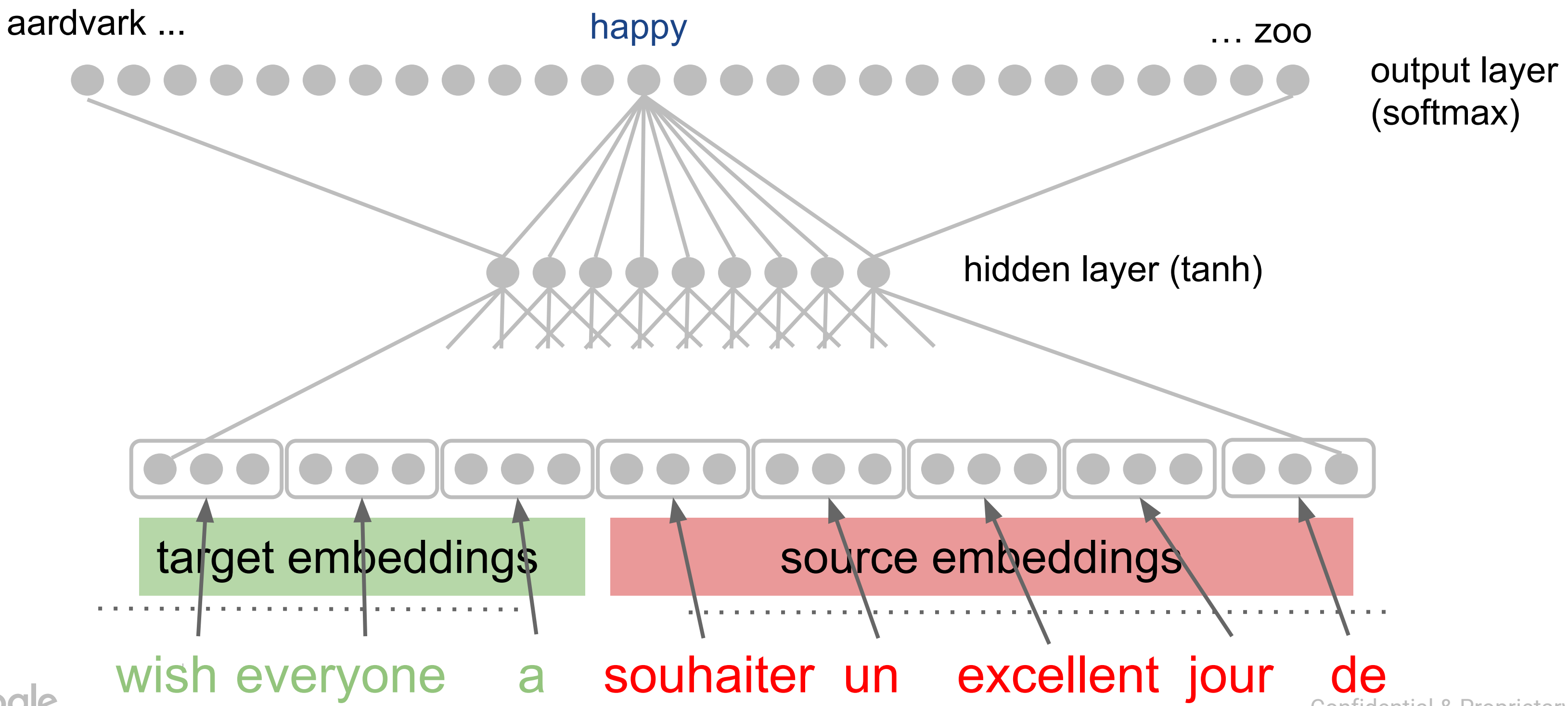
Neural Network Joint Model

Model $$p(t_i | t_{i-n} \ldots t_{i-1}, s_{a_i - m} \ldots s_{a_i + m})$$

- **Input layer** concatenates embeddings from
  - each word in n-gram target history
  - each word in 2m+1-gram source window
- **Hidden layer** weights input values, then applies tanh
- **Output layer** weights hidden activations:
  - scores every word in output vocabulary using its output embedding
  - softmax (exponentiate and normalize) scores to get probabilities
  - bottleneck due to softmax
    - O(voc-size) operations per target token in output vocabulary

NNJM

$p($ **happy** | wish everyone a souhaiter un **excellent** jour de $)$



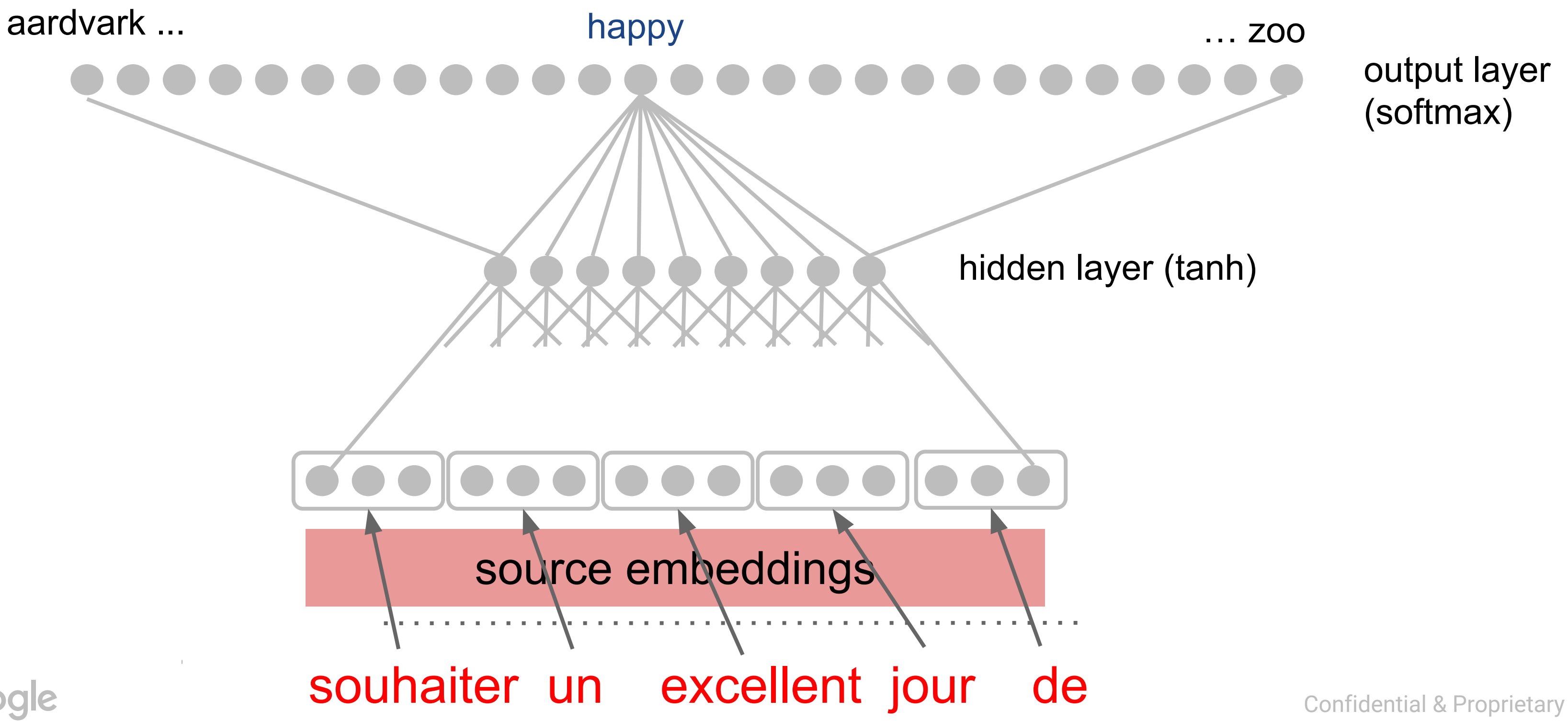
Neural Network Lexical Translation Model

Model $$p(t_i | s_{a_i - m} \ldots s_{a_i + m})$$

- **Input layer** concatenates embeddings from
  - each word in source window
- **Hidden layer** weights input values, then applies tanh
- **Output layer** weights hidden activations:
  - scores every word in output vocabulary using its output embedding
  - softmax (exponentiate and normalize) scores to get probabilities
  - bottleneck due to softmax
    - O(voc-size) operations per target token in output vocabulary

NNLTM

p(**happy** | souhaiter un **excellent** jour de)



Making NNJM & NNLTM Fast

NNJM & NNLTM Gains

Obviamente, é necessário calcular também os custos não cobertos com os médicos e a ortodontia.

Obviously, you must also calculate the cost share with doctors and orthodontics.

Obviously, you must also calculate the costs not covered with doctors and orthodontics.

A coleira transmite um pequeno "choque" eletrônico quando o cão se aproxima do limite.

The collar transmits a small "shock" when the electronic dog approaches the boundary.

The collar transmits a small "shock" electronic when the dog approaches the boundary.

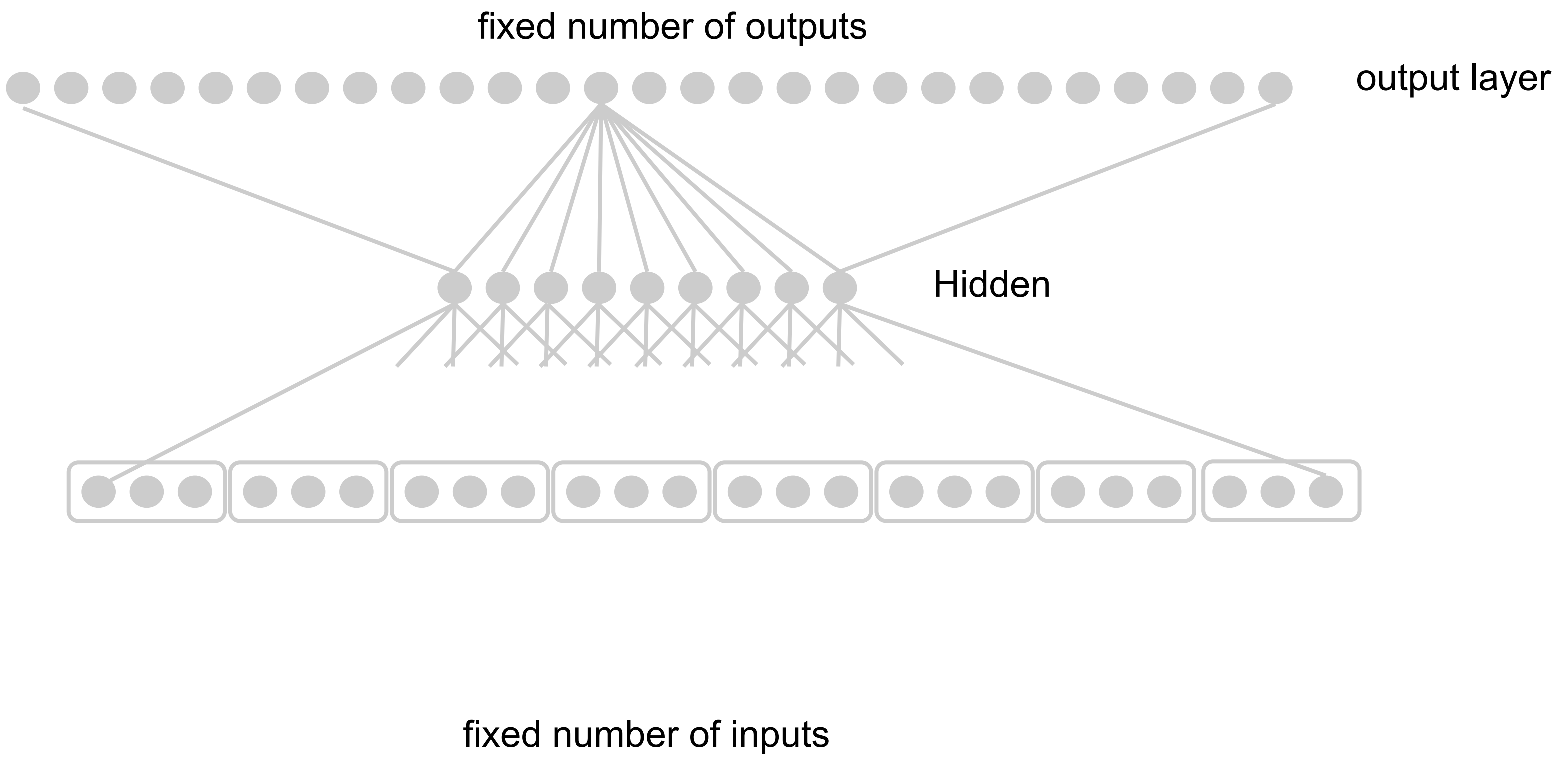
Replacing Phrase Based Models!

Ideally, use a model that jointly captures source & target information:

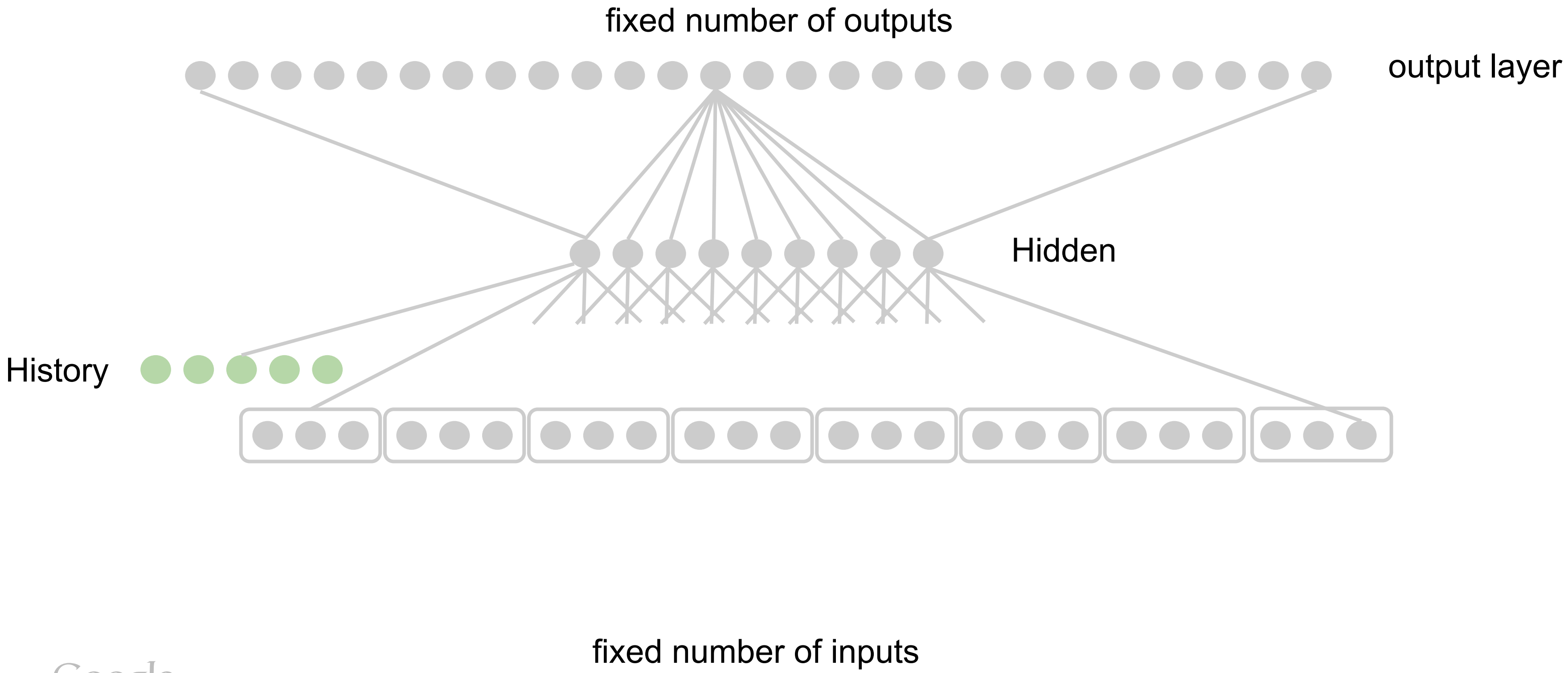
$$p(t_i | t_1 \cdots t_{i-1}, s_1 \cdots s_L)$$

Let’s actually do it!

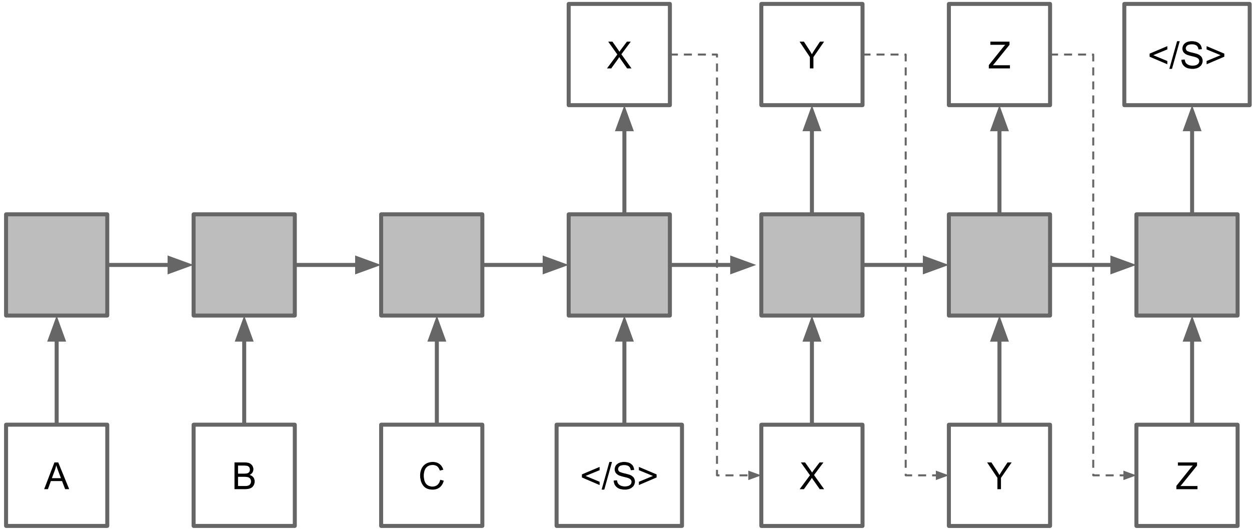
Feedforward Neural Networks



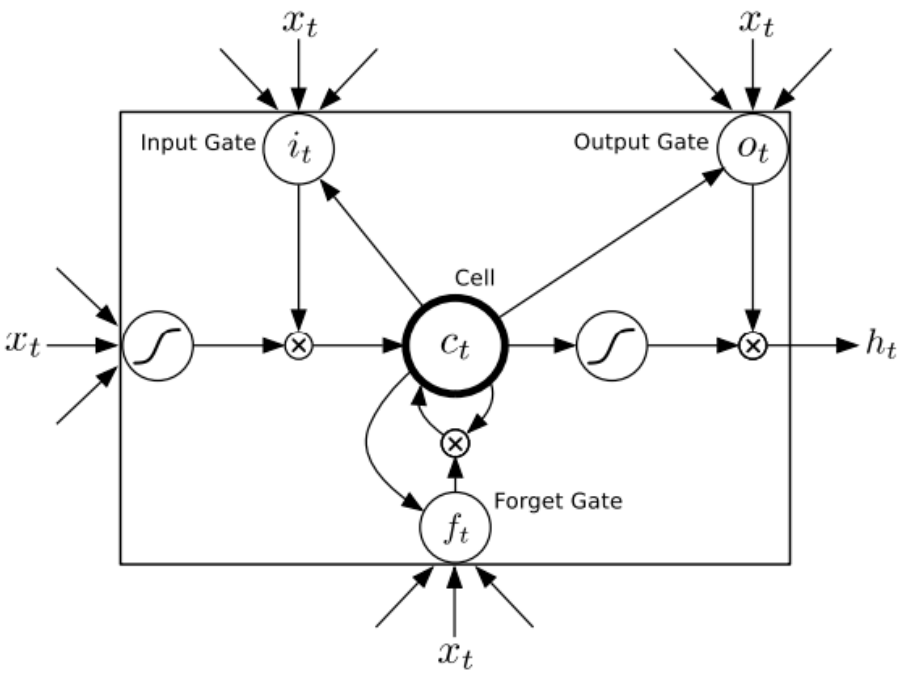
Recurrent Neural Networks



Long Short-Term Memory



A Single LSTM Cell



Model formulation

NNJM: $p(t_i|t_{i-n}...t_{i-1}, s_{a_i-m}...s_{a_i+m})$

NNLTM: $p(t_i|s_{a_i-m}...s_{a_i+m})$

S:	我	[3] 就	[4] 取	[5] 钱	[6] 给	[7] 了	她们
	i	*will*	*get*	*money*	*to*	perf.	*them*
T:	[2] i	[1] will	[0] get	the	money	to	them

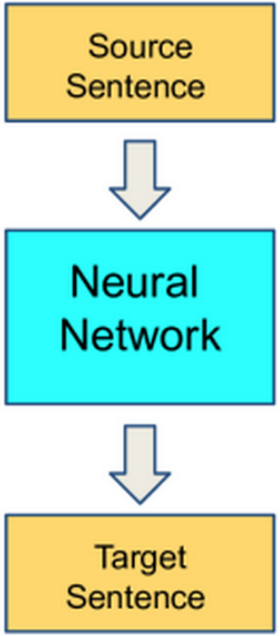
P(the | get, will, i, 就, 取, 钱, 给, 了)

(Devlin et al, ACL 2014)

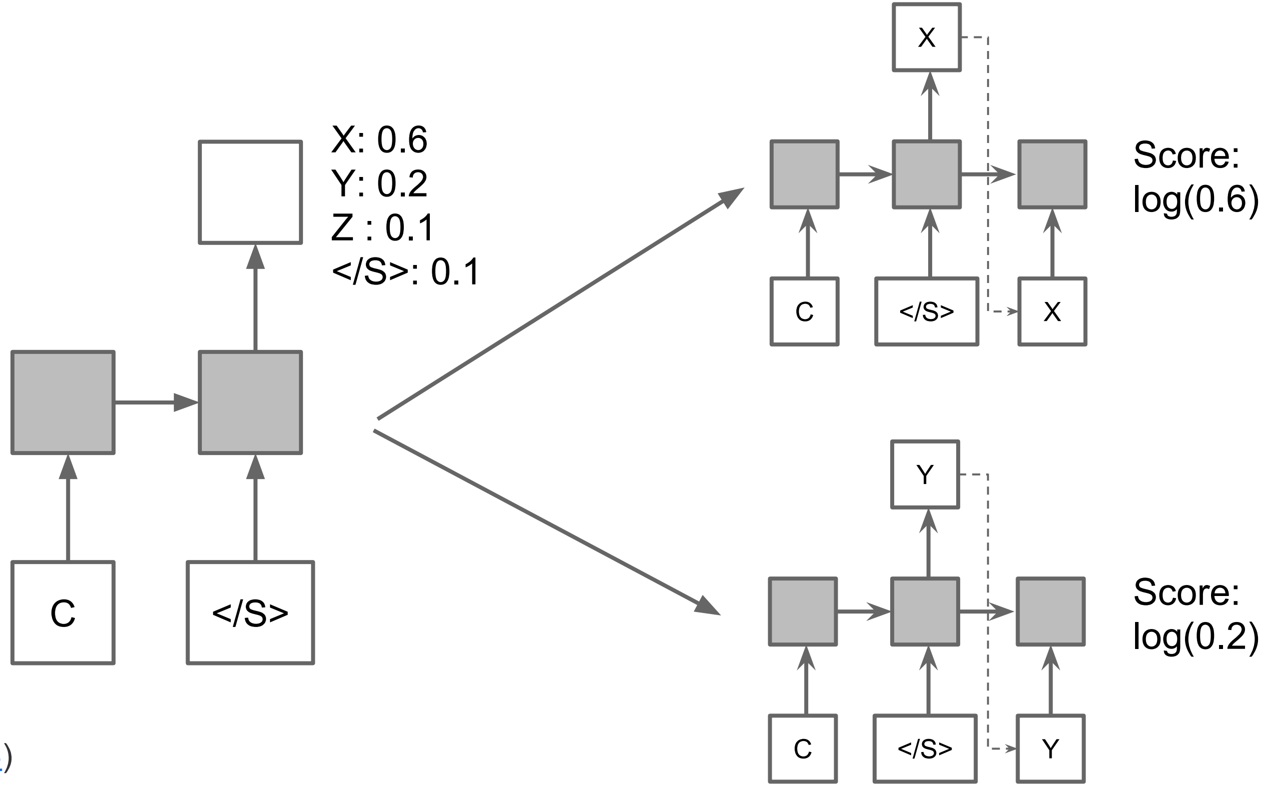
LSTM: $p(t_i|t_1 \cdots t_{i-1}, s_1 \cdots s_L)$

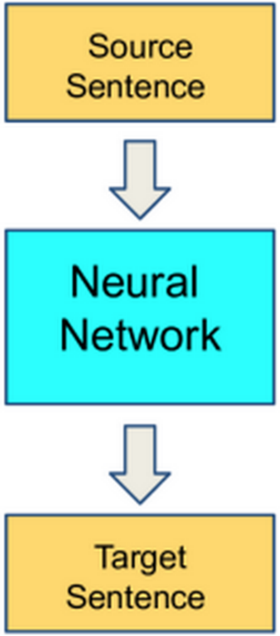
LSTMs can condition on everything; (theoretically) most powerful.

No reliance on a phrase-based system.

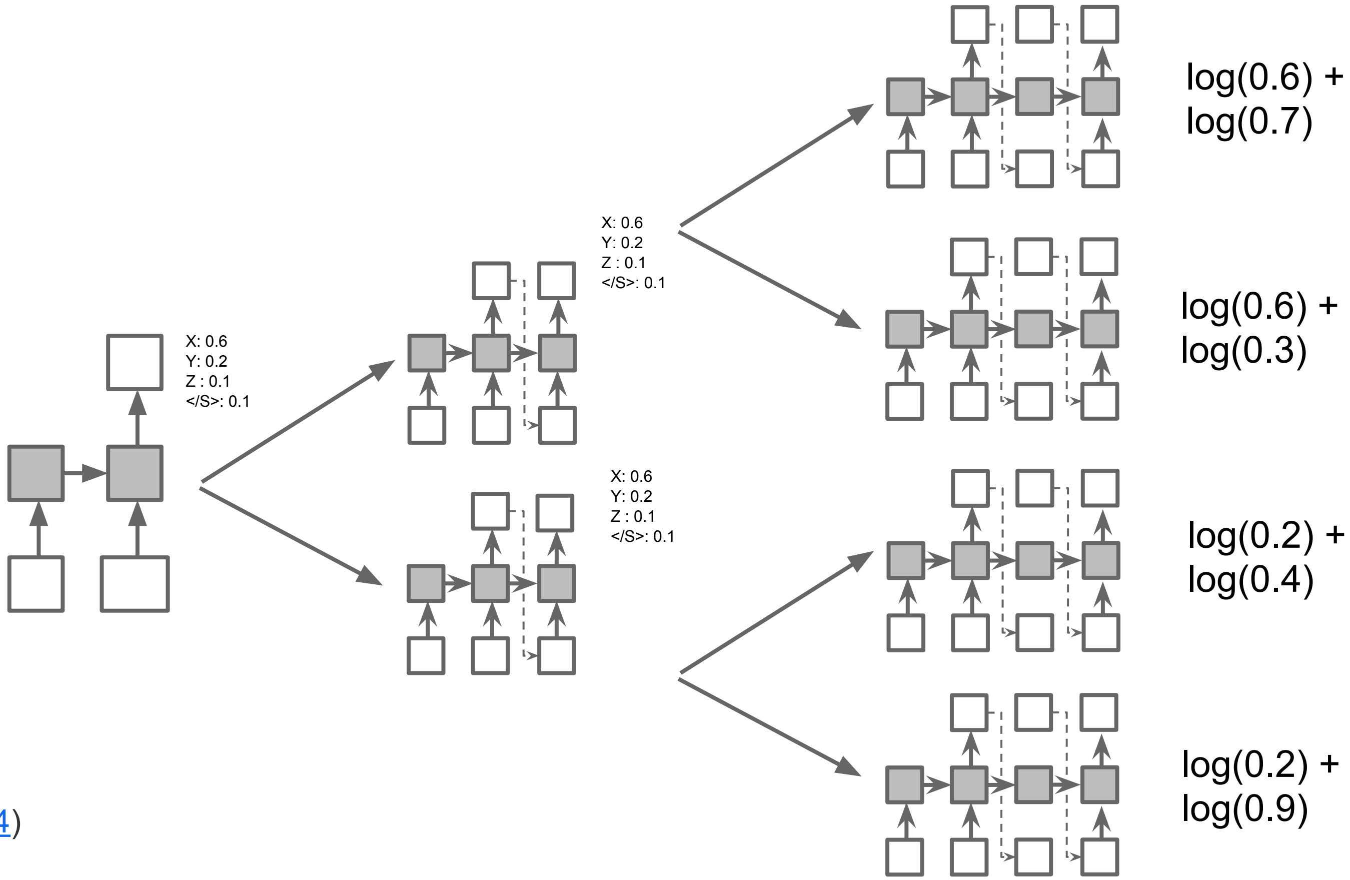


(Sutskever et al, NIPS 2014)

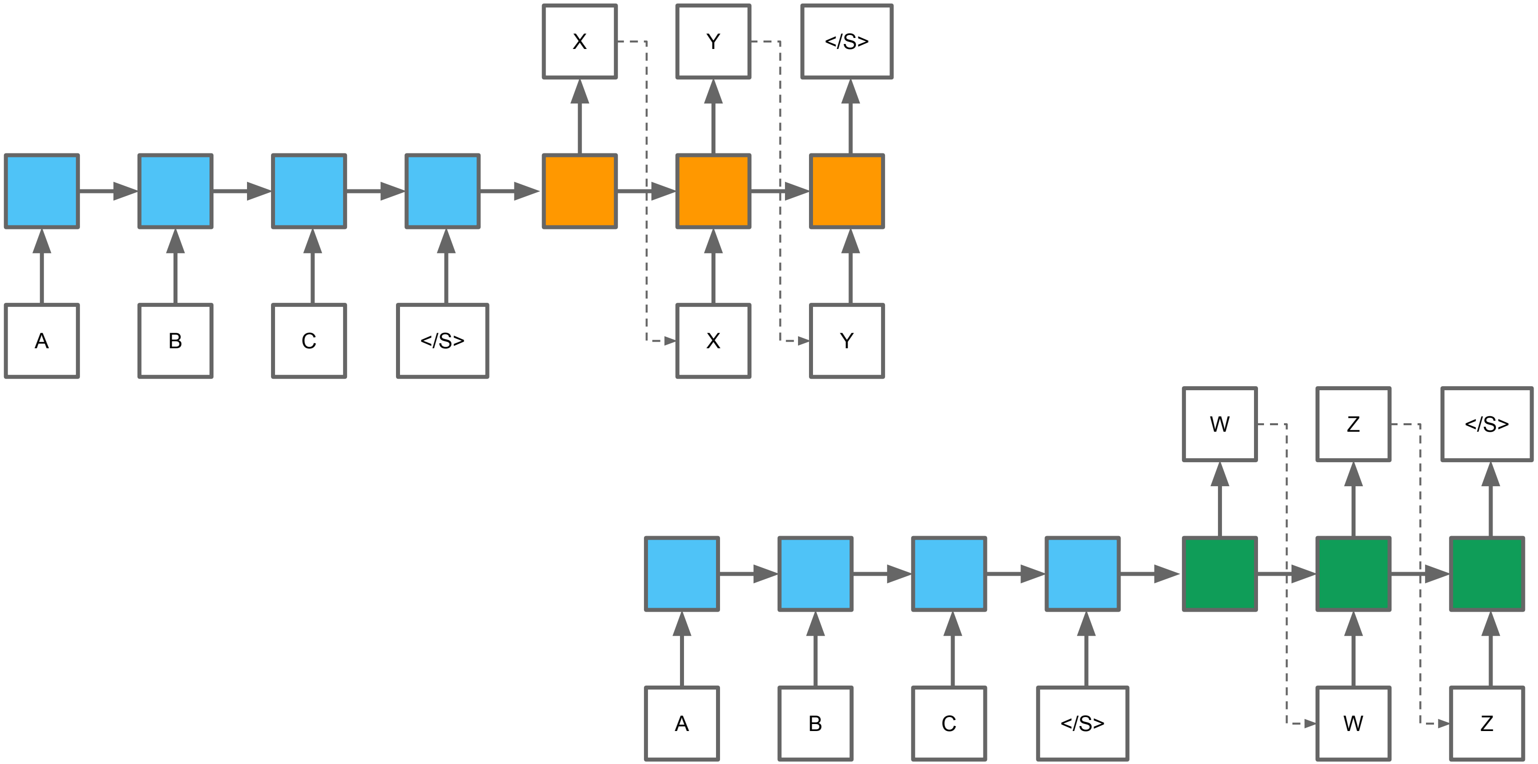




(Sutskever et al, NIPS 2014)



LSTM interlingua

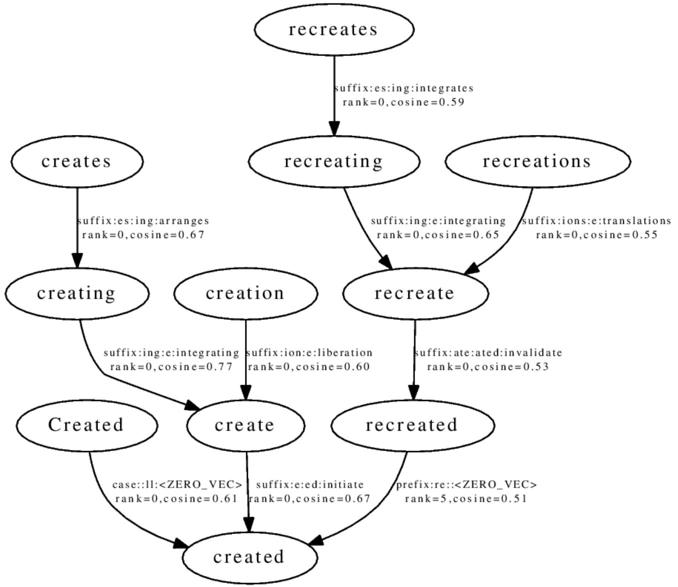


Summary

NNJM: $p(t_i | t_{i-n} ... t_{i-1}, s_{a_i - m} ... s_{a_i + m})$

NNLTM: $p(t_i | s_{a_i - m} ... s_{a_i + m})$

LSTM: $p(t_i | t_1 \cdots t_{i-1}, s_1 \cdots s_L)$



JOIN US!

HOW TO APPLY

1. Visit google.com/careers/students to find the best role(s) for you.
2. Upload your resumé and transcript (unofficial is fine)
3. Click submit!

HOW TO APPLY

kstevens@google.com

1. Visit goo ne best
   role(s) fo
2. Upload y al is fine)
3. Click sub