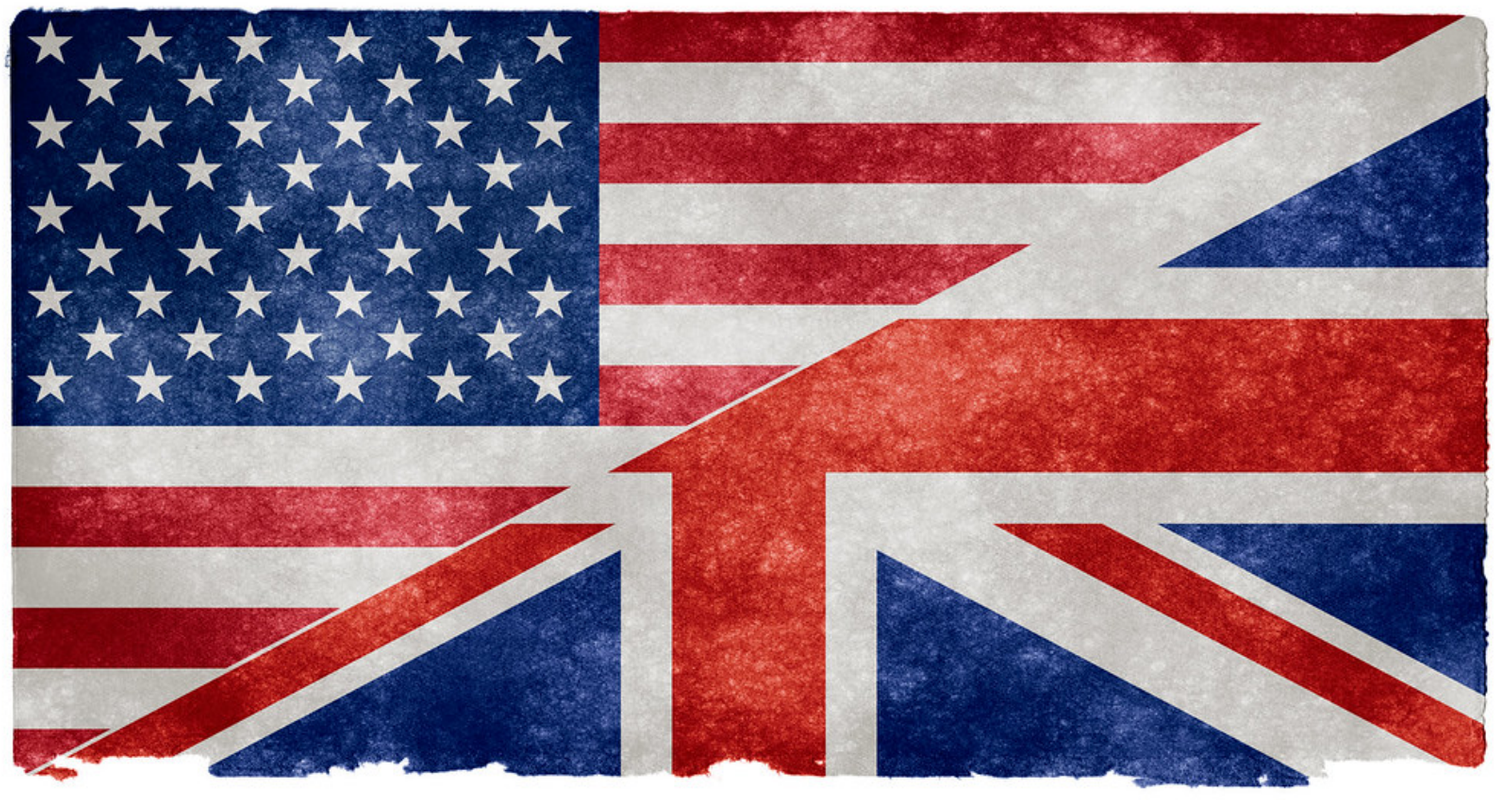
Transfer Learning for Natural Language Processing

Katharina Kann — SMILES 2020

Current NLP



The Problem: Limited Resources

- Deep learning models need a lot of training data
- Data can be limited for:

The Problem: Limited Resources

- Deep learning models need a lot of training data
- Data can be limited for:
  - Languages

The Problem: Limited Resources

- Deep learning models need a lot of training data
- Data can be limited for:
  - Languages
  - Tasks

The Problem: Limited Resources

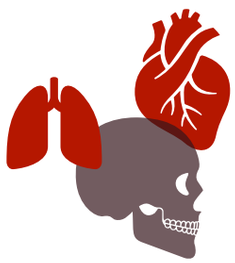
- Deep learning models need a lot of training data

- Data can be limited for:

  - Languages

  - Tasks

  - Domains



Wikipedia

- English: ~49M pages
- Chinese: ~5M pages
- Spanish: ~5M pages
- German: ~5M pages
- Norwegian: ~1M pages
- Afrikaans: ~200k pages

Wikipedia

- English: ~49M pages
- Chinese: ~5M pages
- Spanish: ~5M pages
- German: ~5M pages
- Norwegian: ~1M pages
- Afrikaans: ~200k pages

NLP systems cannot just be trained for Afrikaans as they can for English!

What is Transfer Learning?

- Sharing knowledge across tasks, languages, domains
- Different sources complement and inform each other
- Reduces the amount of data required for certain…
  - …languages
  - …tasks
  - …domains

What is Transfer Learning?

- Sharing knowledge across tasks, languages, domains
- Different sources complement and inform each other
- Reduces the amount of data required for certain…
  - …languages
  - …tasks
  - …domains

Has lead to important improvements in NLP in the last years!

Why Transfer Learning?

SuperGLUE GLUE Paper Code Tasks Leaderboard FAQ Diagnostics Submit Login

Leaderboard Version: **2.0**

Rank	Name	Model	URL	Score	BoolQ	CB	COPA	MultiRC	ReCoRD	RTE	WiC	WSC	AX-b	AX-g
1	SuperGLUE Human Baselines	SuperGLUE Human Baselines		89.8	89.0	95.8/98.9	100.0	81.8/51.9	91.7/91.3	93.6	80.0	100.0	76.6	99.3/99.7
2	Facebook AI	RoBERTa		84.6	87.1	90.5/95.2	90.6	84.4/52.5	90.6/90.0	88.2	69.9	89.0	57.9	91.0/78.1
3	SuperGLUE Baselines	BERT++		71.5	79.0	84.8/90.4	73.8	70.0/24.1	72.0/71.3	79.0	69.6	64.4	38.0	99.4/51.4
		BERT		69.0	77.4	75.7/83.6	70.6	70.0/24.1	72.0/71.3	71.7	69.6	64.4	23.0	97.8/51.7
		Most Frequent Class		47.1	62.3	21.7/48.4	50.0	61.1/0.3	33.4/32.5	50.3	50.0	65.1	0.0	100.0/50.0
		CBoW		44.5	62.2	49.0/71.2	51.6	0.0/0.5	14.0/13.6	49.7	53.1	65.1	-0.4	100.0/50.0
		Outside Best		-	80.4	-	84.4	70.4/24.5	74.8/73.0	82.7	-	-	-	-
4	IBM Research AI	BERT-mtl		71.3	84.8	89.6/94.0	72.2	73.2/30.5	74.6/74.0	84.1	50.0	61.0	29.6	97.8/57.3
-	Stanford Hazy Research	Snorkel [SuperGLUE v1.9]		-	-	88.6/93.2	76.2	76.4/36.3	-	78.9	72.1	72.6	47.6	-

Click on a submission to see more information

Timeline Today

- Pretraining
- Adaptation
- Multi-Task Training
- Cross-Lingual Transfer

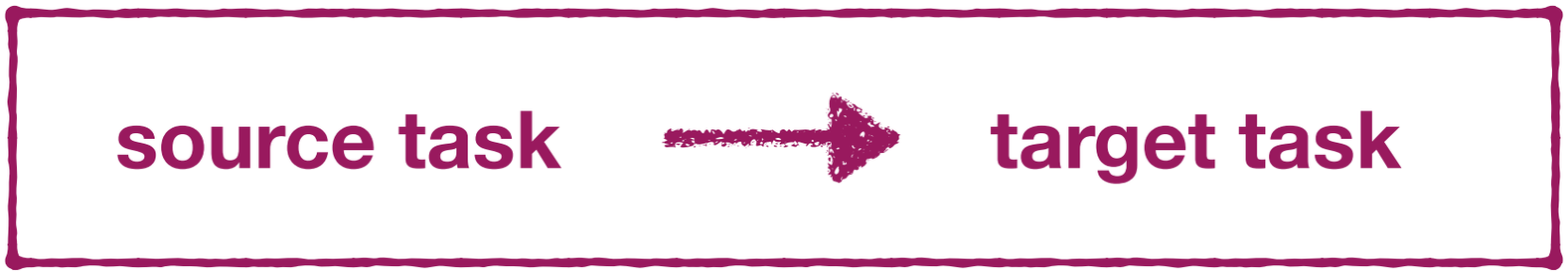
Pretraining

Pretraining

- Train a model on large amounts of data from a source task which is different from the target task
- Learned knowledge will (hopefully) be useful for target task
- General properties of language can even be learned from raw text

Pretraining

- Train a model on large amounts of data from a source task which is different from the target task
- Learned knowledge will (hopefully) be useful for target task
- General properties of language can even be learned from raw text

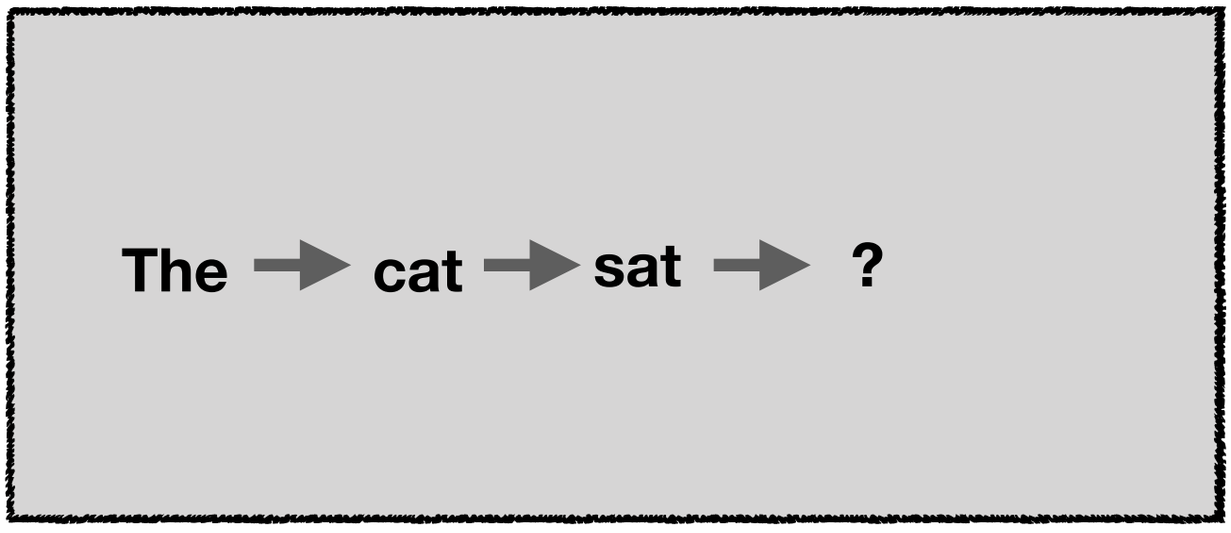


Pretraining

- Any task could be used in theory
- Popular examples:
  - Language modeling
  - Copy task
  - Natural language inference

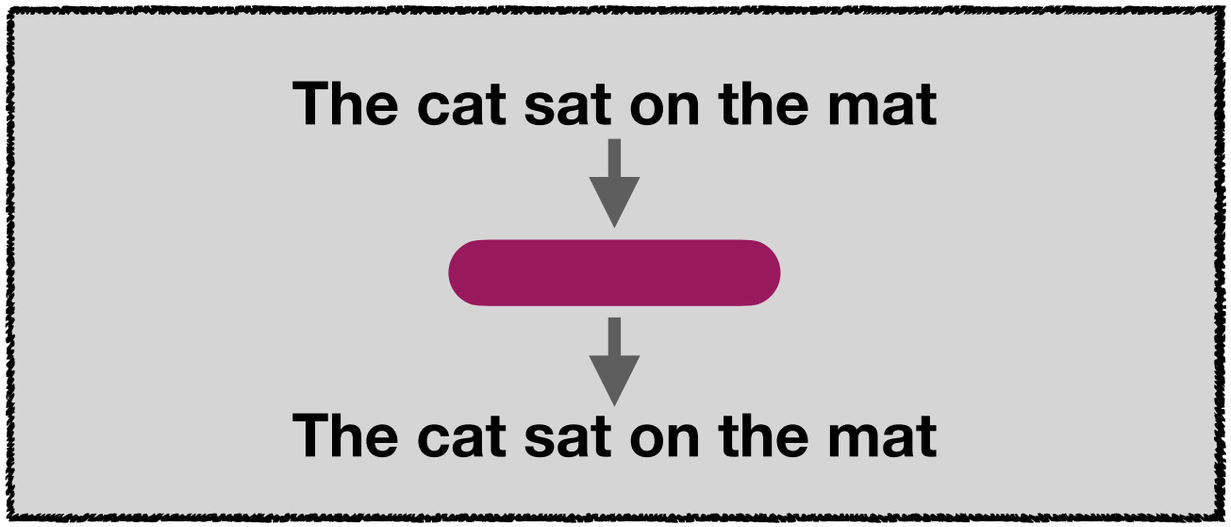
Pretraining

- Any task could be used in theory
- Popular examples:
  - Language modeling
  - Copy task
  - Natural language inference



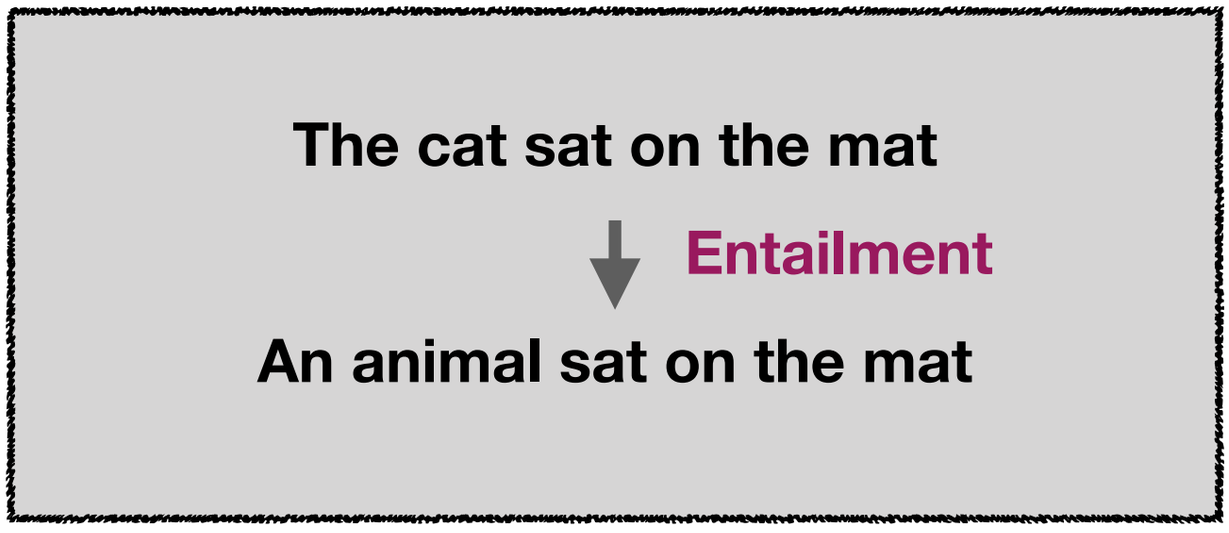
Pretraining

- Any task could be used in theory
- Popular examples:
  - Language modeling
  - Copy task
  - Natural language inference



Pretraining

- Any task could be used in theory
- Popular examples:
  - Language modeling
  - Copy task
  - Natural language inference



Pretraining

- Any task could be used in theory
- Popular examples:
  - Language modeling
  - Copy task
  - Natural language inference

Many source tasks are chosen because a lot of data are available

Example: Word Embeddings

- **Firth** (1957): “You shall know a word by the company it keeps.”

- **Harris** (1954): “distributional statements can cover all of the material of a language without requiring support from other types of information.”

Word Embeddings

- The **<word>** is a small carnivorous mammal.

Word Embeddings

- The **<word>** is a small carnivorous mammal.
- The **<word>** is a predator that is most active at dawn and dusk.

Word Embeddings

- The **<word>** is a small carnivorous mammal.
- The **<word>** is a predator that is most active at dawn and dusk.
- As of 2017, the domestic **<word>** was the second-most popular pet in the United States by number of pets owned.

Main idea:

Combine the distributional hypothesis with the idea of vectors to represent words!

Input: A Word–Document Matrix

	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10
against	0	0	0	1	0	0	3	2	3	0
age	0	0	0	1	0	3	1	0	4	0
agent	0	0	0	0	0	0	0	0	0	0
ages	0	0	0	0	0	2	0	0	0	0
ago	0	0	0	2	0	0	0	0	3	0
agree	0	1	0	0	0	0	0	0	0	0
ahead	0	0	0	1	0	0	0	0	0	0
ain’t	0	0	0	0	0	0	0	0	0	0
air	0	0	0	0	0	0	0	0	0	0
aka	0	0	0	1	0	0	0	0	0	0

Or...: A Word–Word Co-Occurrence Matrix

	against	age	agent	ages	ago	agree	ahead	ain't	air	aka	al
against	2003	90	39	20	88	57	33	15	58	22	24
age	90	1492	14	39	71	38	12	4	18	4	39
agent	39	14	507	2	21	5	10	3	9	8	25
ages	20	39	2	290	32	5	4	3	6	1	6
ago	88	71	21	32	1164	37	25	11	34	11	38
agree	57	38	5	5	37	627	12	2	16	19	14
ahead	33	12	10	4	25	12	429	4	12	10	7
ain't	15	4	3	3	11	2	4	166	0	3	3
air	58	18	9	6	34	16	12	0	746	5	11
aka	22	4	8	1	11	19	10	3	5	261	9
al	24	39	25	6	38	14	7	3	11	9	861

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty							
has							
brown							
hair							
and							
John							
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	1	1					
has							
brown							
hair							
and							
John							
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	**1**	**1**					
has	**1**	**1**	**1**				
brown							
hair							
and							
John							
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	**1**	**1**					
has	**1**	**1**	**1**				
brown		**1**	**1**	**1**			
hair							
and							
John							
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	1	1					
has	1	1	1				
brown		1	1	1			
hair			1	1		1	
and							
John							
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	1	1					
has	1	1	1				
brown		1	1	1			
hair			1	1	1		
and				1	1	1	
John							
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	**1**	**1**					
has	**1**	**1**	**1**				
brown		**1**	**1**	**1**			
hair			**1**	**1**	**1**		
and				**1**	**1**	**1**	
John	**1**				**1**	**1**	
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	1	1					
has	1	2	1			1	1
brown	1	1	1				
hair		1	1	1			
and			1	1	1		
John	1			1	1		
black							

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	1	1					
has	1	2	1			1	1
brown		1	1	1			
hair			1	1	1		
and				1	1	1	
John	1				1	1	
black	1			1			1

Or...: A Word–Word Co-Occurrence Matrix

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	1	1					
has	1	2	1			1	1
brown		1	1	1			
hair			1	2	1		1
and				1	1	1	
John	1				1	1	
black	1			1			1

Co-occurrence

How do we define “co-occurrence”? (What is the *context*?)

- Documents
- Sentences
- Tweets
- Windows
  - Size 1
  - Size 5
  - …
- …

Tip: Smaller contexts lead to more focus on syntax, larger contexts lead to more focus on semantics!

Measuring similarity

Measuring Similarity

Betty has brown hair and John has black hair.

	Betty	has	brown	hair	and	John	black
Betty	**1**	**1**					
has	**1**	**2**	**1**			**1**	**1**
brown		**1**	**1**	**1**			
hair			**1**	**2**	**1**		**1**
and				**1**	**1**	**1**	
John		**1**			**1**	**1**	
black		**1**		**1**			**1**

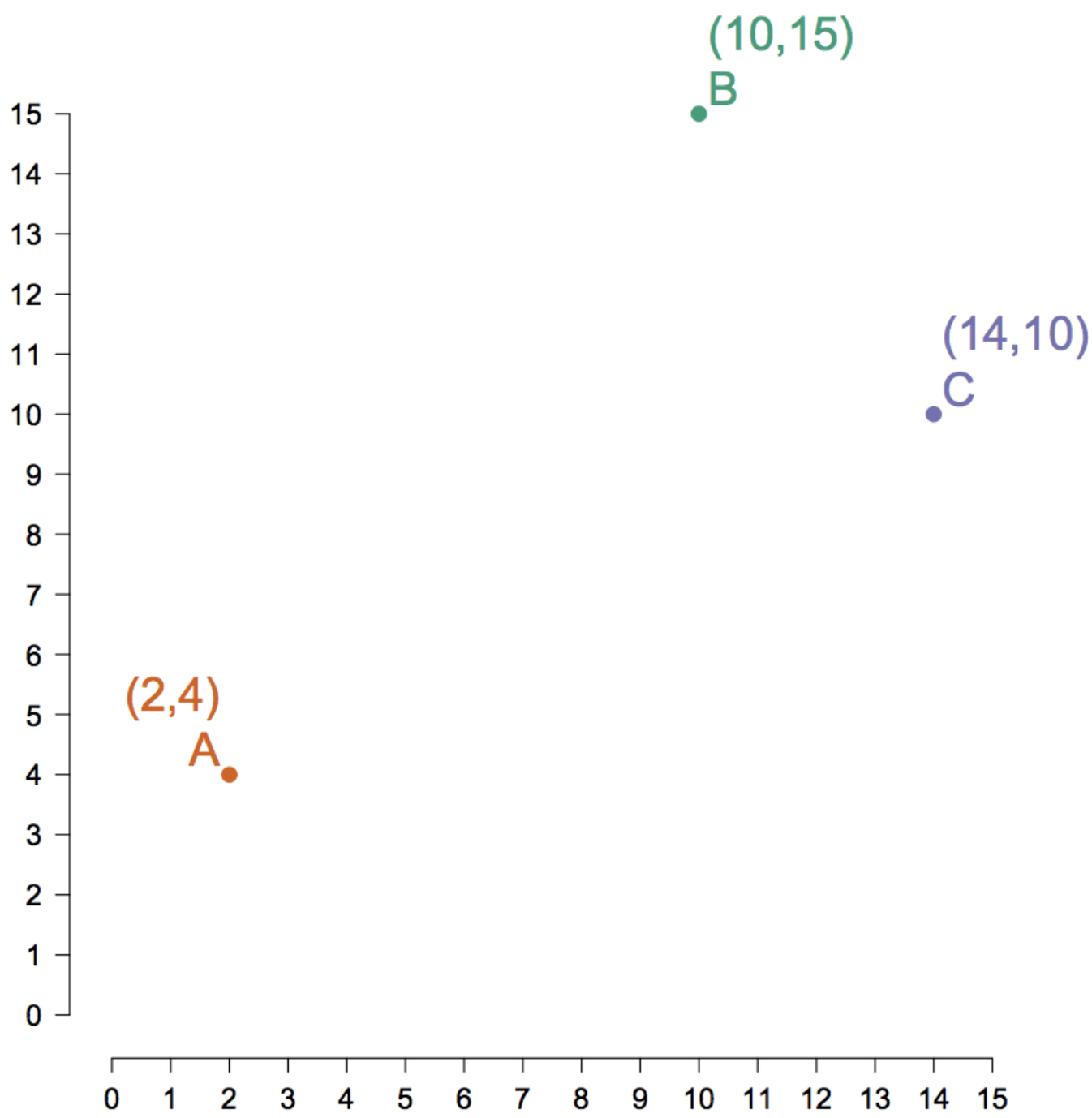
How similar are: *brown* and *black*? *brown* and *and*?

Measuring Similarity

	d_x	d_y
A	2	4
B	10	15
C	14	10

Measuring Similarity

	d_x	d_y
A	2	4
B	10	15
C	14	10



Euclidean Distance

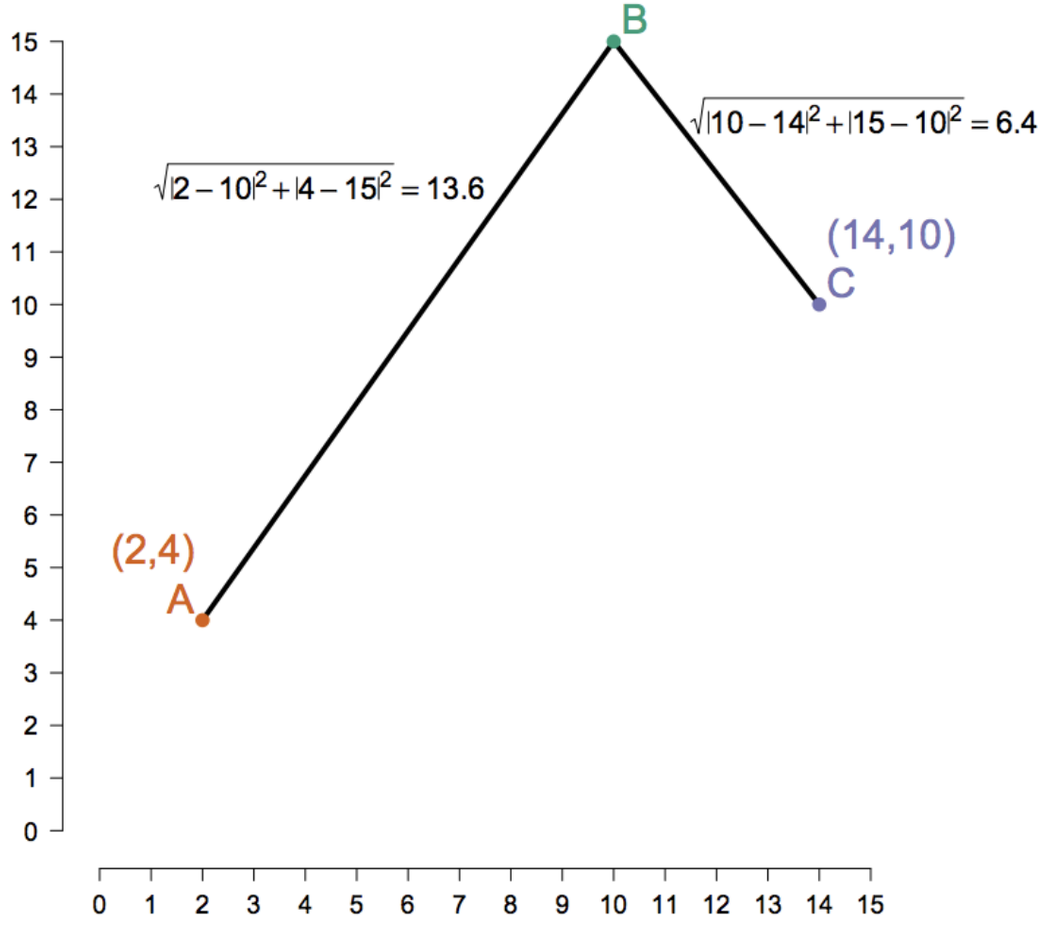
Definition

Between vectors u and v of dimension n:

$$\sqrt{\sum_{i=1}^{n} |u_i - v_i|^2}$$

(10,15)

	d_x	d_y
A	2	4
B	10	15
C	14	10



Length (L2) Normalization

Definition

Given a vector u of dimension n, the normalization of u is a vector $\hat{u}$ of dimension n obtained by dividing each element of u by $\|u\| = \sqrt{\sum_{i=1}^{n} u_i^2}$.

	d_x	d_y
A	2	4
B	10	15
C	14	10

L2 norm the rows

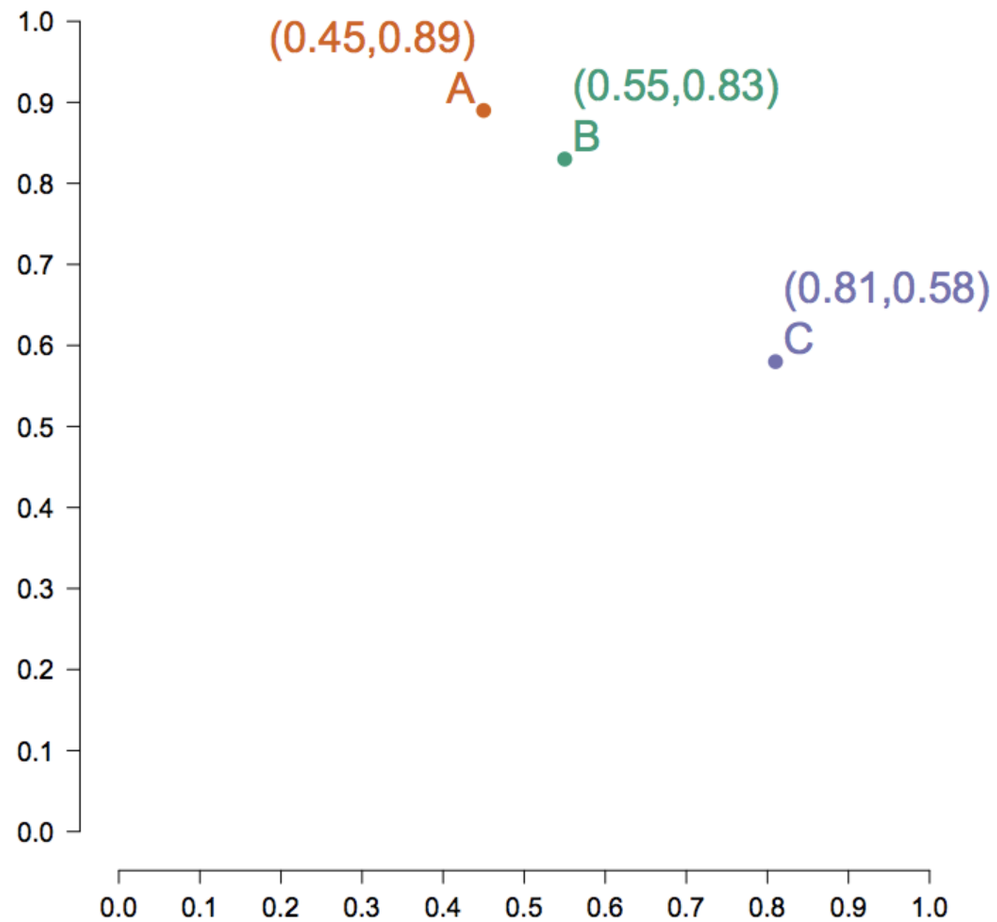
$\Rightarrow$

	d_x	d_y
A	0.45	0.89
B	0.55	0.83
C	0.81	0.58

Length (L2) Normalization

Definition

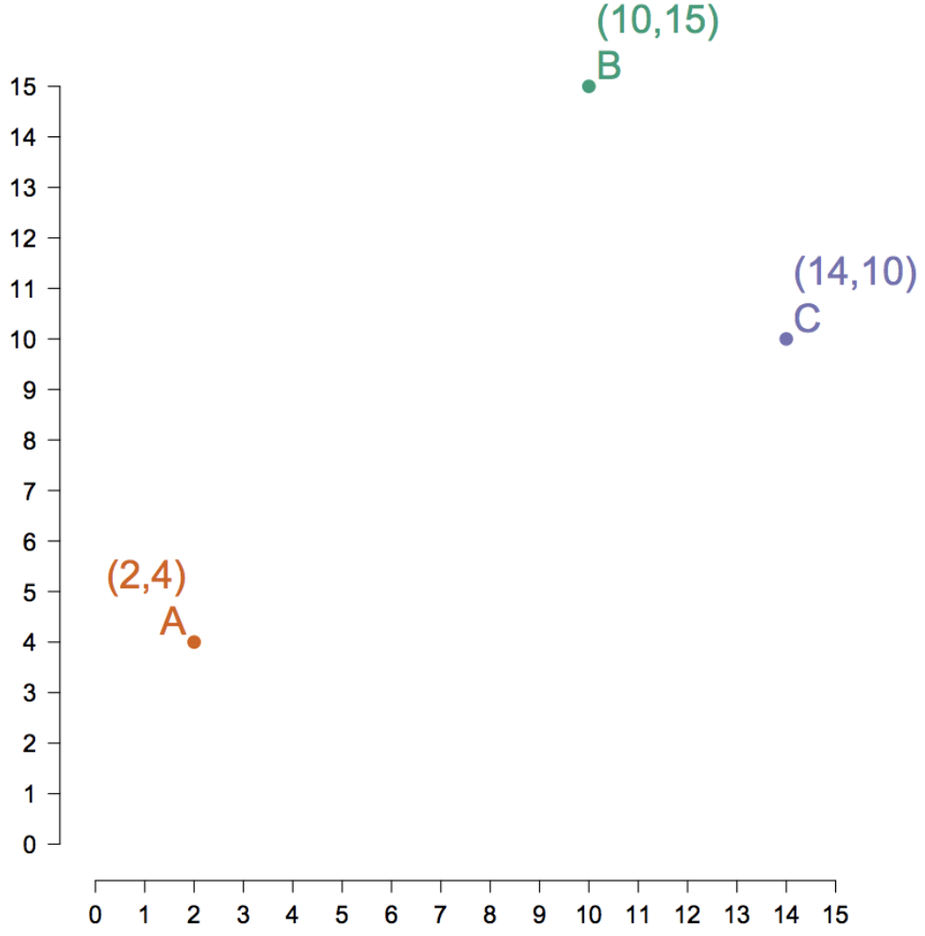
Given a vector u of dimension n, the normalization of u is a vector $\hat{u}$ of dimension n obtained by dividing each element of u by $\|u\| = \sqrt{\sum_{i=1}^{n} u_i^2}$.

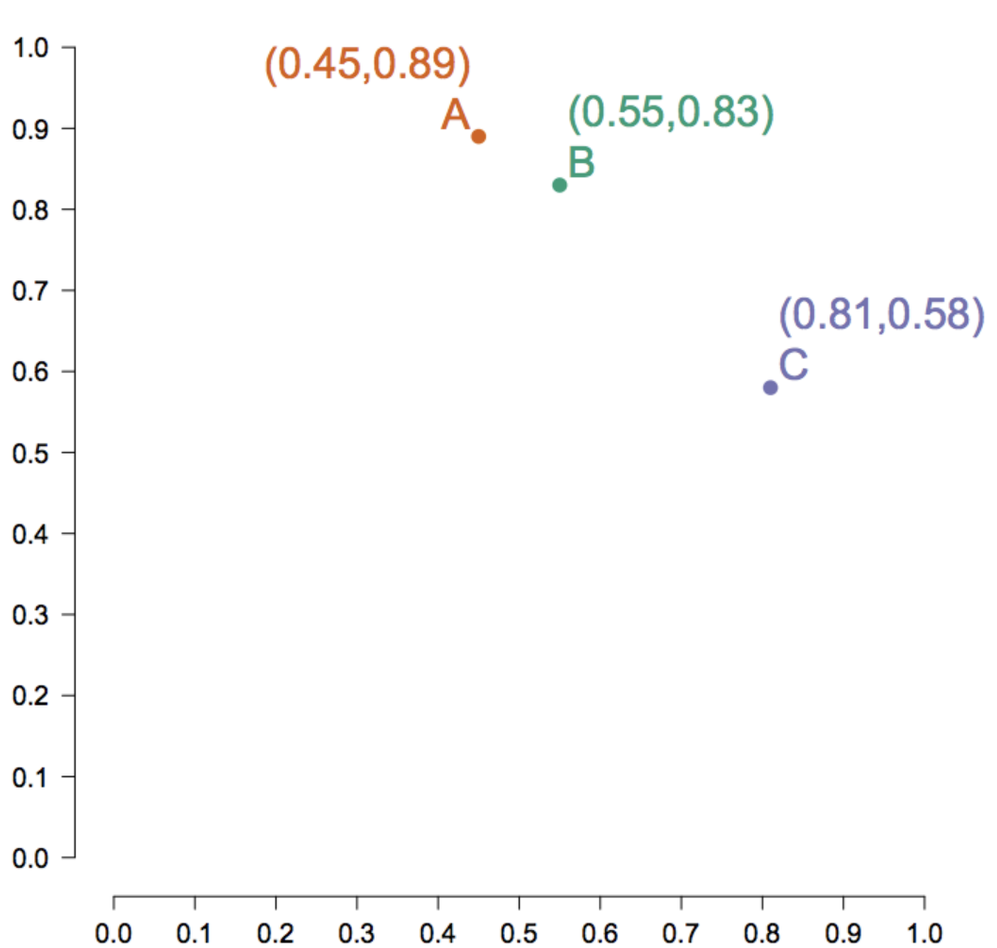


Length (L2) Normalization

Definition

Given a vector u of dimension n, the normalization of u is a vector $\hat{u}$ of dimension n obtained by dividing each element of u by $\|u\| = \sqrt{\sum_{i=1}^{n} u_i^2}$.



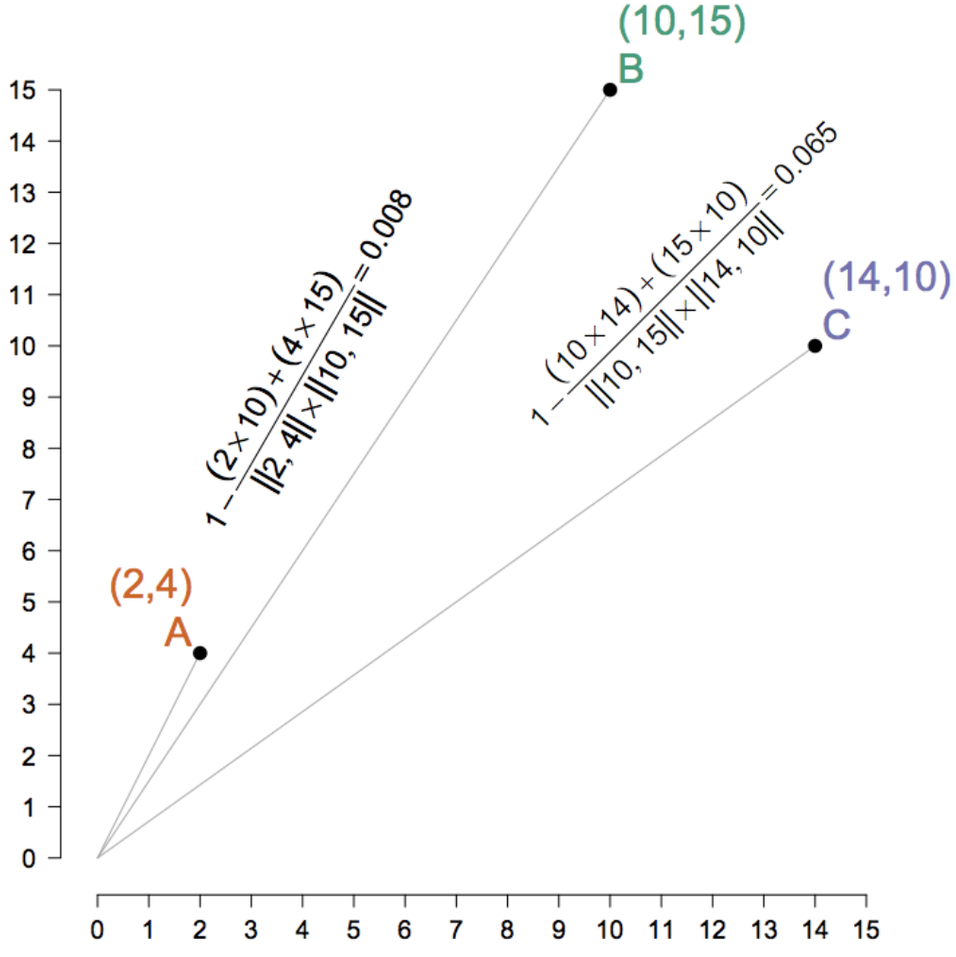


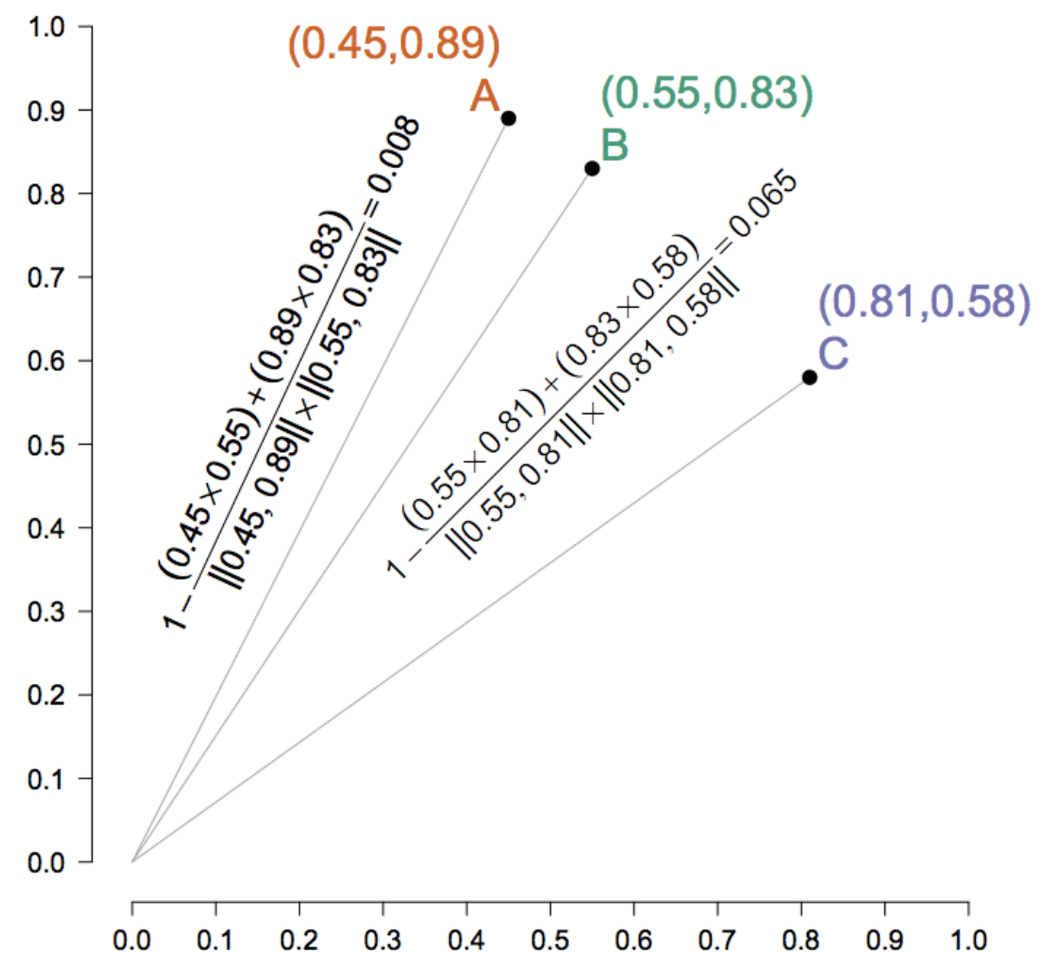
Cosine Distance

Definition (Cosine distance)

Between vectors u and v of dimension n:

$$1 - \frac{\sum_{i=1}^{n} u_i \times v_i}{\sqrt{\sum_{i=1}^{n} u_i^2} \times \sqrt{\sum_{i=1}^{n} v_i^2}}$$





Picking a Distance Metric

	d_x	d_y
A	2	4
B	10	15
C	14	10

$$\|A\| = 4.47$$
$$\|B\| = 18.03$$
$$\|C\| = 17.20$$

	A and B closer than B and C?
Euclidean distance	No
Cosine distance	Yes

Word2vec (Mikolov et al., 2013)

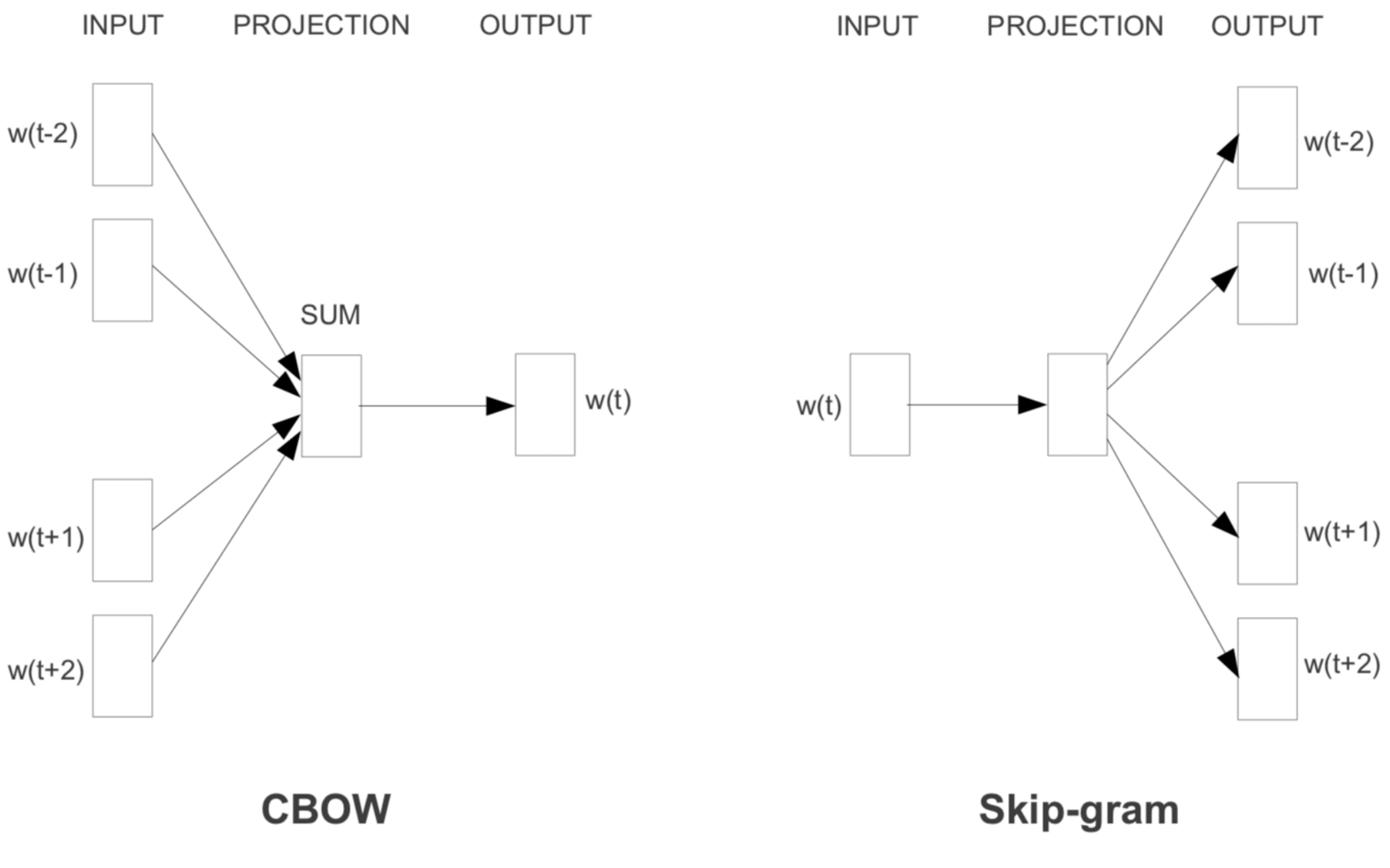


Figure from Mikolov et al. (2013)

Word Embeddings

- **Apple** will release a new iPhone in September.
- This **apple** tastes better than it looks.

Word Embeddings

- **Apple** will release a new iPhone in September.
- This **apple** tastes better than it looks.

Contextualized word embeddings!

Contextualized Word Embeddings

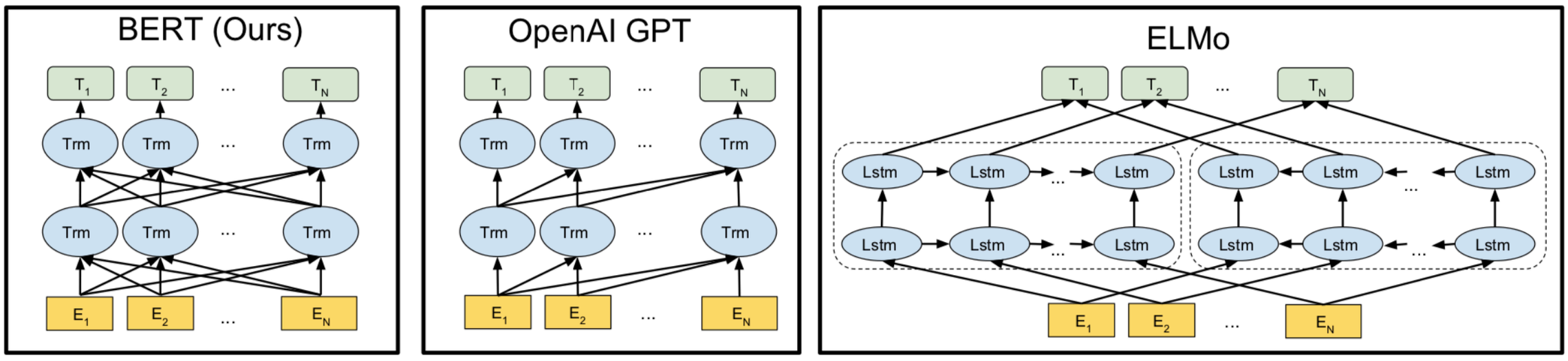


Figure from Devlin et al. (2018)

ELMo (Peters et al., 2018), GPT (Radford et al., 2018)

The Transformer Architecture

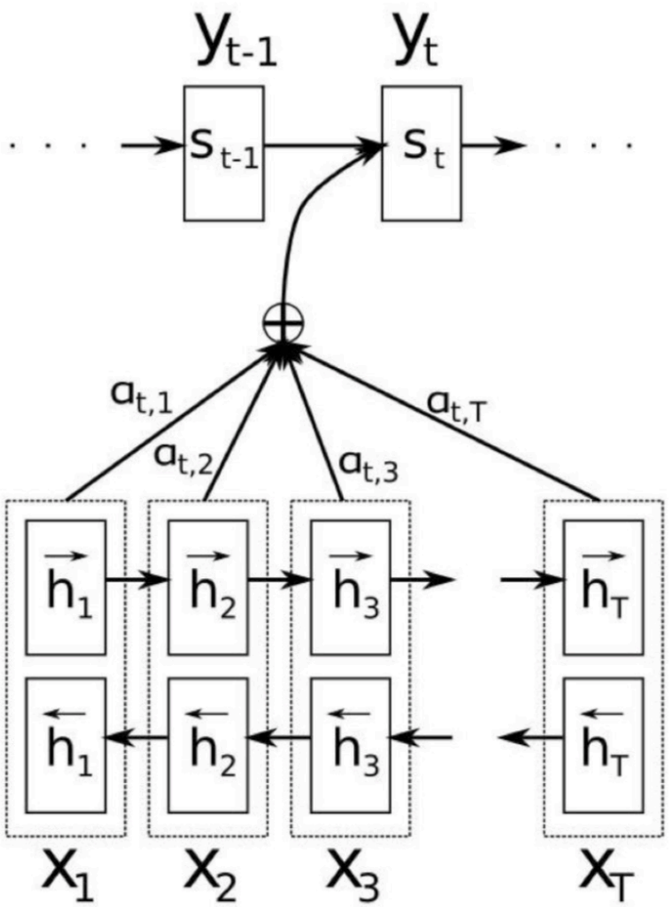
Attention (Bahdanau et al., 2015)

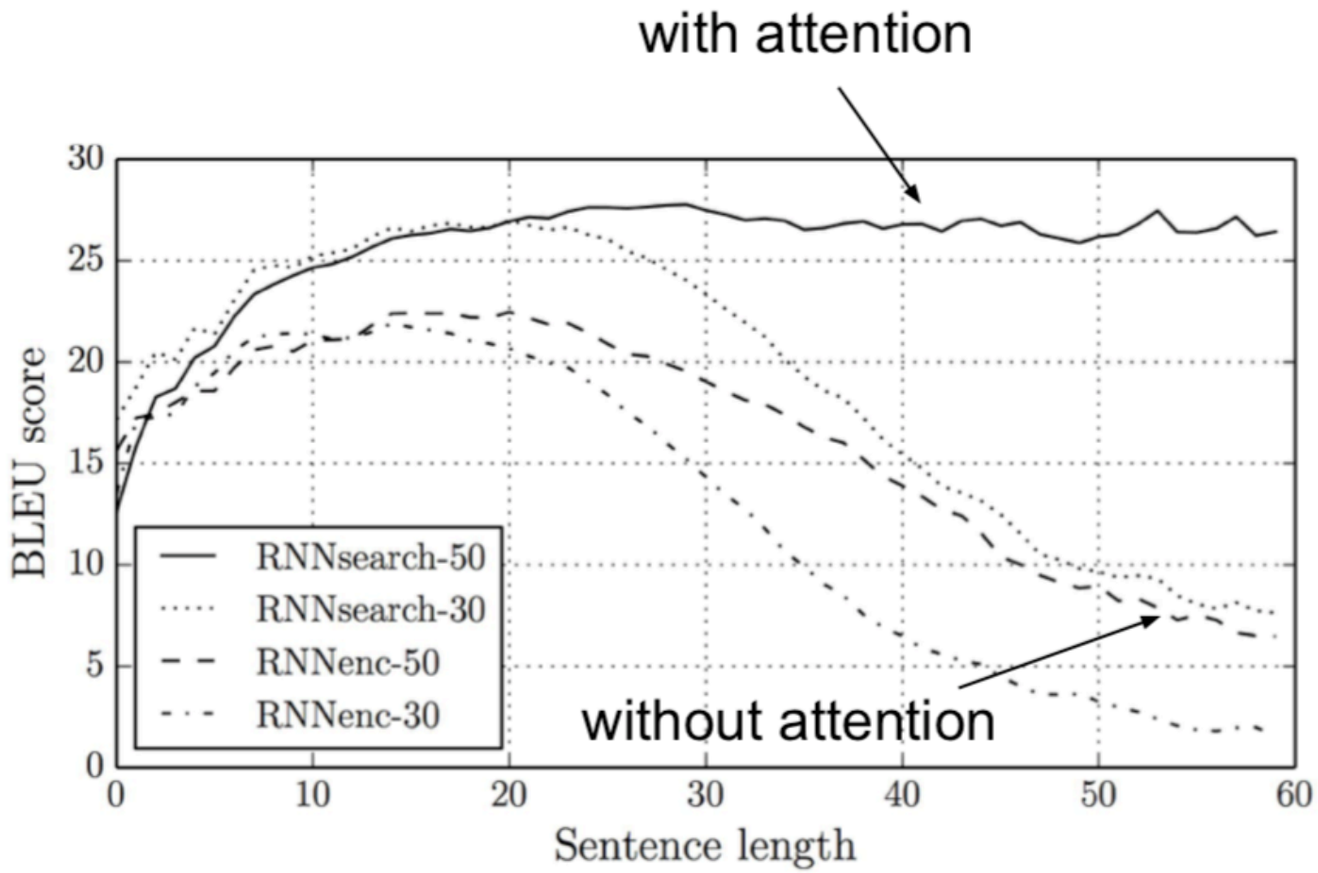
- We want to compute how important each input is for a given output

Attention (Bahdanau et al., 2015)

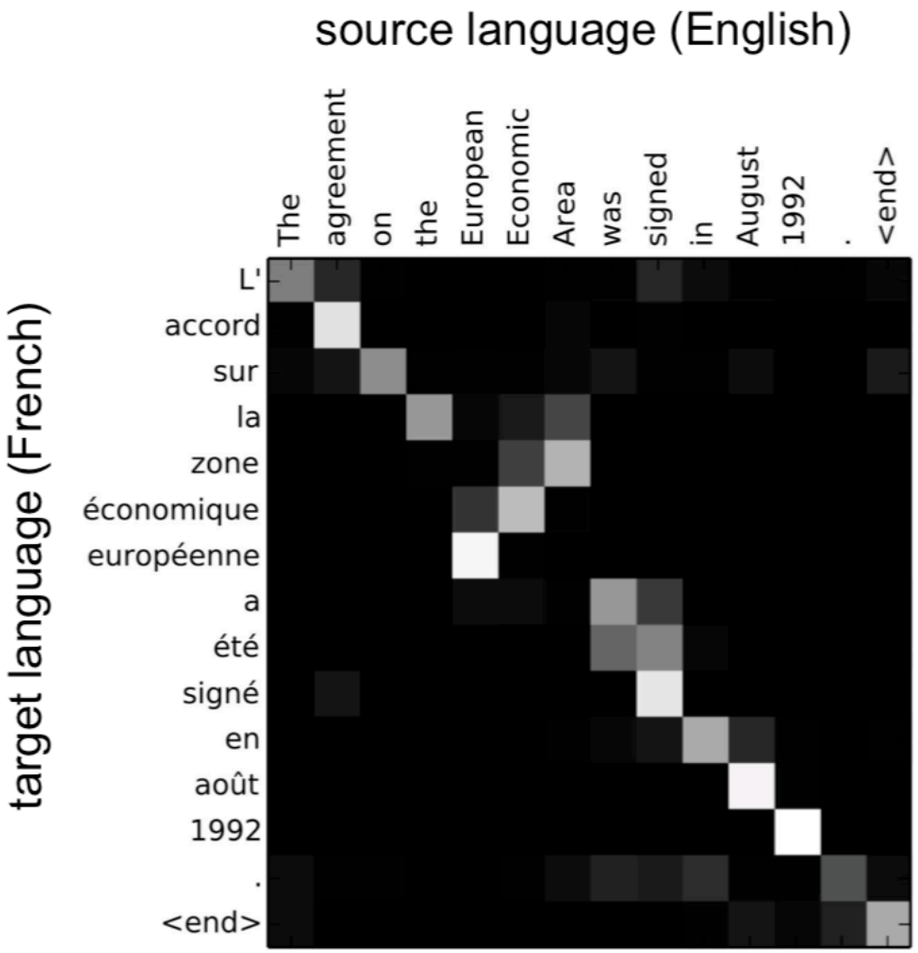
- We want to compute how important each input is for a given output
- “Attention” corresponds to an automatically computed weight for each input (at each time step in sequence-to-sequence models)
  - We have seen this for sequence-to-sequence models already!

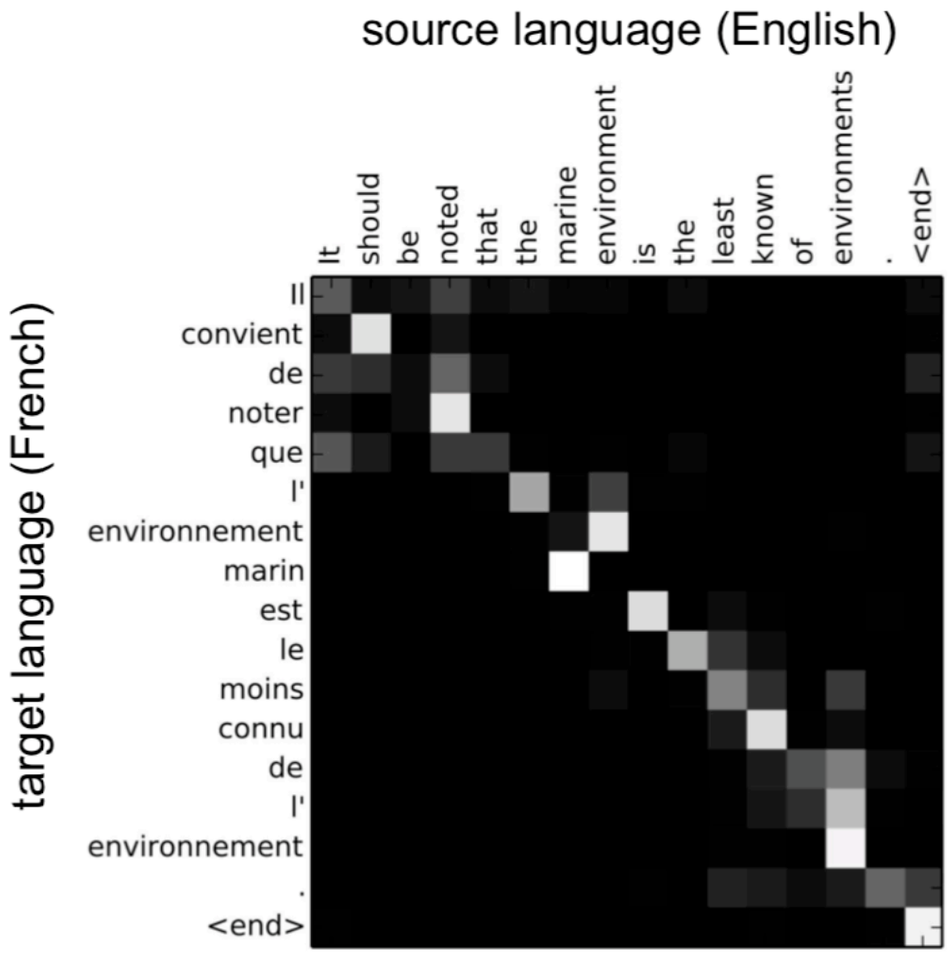
Attention (Bahdanau et al., 2015)





Attention (Bahdanau et al., 2015)





“Attention is all you need”
(Vaswani et al., 2017)

Problem with RNN seq2seq architecture:

- Not parallelizable! (Why?)

“Attention is all you need” (Vaswani et al., 2017)

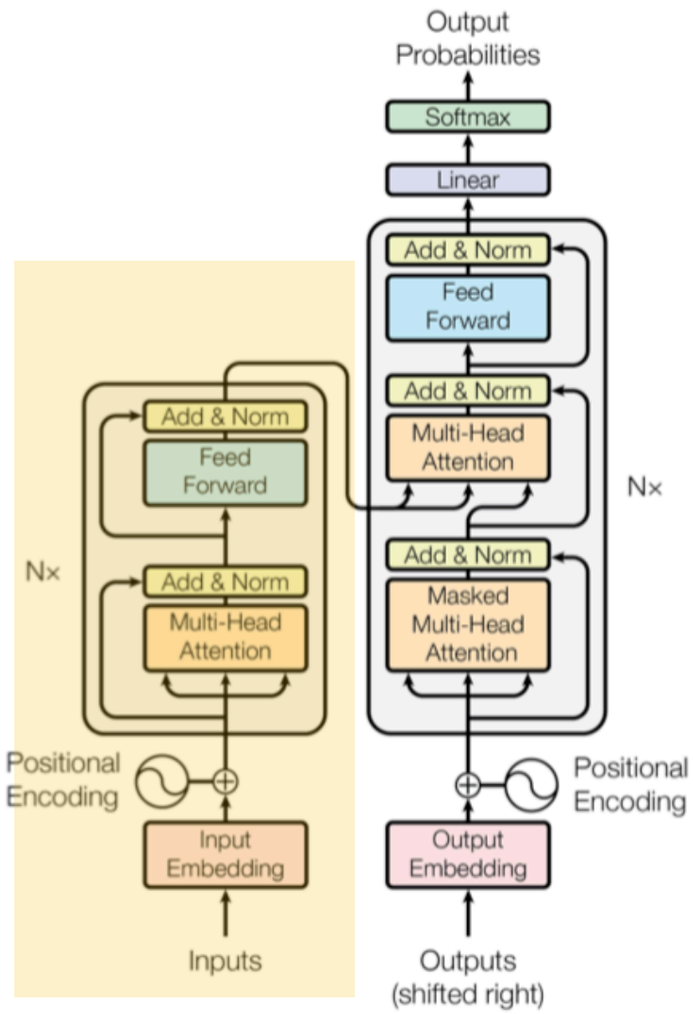
Problem with RNN seq2seq architecture:

- Not parallelizable! (Why?)

Suggested solution:

- Transformer architecture: parallelizable!
- Can reach state-of-the-art performance quicker

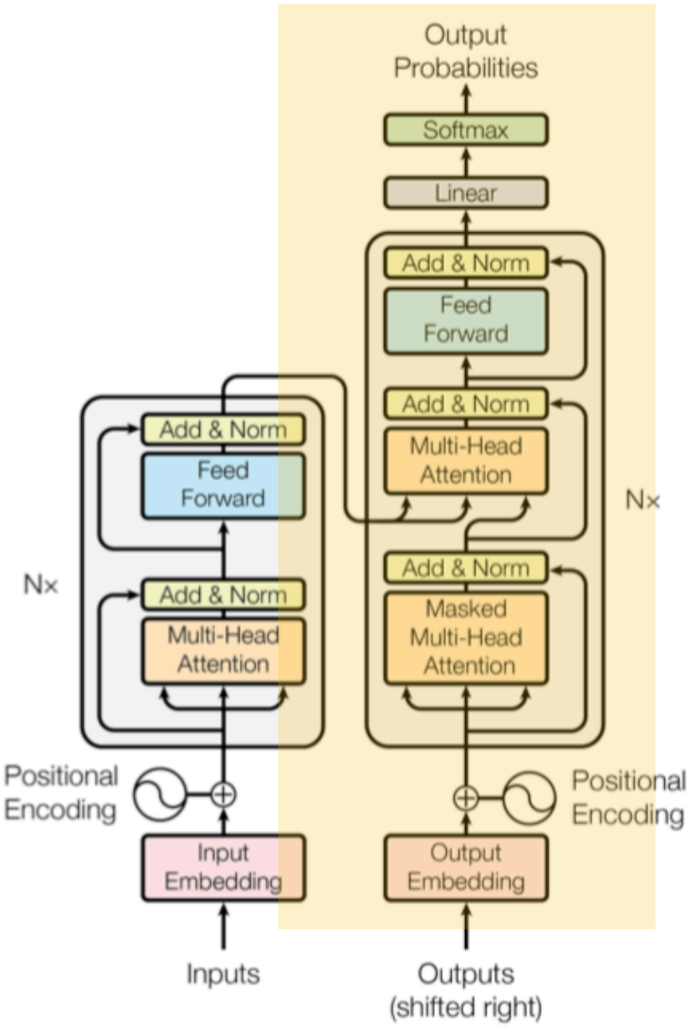
“Attention is all you need” (Vaswani et al., 2017)



Encoder:

- Consists of N=6 layers
- Each layer consists of 2 sub-layers
  - Self-attention (access to all positions in the encoder)
  - Feed-forward network
- Inputs:
  - Current embedding
  - Position embedding

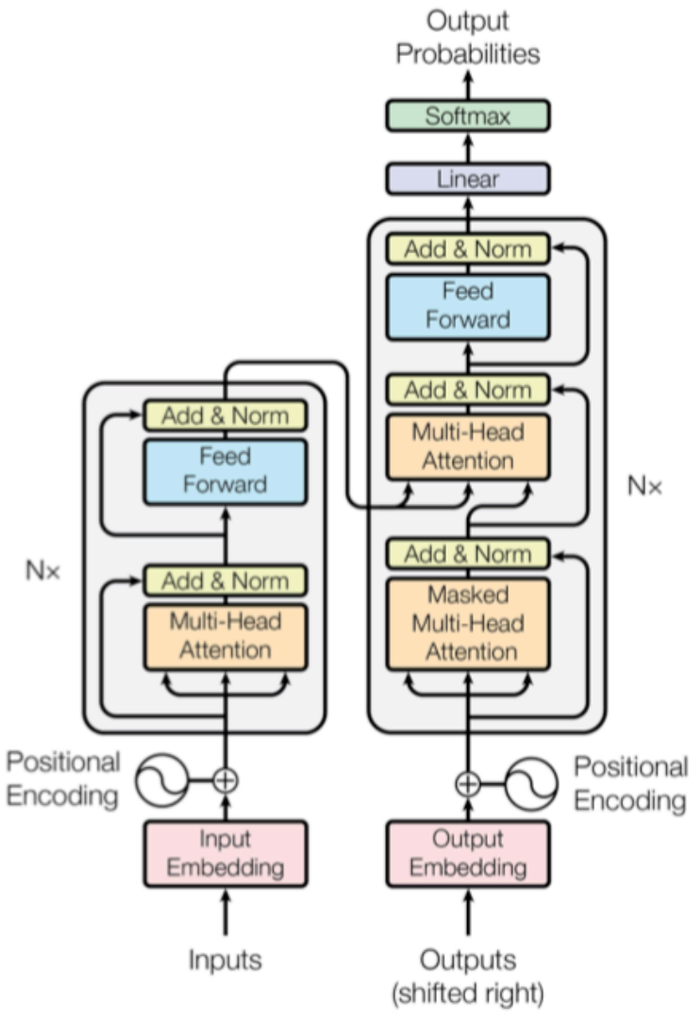
“Attention is all you need” (Vaswani et al., 2017)



Decoder:

- Also consists of N=6 layers
- Each layer consists of **3 sub-layers**
  - Self-attention (access to all previous positions in the decoder)
  - **Encoder-decoder attention**
  - Feed-forward network
- Inputs:
  - Current embedding
  - Position embedding

“Attention is all you need” (Vaswani et al., 2017)



Training:

- Can be done in parallel for all output positions

How about testing?

“Attention is all you need” (Vaswani et al., 2017)

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [15]	23.75			
Deep-Att + PosUnk [32]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [31]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [8]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [26]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [32]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [31]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [8]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.0**	$2.3 \cdot 10^{19}$	

“Attention is all you need” (Vaswani et al., 2017)

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [15]	23.75			
Deep-Att + PosUnk [32]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [31]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [8]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [26]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [32]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [31]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [8]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.0**	$2.3 \cdot 10^{19}$	

BERT and Co.

BERT (Devlin et al., 2018)

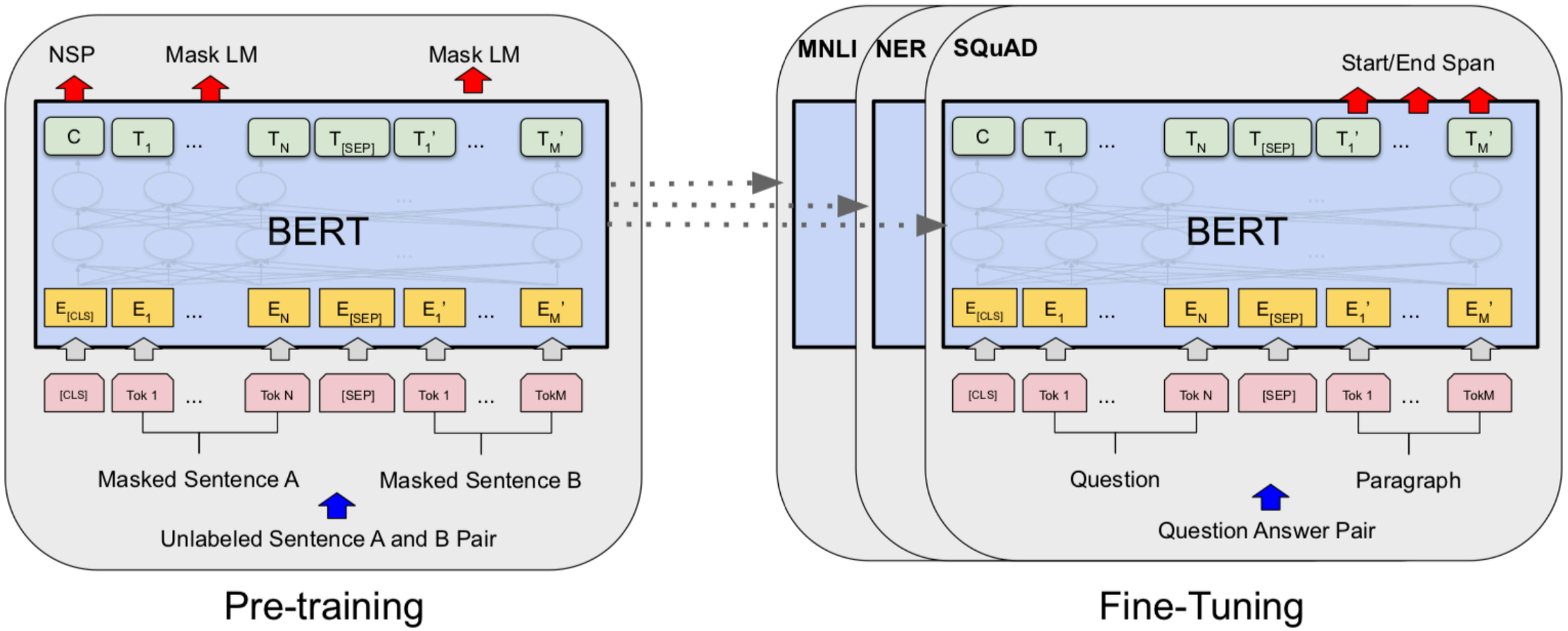


Figure from Devlin et al. (2018)

BERT (Devlin et al., 2018)

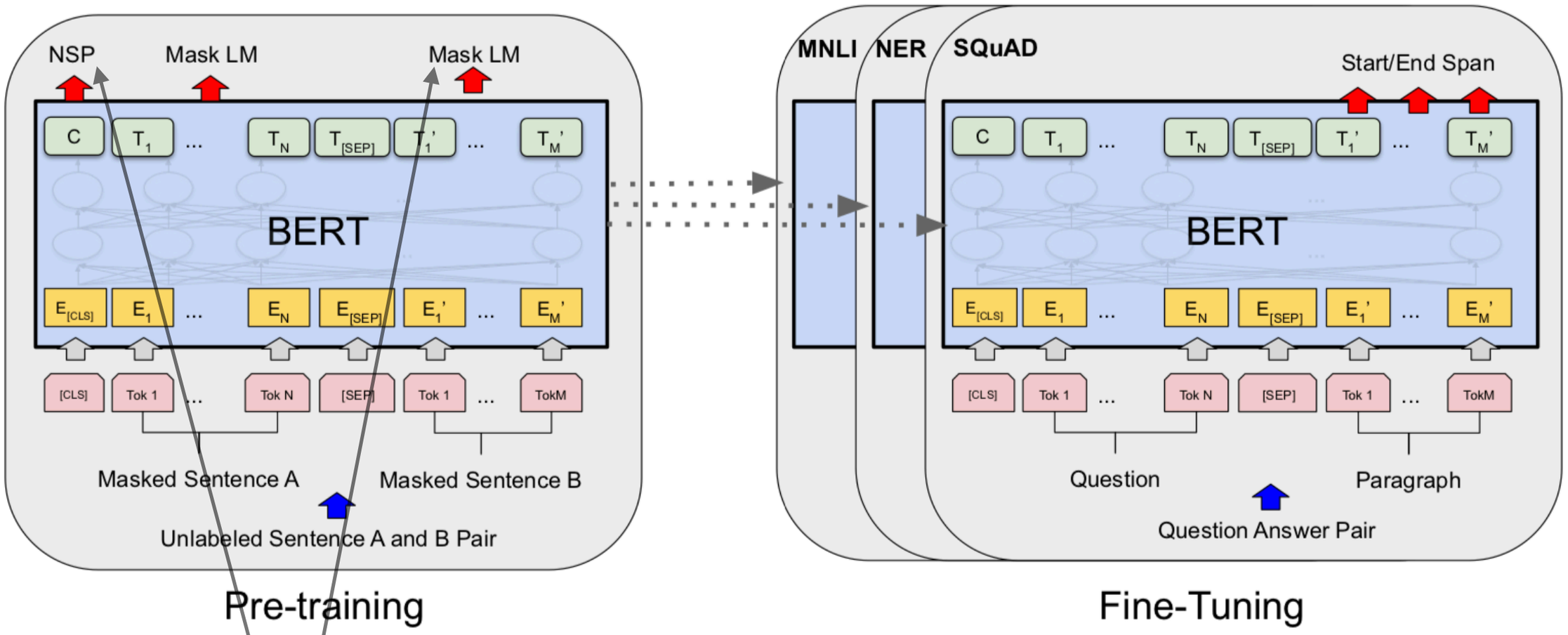


Figure from Devlin et al. (2018)

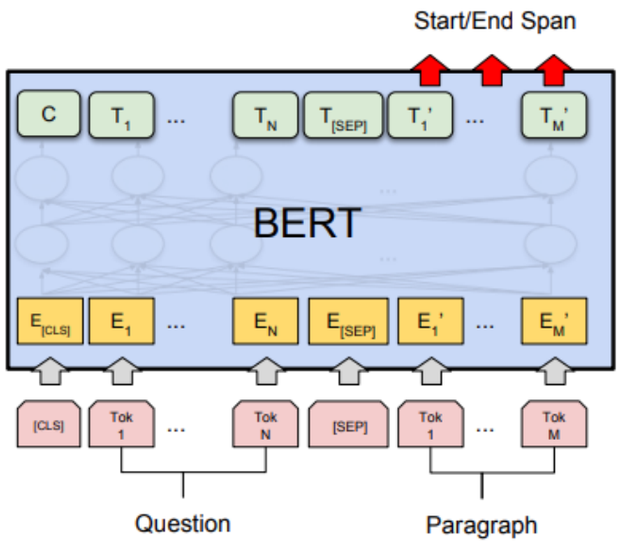
Training objectives

The BERT Model

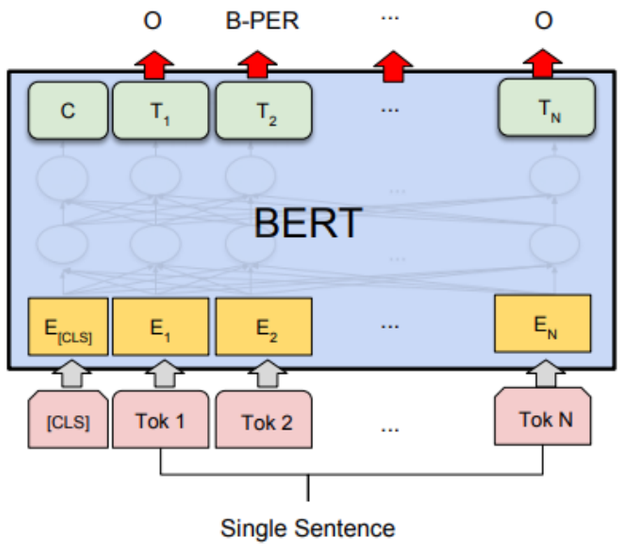
- Same basic idea as OpenAI, but many small differences, including:
  - Two different unlabeled data tasks in place of language modeling.
    - Neither requires 'predicting the future', so we can use an encoder-style Transformer rather than decoder-style.
- Very big (24 layers, >300M params).

The BERT Model

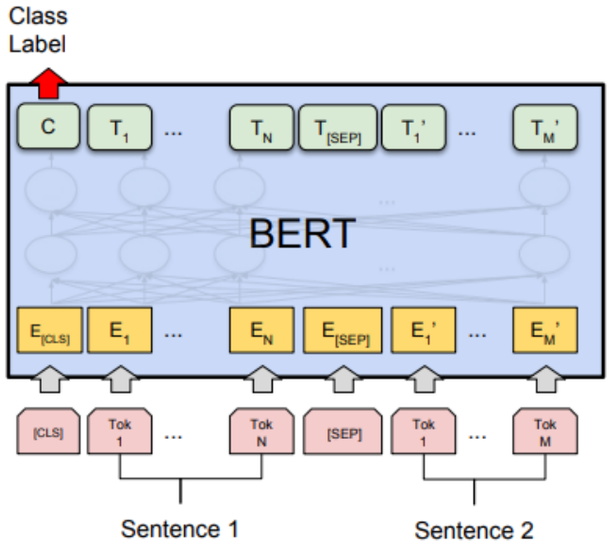
- For downstream tasks, an additional classification layer is added
  - The original output layer is discarded
- On top of what exactly we add the layer depends on the task



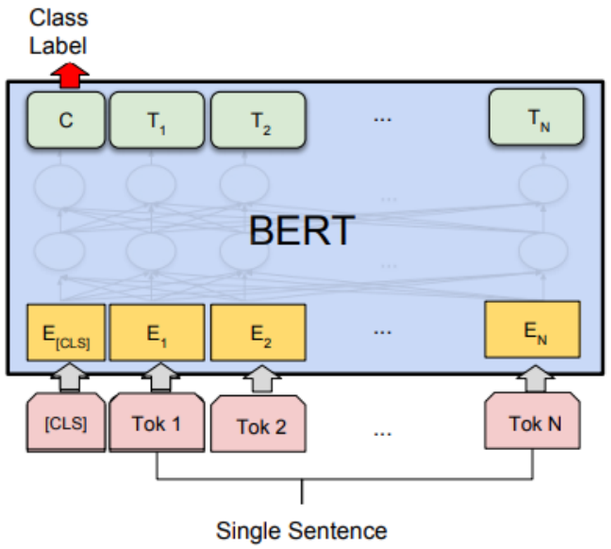
(c) Question Answering Tasks:
SQuAD v1.1



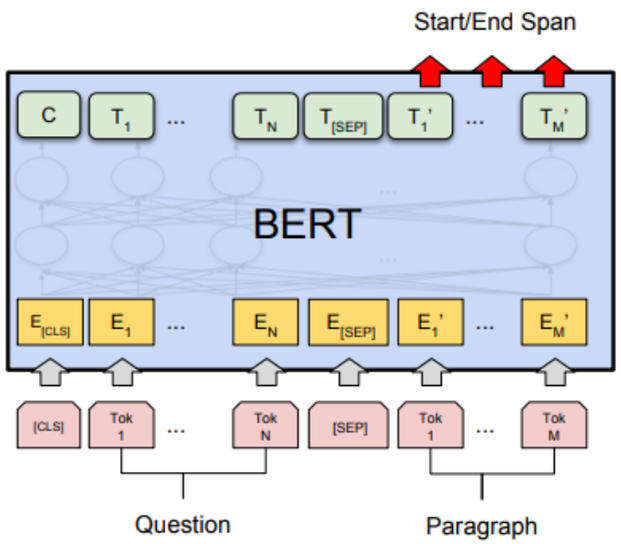
(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER



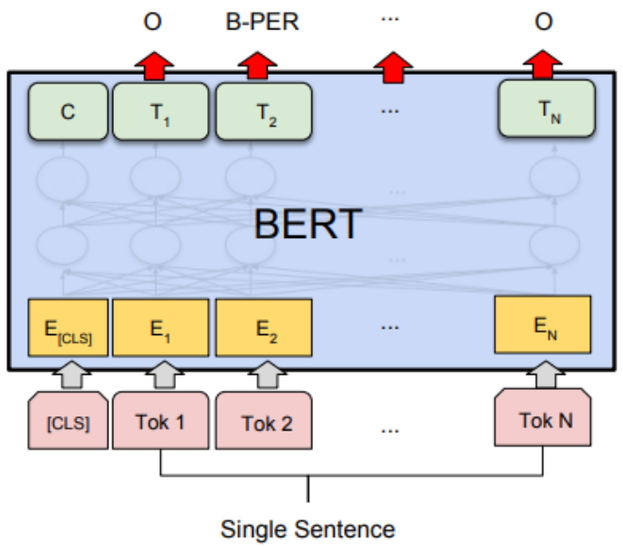
(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG



(b) Single Sentence Classification Tasks: SST-2, CoLA

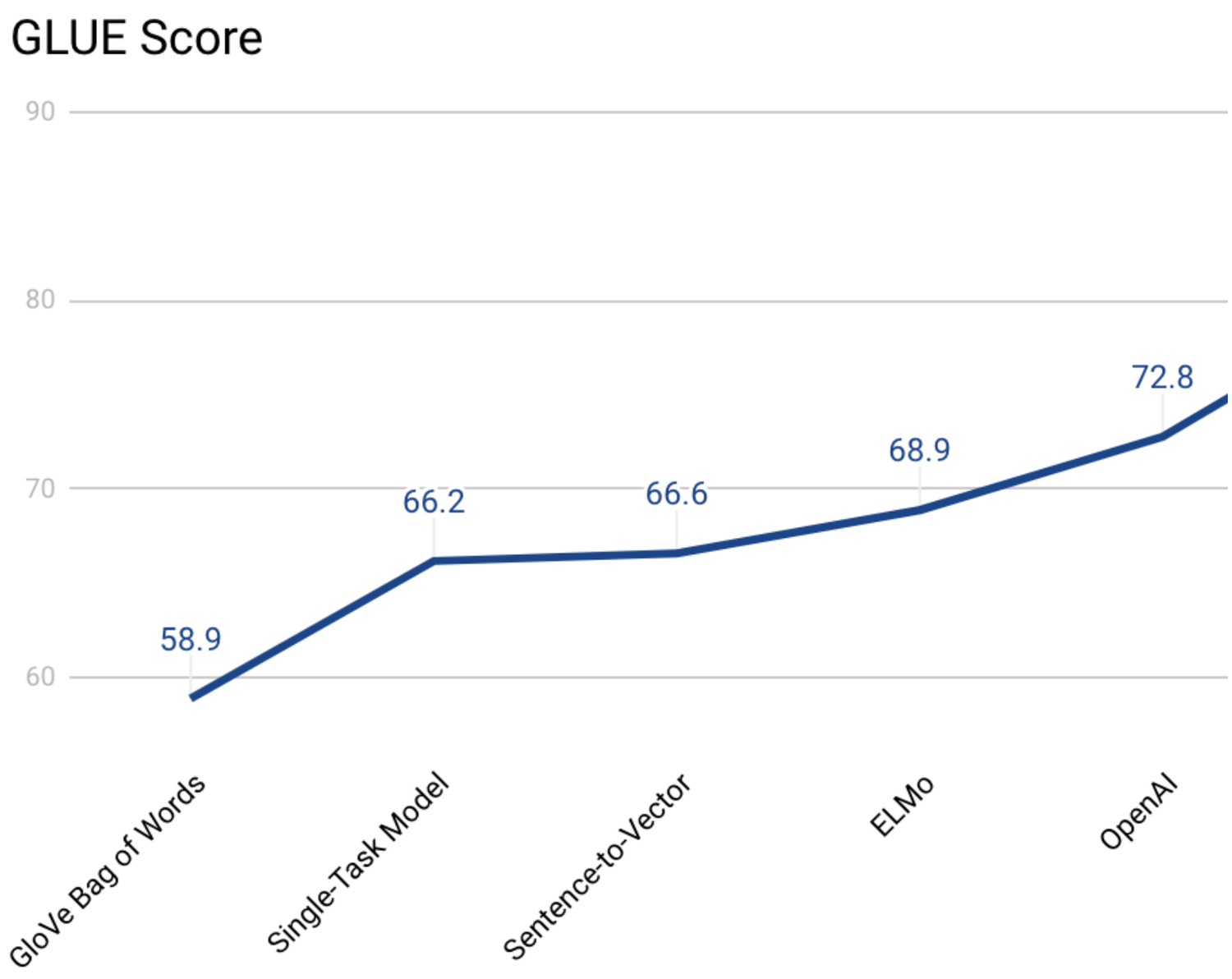


(c) Question Answering Tasks: SQuAD v1.1

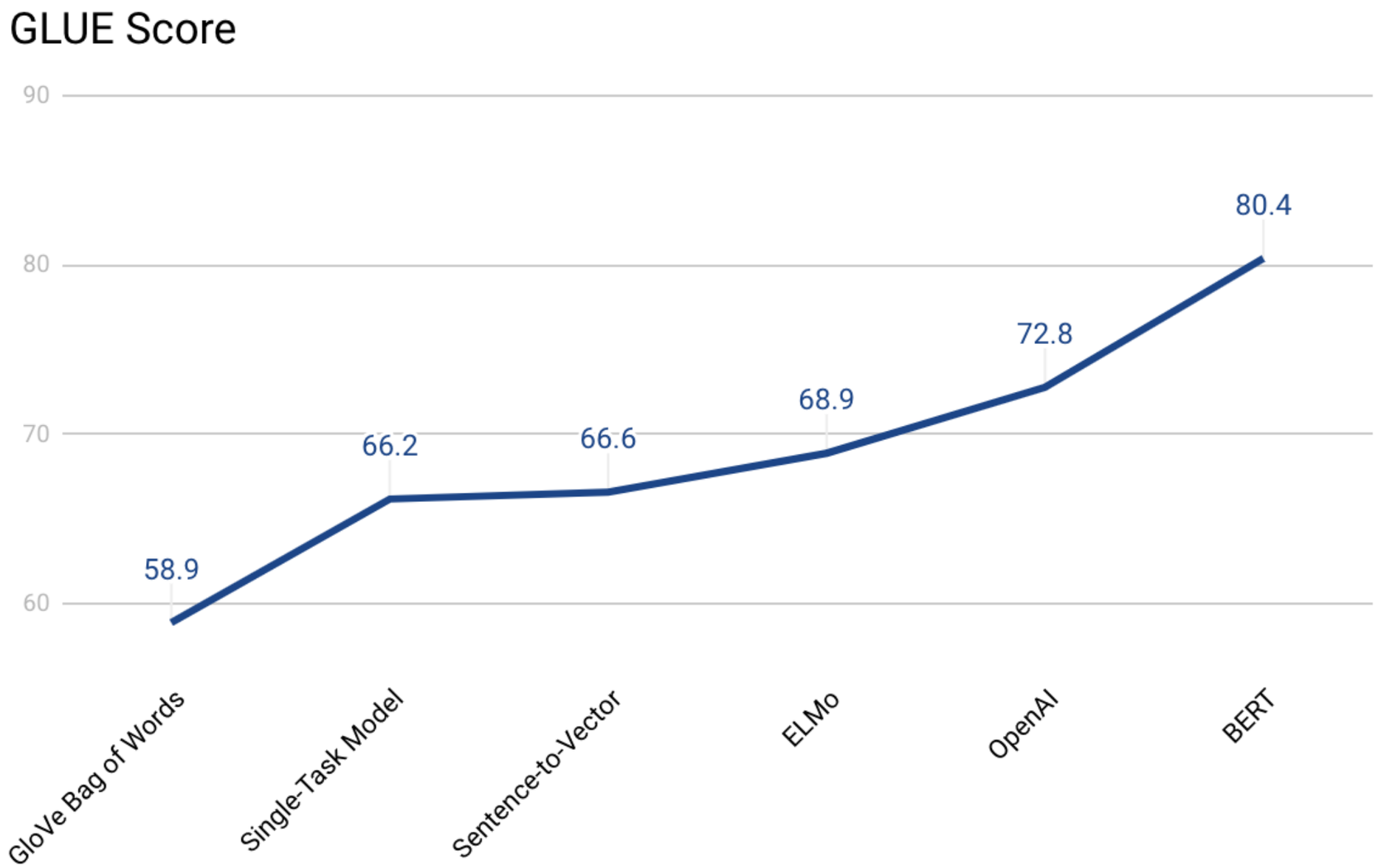


(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

What Methods Work?



What Methods Work?



After BERT…

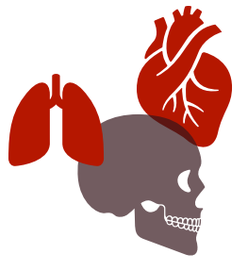
- Improved versions of BERT are constantly being proposed
- Keep an eye out for that
- Make sure you know the state of the art for your task

Adaptation

When?

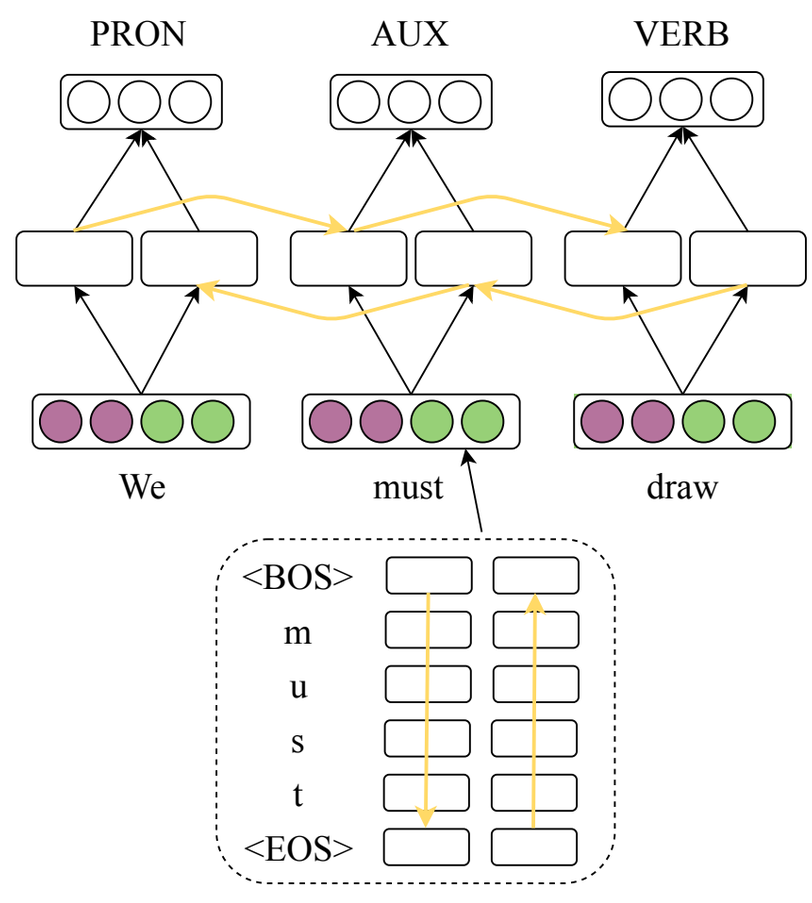
- Target task is different from source task
- Target domain is different from source domain (“domain adaptation”)

- Target language is different from source language



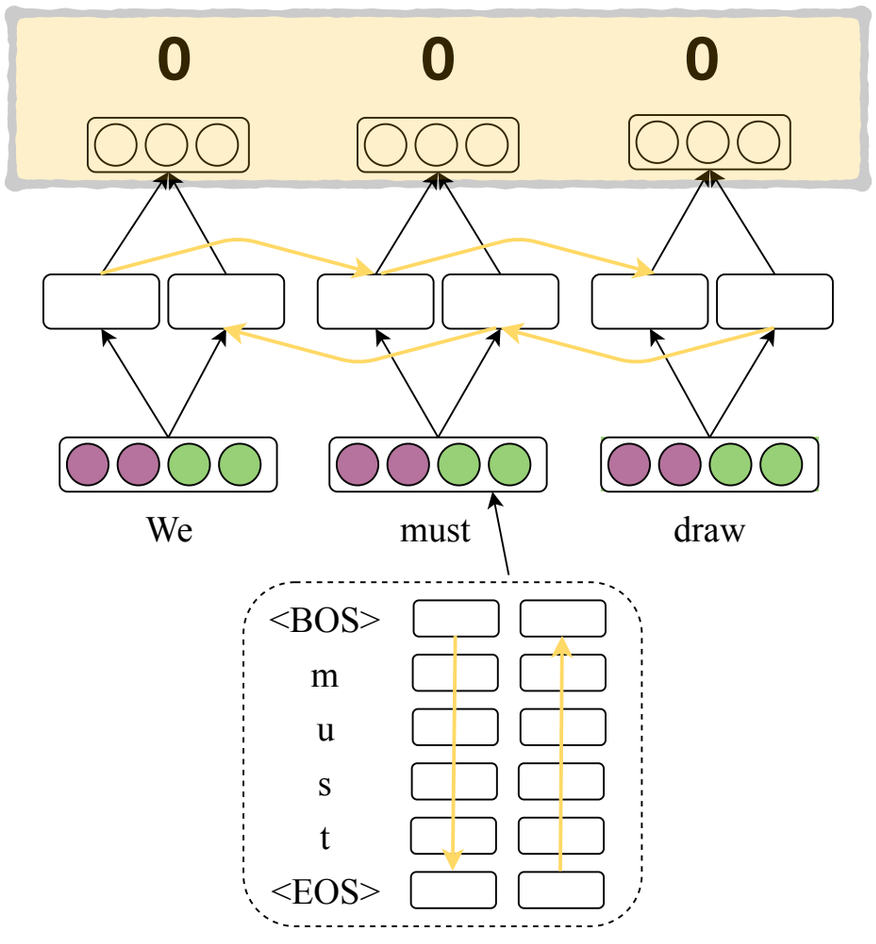
Design Decisions

- Sharing Information
  - Which parameters should be updated?
  - How about unfreezing?



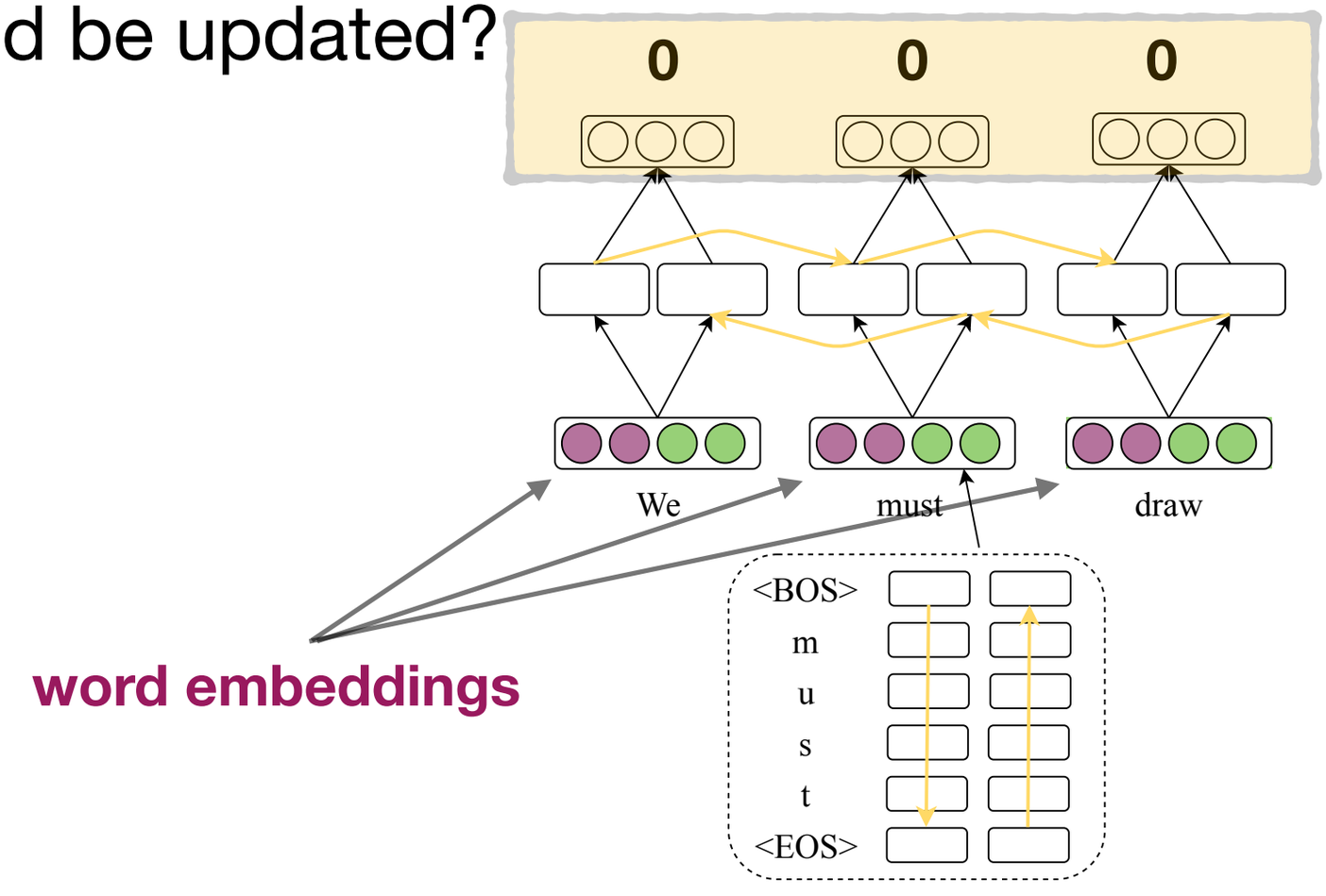
Design Decisions

- Sharing Information
  - Which parameters should be updated?
  - How about unfreezing?



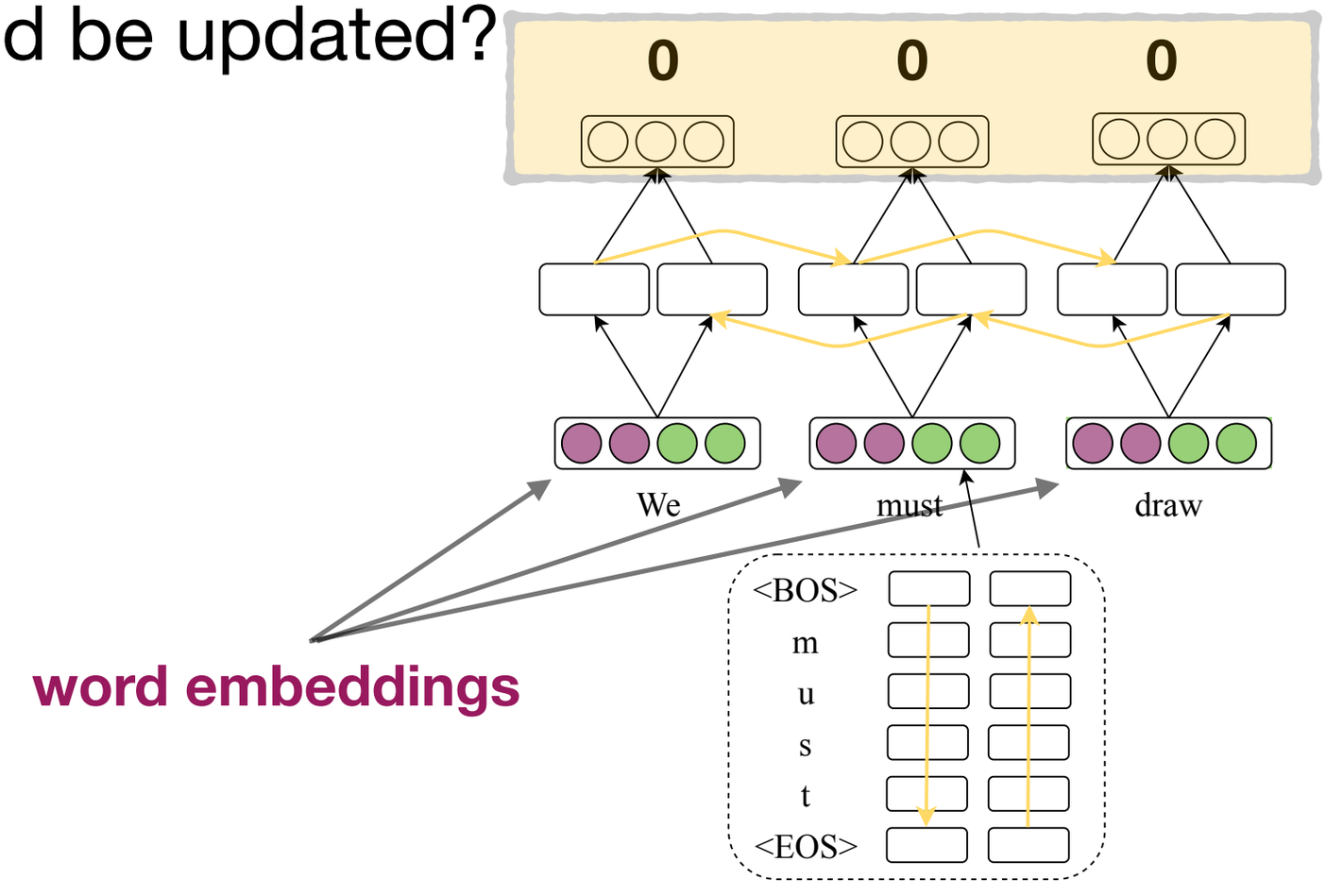
Design Decisions

- Sharing Information
  - Which parameters should be updated?
  - How about unfreezing?



Design Decisions

- Sharing Information
  - Which parameters should be updated?
  - How about unfreezing?
- Training Regime
  - Learning rate?
  - How many epochs?



Target Tasks

- Commonly tasks with small training sets
- Related to a source task

- SuperGLUE (Wang et al., 2019) recently popular benchmark

SuperGLUE (Wang et al., 2019) Tasks

- **BoolQ**: Yes/no questions
- **WiC**: Word-in-context, i.e., word sense disambiguation
- **RTE**: Recognizing textual entailment
- …

Multi-task Training

Multi-Task Training

Given: Main task data and auxiliary task data

- Decide which model parameters to share

- Train model on both data sets together:

$$\mathscr{L}(\boldsymbol{\theta}) = \alpha \mathscr{L}_{\textbf{main}}(\boldsymbol{\theta}) + \beta \mathscr{L}_{\textbf{aux}}(\boldsymbol{\theta})$$

Multi-Task Training

Given: Main task data and auxiliary task data

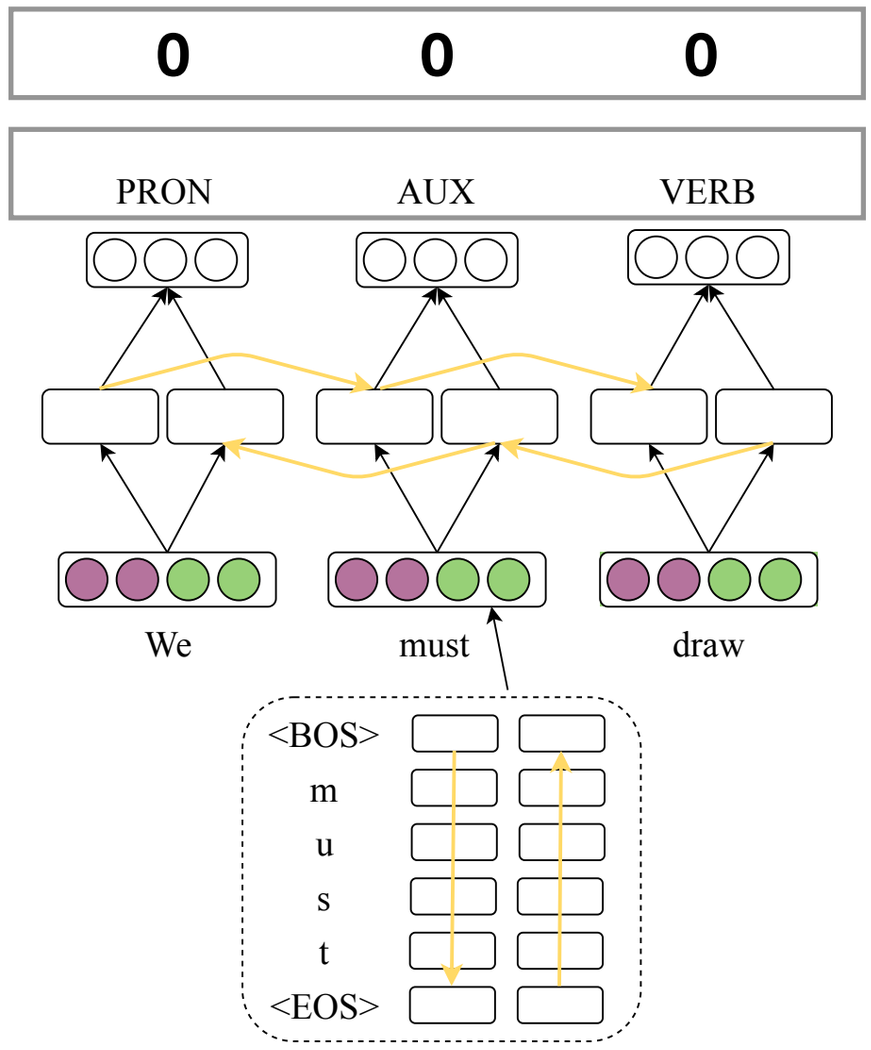
- Decide which model parameters to share
- Train model on both data sets together:

$$\mathscr{L}(\boldsymbol{\theta}) = \alpha \mathscr{L}_{\textbf{main}}(\boldsymbol{\theta}) + \beta \mathscr{L}_{\textbf{aux}}(\boldsymbol{\theta})$$

Multi-task training: two tasks at the same time
Pretraining + adaptation: one task after the other

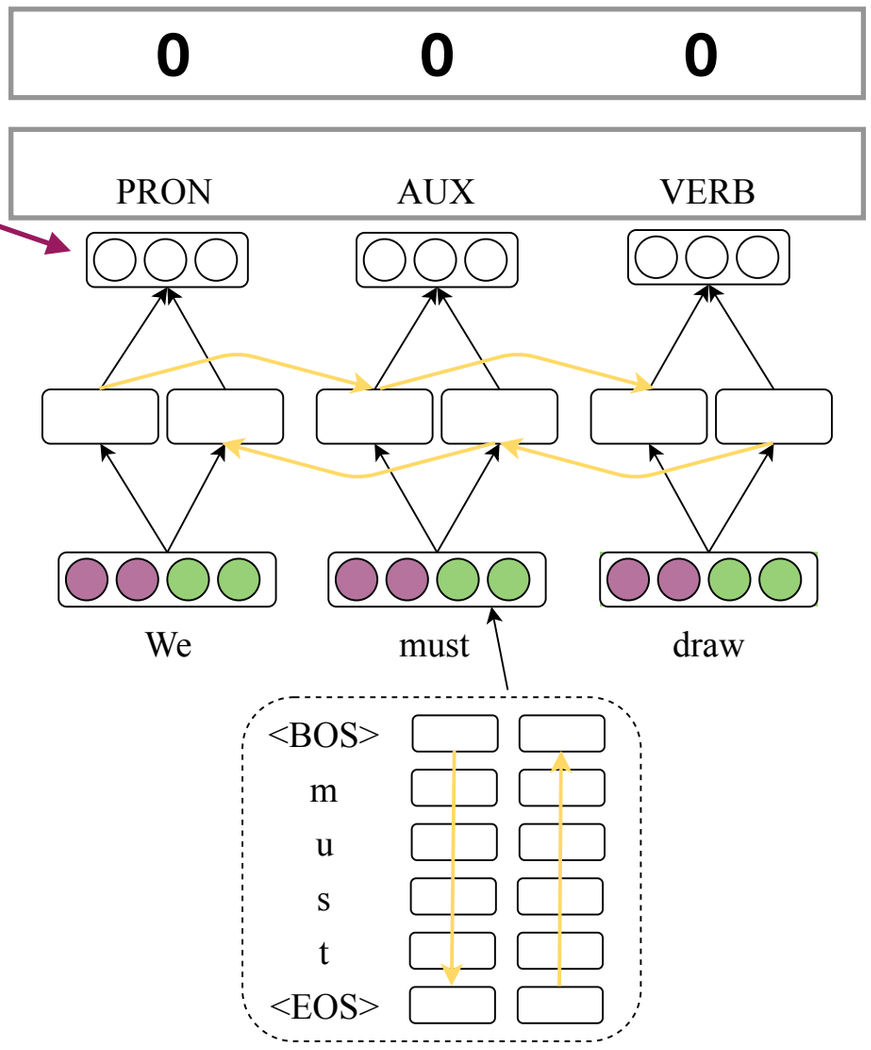
Again: Design Decisions

- Sharing Information
  - Which parameters should be shared?
- Training Regime
  - How should losses be weighted?
  - Upsampling/downsampling of data?



Again: Design Decisions

- Sharing Information
  - Which parameters should be shared?
- Training Regime
  - How should losses be weighted?
  - Upsampling/downsampling of data?



Target Tasks

- Commonly tasks with small training sets
- Related to a source task

Target Tasks

- Commonly tasks with small training sets
- Related to a source task

This is the same as for pretraining + adaptation!

Which Source Tasks for Which Target Tasks?

	CCG	CHU	COM	FNT	POS	HYP	KEY	MWE	SEM	STR
CCG		1.4	0.45	0.58	1.8	0.24	0.3	0.45	1.4	0.84
CHU	-0.052		-0.15	-0.12	-0.45	-0.5	-0.22	-0.27	-0.099	-0.32
COM	-5	1.3		1.3	-1.4	-2.4	-4.8	0.82	-3	-0.63
FNT	-5.8	-1	-6.1		-9.4	-5.7	-3.6	-9.4	-3	-0.68
POS	4.9	2.9	1.9	0.9		-0.85	-0.26	1.3	3.4	2.9
HYP	12	4	-11	9.2	22		1.5	-7.7	23	8.1
KEY	5.7	3.2	-1	-0.43	-1.3	-2.6		-4.7	0.59	0.69
MWE	18	20	7.4	5.5	1.6	-3.8	-5.8		16	8.6
SEM	-5	-0.76	-1.2	-0.81	-0.85	-1.3	-0.83	-1.1		-1.7
STR	-1.7	1.5	-0.26	-0.72	0.037	-1.5	-1.4	-1.6	1.7	

Figure from Bingel and Søgaard (2017)

1. CCG tagging
2. Chunking
3. Sentence compression
4. Semantic frames
5. POS tagging
6. Hyperlink prediction
7. Keyphrase detection
8. Multi-word expression detection
9. Super-sense tagging (1 and 2)

Which Target Tasks for Which Source Tasks?

- Multi-task gains are more likely for target tasks that quickly plateau with non-plateauing auxiliary tasks
- Tasks with vocabulary overlap help more
- Closely related tasks help more

Which Target Tasks for Which Source Tasks?

- Multi-task gains are more likely for target tasks that quickly plateau with non-plateauing auxiliary tasks
- Tasks with vocabulary overlap help more
- Closely related tasks help more

Still very much an empirical question

Pretraining + Multi-Task Training?

- All combinations are possible
- If target task training set is small, fine-tuning on this alone might lead to **overfitting**

Cross-lingual Transfer

Cross-lingual Transfer

- Translation of training data from a high-resource language into a low-resource language

Cross-lingual Transfer

- Translation of training data from a high-resource language into a low-resource language
- Translation of test data from a low-resource language into a high-resource language

Cross-lingual Transfer

- Translation of training data from a high-resource language into a low-resource language
- Translation of test data from a low-resource language into a high-resource language
- Multilingual embeddings

Cross-lingual Transfer

- Translation of training data from a high-resource language into a low-resource language
- Translation of test data from a low-resource language into a high-resource language
- Multilingual embeddings
- Multi-task training for cross-lingual transfer

Translation of Data

- Good machine translation system needed
- Not really doable for many low-resource languages

Cross-Lingual Embeddings

- **No** machine translation system needed
- Different options to obtain embeddings in the same space
- Final system is language-agnostic

Multi-Task Training For Cross-Lingual Transfer

- Sharing Information
  - Which parameters should be shared?

Multi-Task Training For Cross-Lingual Transfer

- Sharing Information
  - Which parameters should be shared?
- Training Regime
  - How should losses be weighted?
  - Upsampling/downsampling of data?

Multi-Task Training For Cross-Lingual Transfer

- Sharing Information
  - Which parameters should be shared?
- Training Regime
  - How should losses be weighted?
  - Upsampling/downsampling of data?
- Task embedding?

Task Embedding

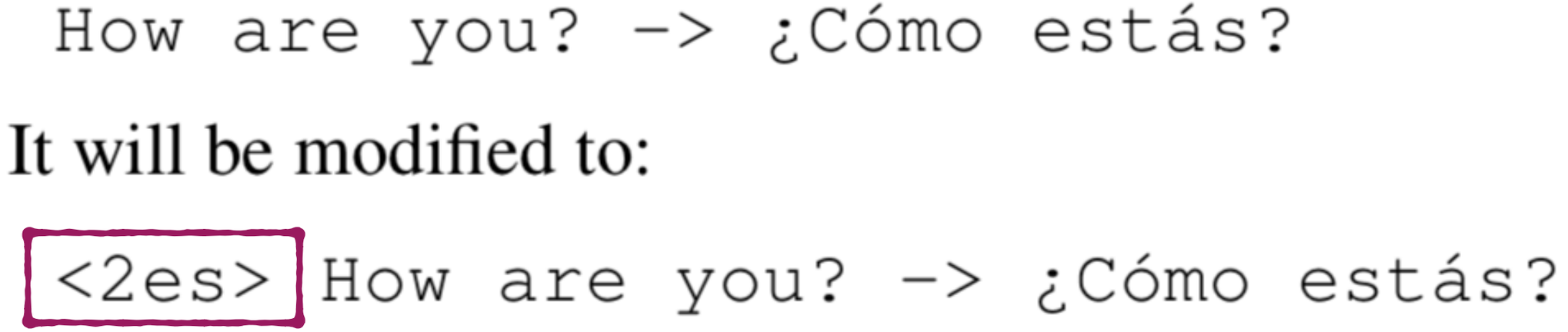


Figure from Johnson et al. (2017)

Transfer Learning for Morphological Inflection

Morphology

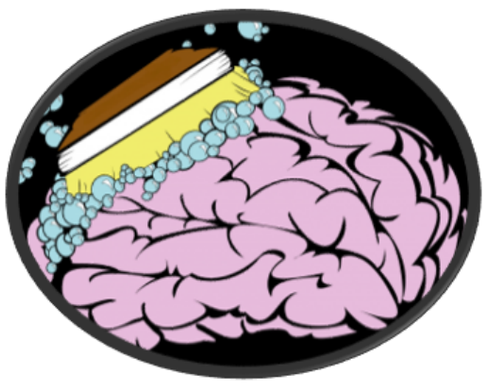
wash

Morphology

washed

Morphology

brainwashed



Morphological Generation

Paradigm of *eat*

eat

eats

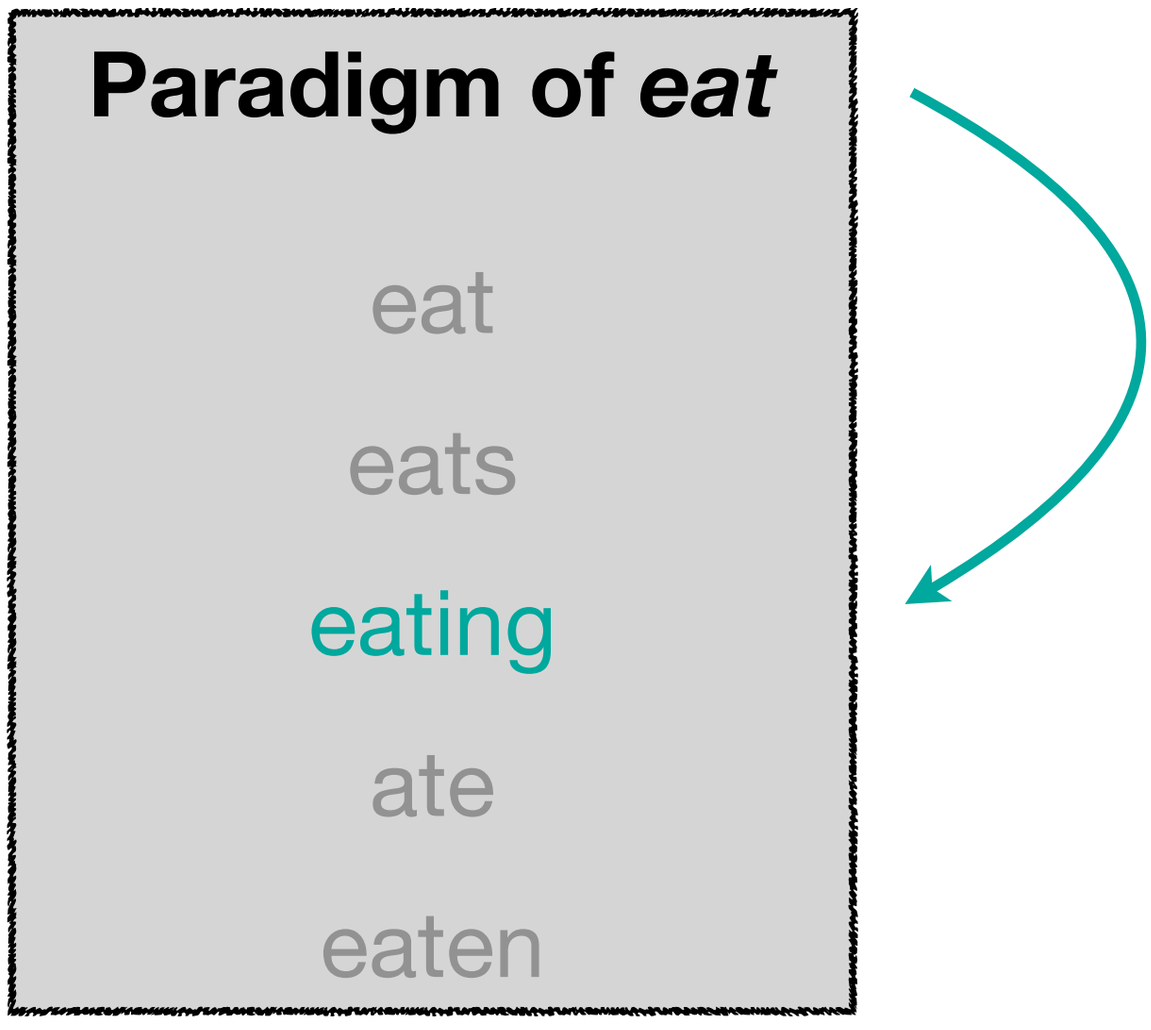
eating

ate

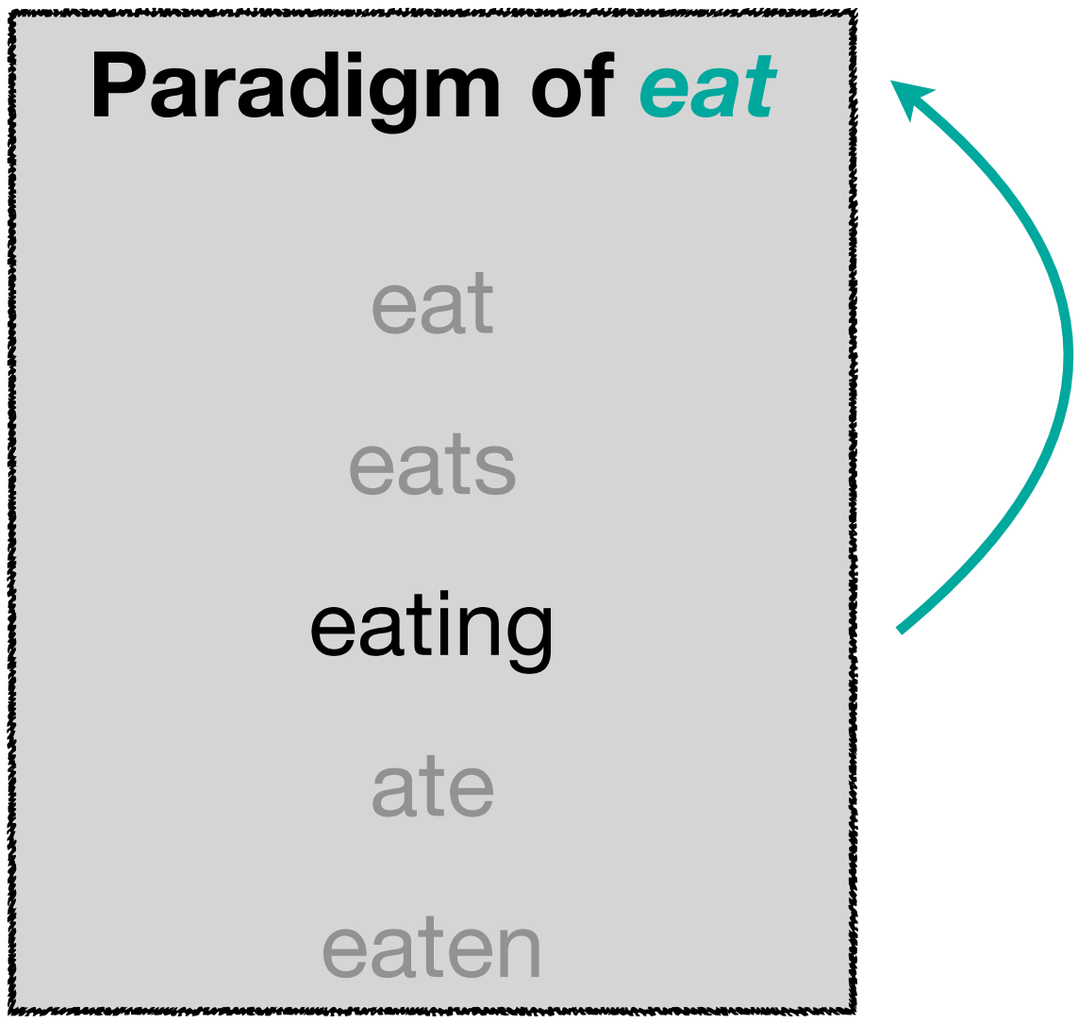
eaten

Morphological Inflection

Present Participle



Lemmatization



Challenges

- **Different inflectional classes**
  - (NOM;PL, Mund) → Münder
  - ...but: (NOM;PL, Hund) → Hunde

Challenges

- **Different inflectional classes**
  - (NOM;PL, Mund) → Münder
  - ...but: (NOM;PL, Hund) → Hunde
- **Stem changes**
  - (1;SG;PRES, aburrir) → aburro
  - ...but: (1;SG;PRES, dormir) → duermo

Challenges

- **Different inflectional classes**
  - (NOM;PL, Mund) → Münder
  - ...but: (NOM;PL, Hund) → Hunde
- **Stem changes**
  - (1;SG;PRES, aburrir) → aburro
  - ...but: (1;SG;PRES, dormir) → duermo
- **Irregular inflections**
  - (PAST, walk) → walked
  - ...but: (PAST, eat) → ate
  - ...and even: (PAST, go) → ?!

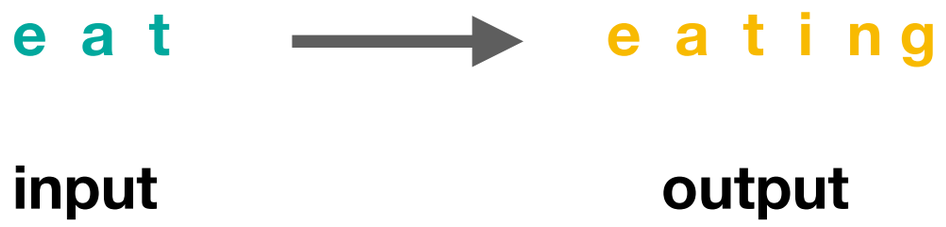
Research question 1:

How can we model morphological inflection?

Kann and Schütze, ACL 2016

Modelling Morphological Inflection

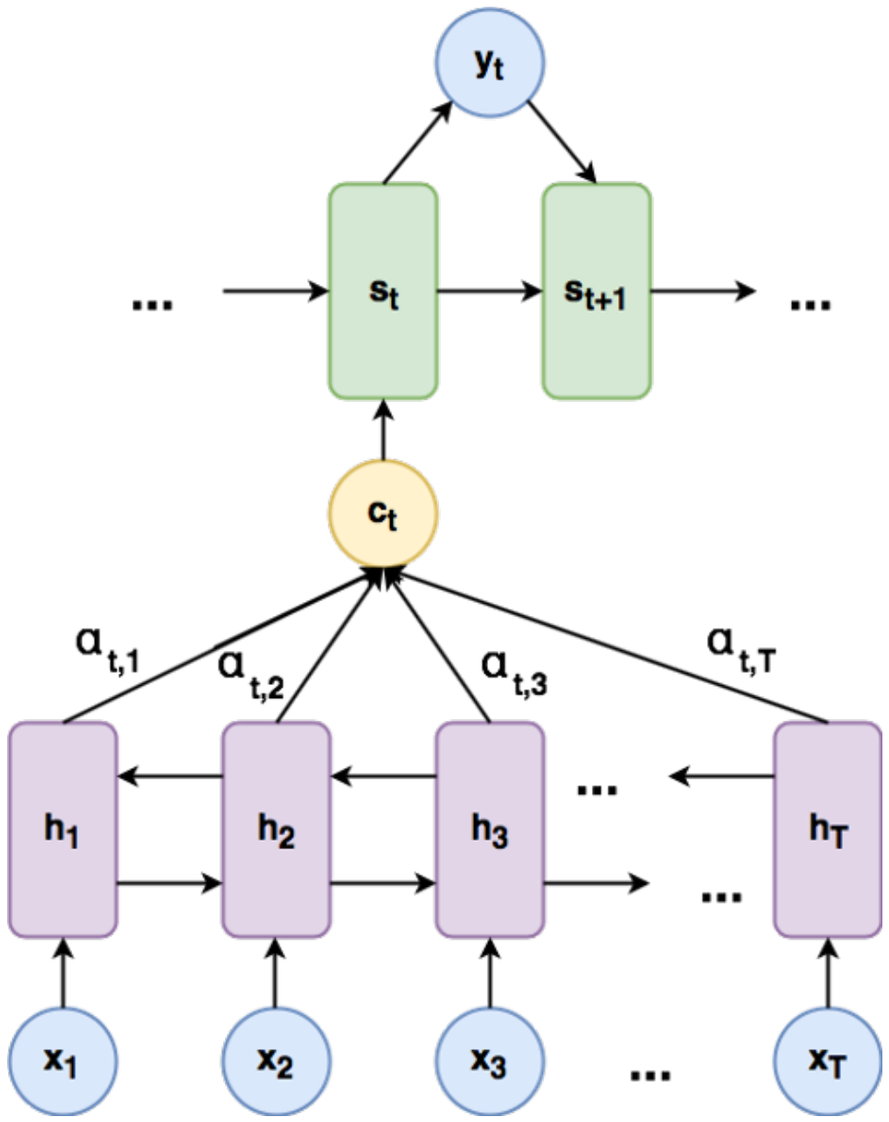
- Cast inflection as a character-based sequence-to-sequence task
- Attention-based encoder-decoder network (Bahdanau et al., 2015)



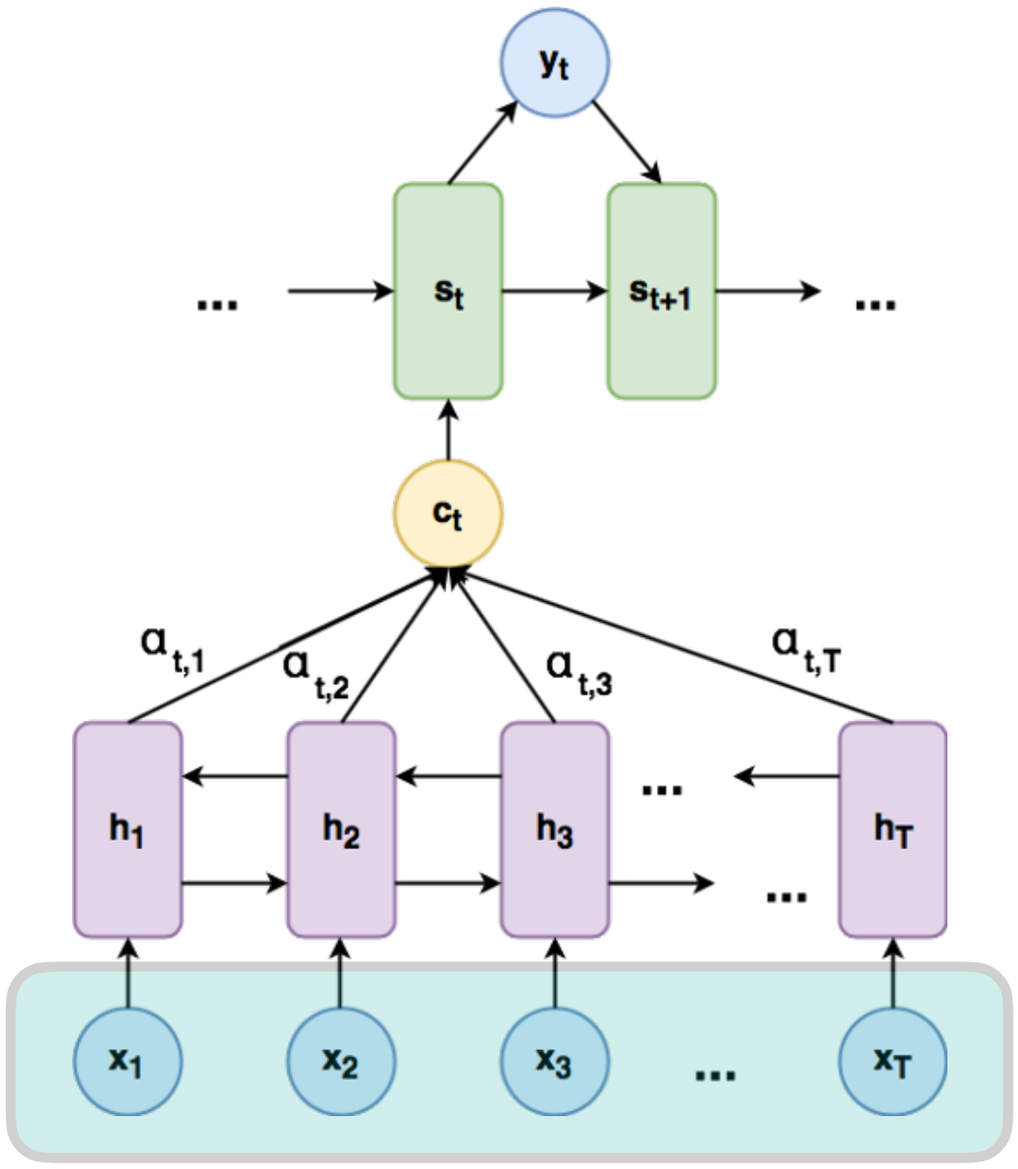
Modelling Morphological Inflection

- Cast inflection as a character-based sequence-to-sequence task
- Attention-based encoder-decoder network (Bahdanau et al., 2015)

Modelling Morphological Inflection



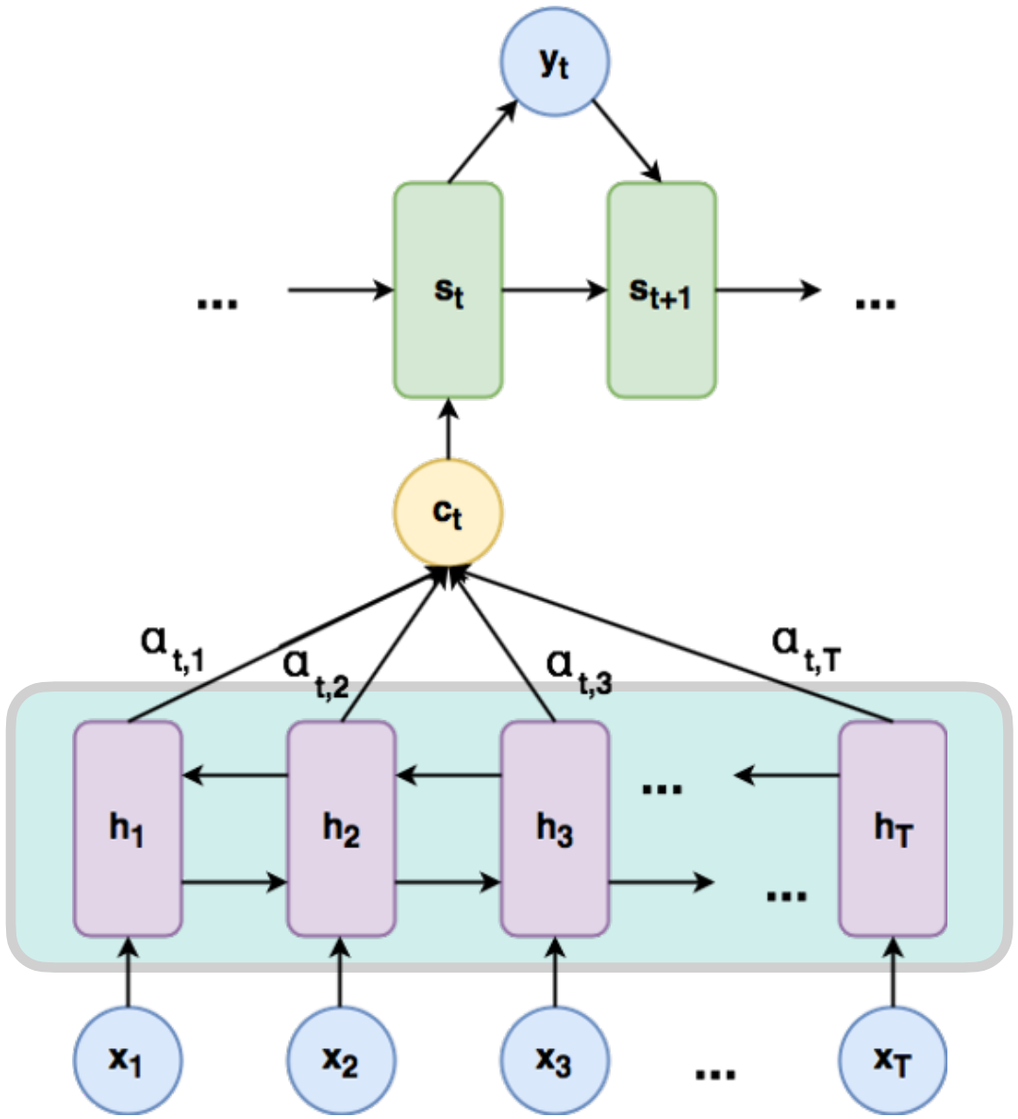
Modelling Morphological Inflection



Input embedding layer

- Character + tag embeddings
- Encodes the input sequence

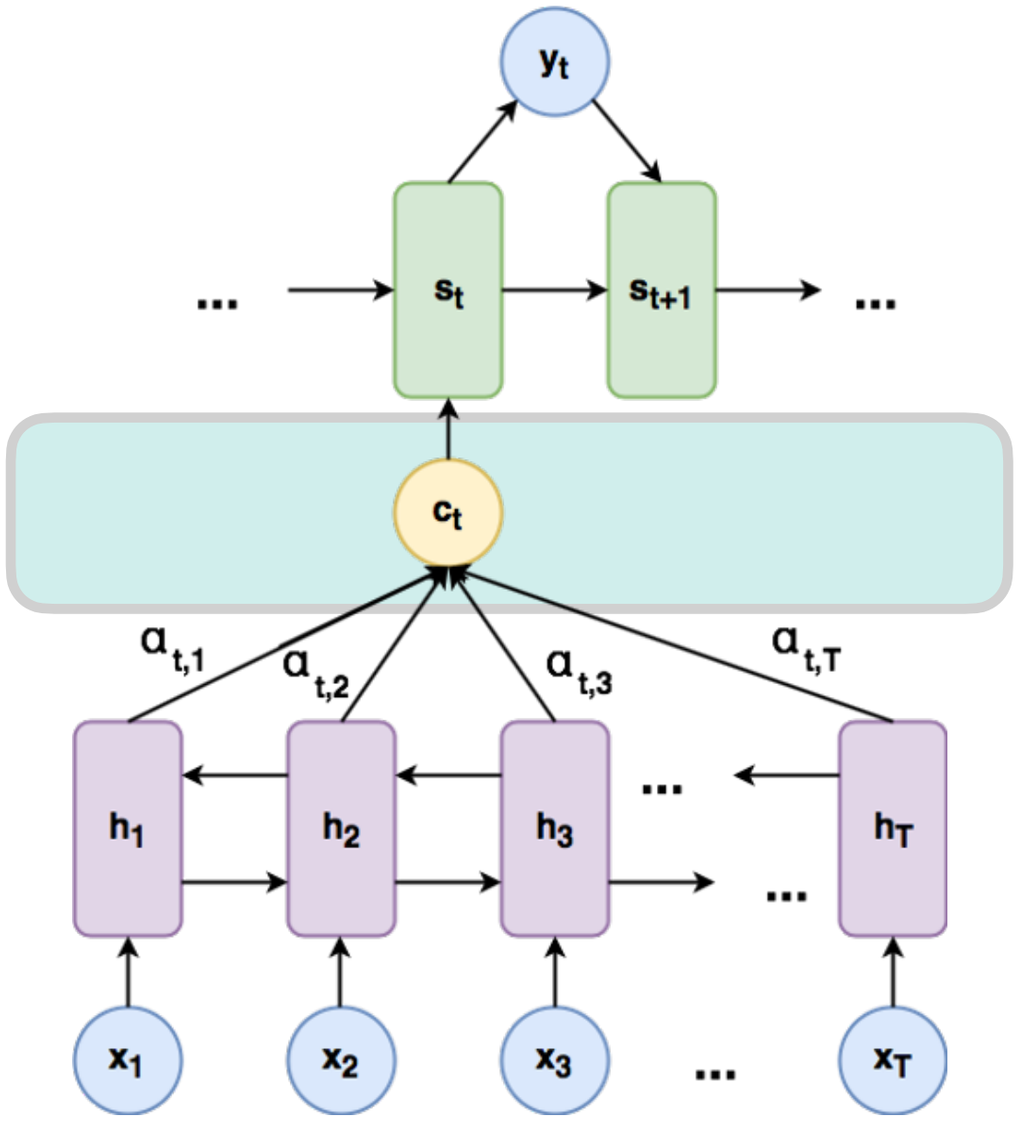
Modelling Morphological Inflection



Encoder

- Bidirectional gated recurrent unit (GRU)
- Hidden states are a representation of the input in context

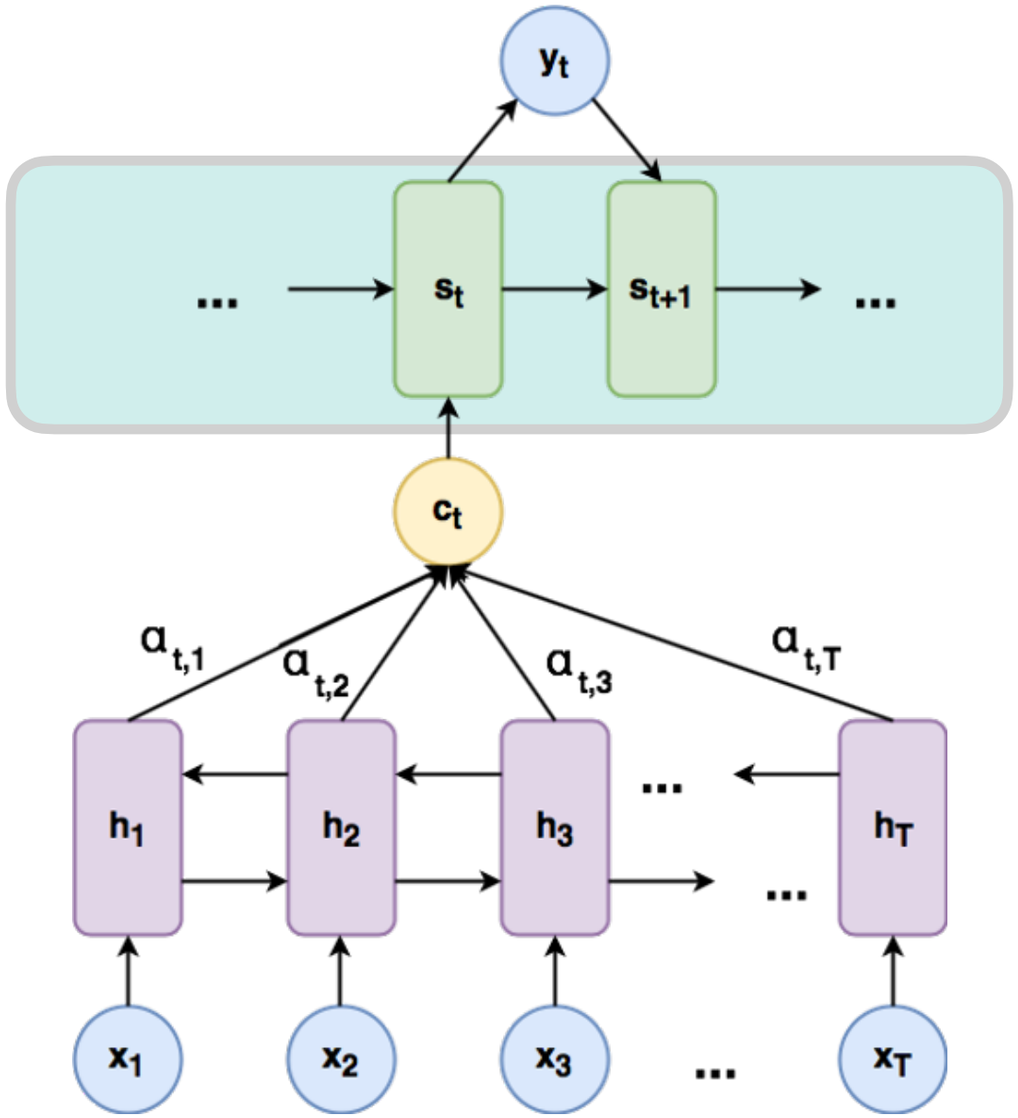
Modelling Morphological Inflection



Attention

- Computes a relevant combination of the encoder hidden states at each time step

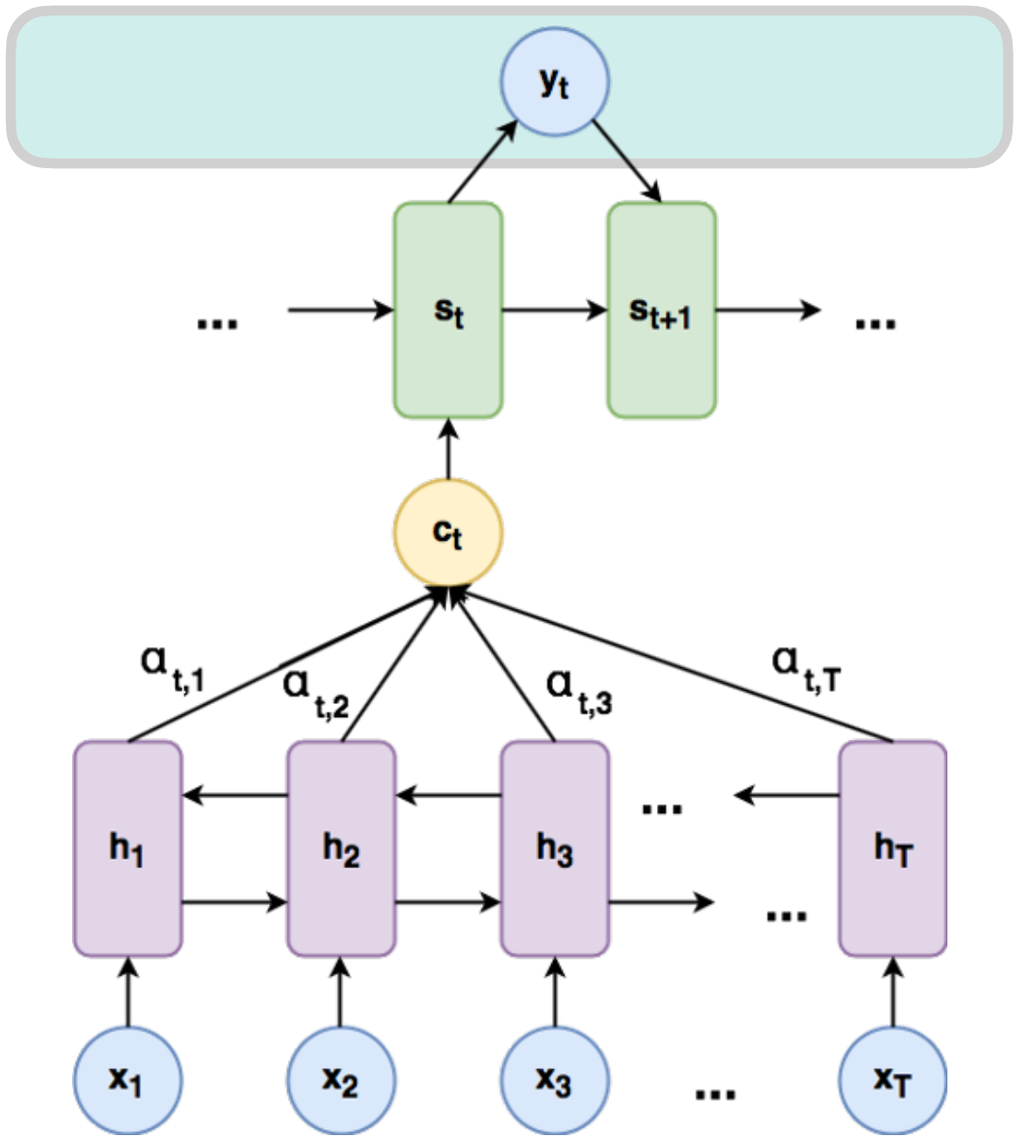
Modelling Morphological Inflection



Decoder

- Unidirectional GRU
- Predicts output characters

Modelling Morphological Inflection



Target embedding layer

- Embeds last predicted character

Modelling Morphological Inflection

Key idea:

Encoding tags enables models to train on all available data at once.

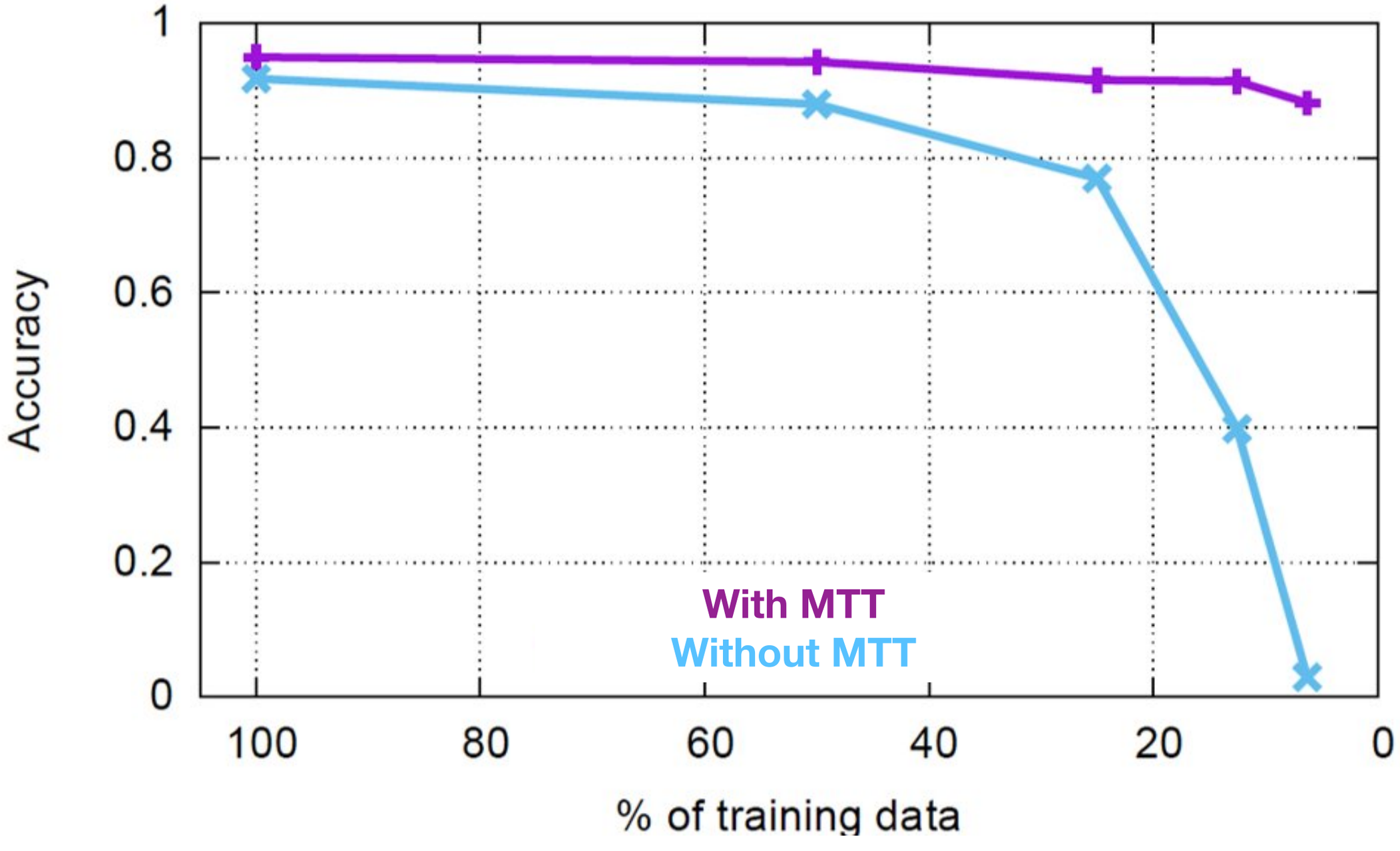
➡ Multi-task training

Humans also do this!

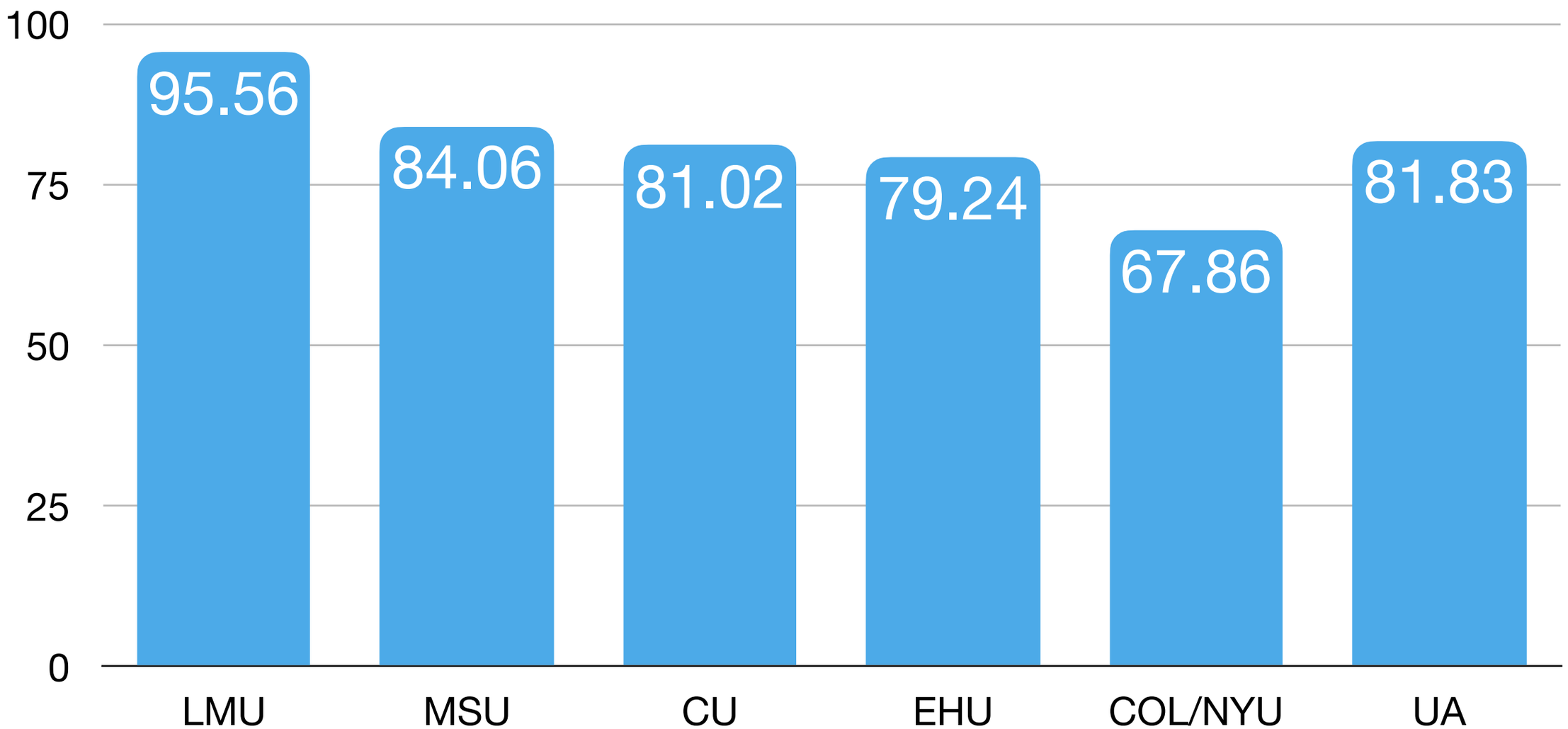
Experiment

- 1000 examples for 3 source tasks (~source paradigm slots)
- Varying number of examples for target task (~target paradigm slot)

Experimental Results



Shared Task Results 2016



Research question 2:

Does cross-lingual transfer work for morphological inflection?

Kann, Cotterell and Schütze, ACL 2017

Cross-lingual Transfer

- Translation of training data from a high-resource language into a low-resource language

- Translation of test data from a high-resource language into a low-resource language

- Multilingual embeddings

Cross-lingual Transfer

- Translation of training data from a high-resource language into a low-resource language

- Translation of test data from a high-resource language into a low-resource language

- Multilingual embeddings

But morphological inflection is not a semantic task!

Cross-lingual Inflection

Cross-lingual Inflection

plural d o g → ?

Cross-lingual Inflection

plural d o g → c a t s

Cross-lingual Inflection

plural c a t → c a t s

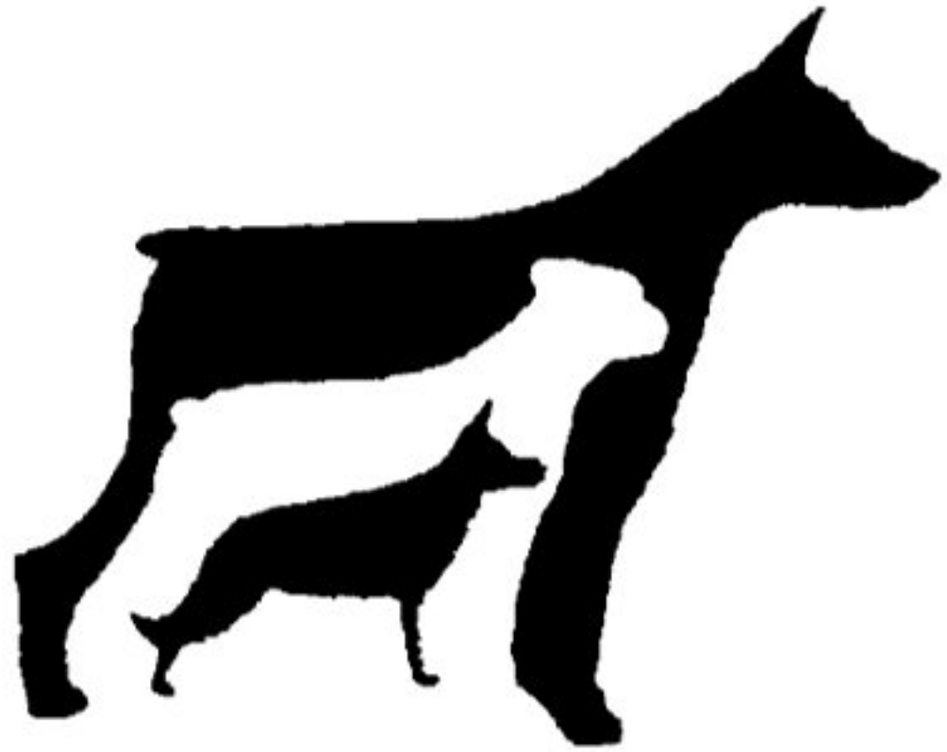
plural g a t o → g a t o s

plural m a n o → m a n o s

...

Cross-lingual Inflection

plural d o g → d o g s



lavar

Spanish	Portuguese
lavo	lavo
lavas	lavas
lava	lava
lavamos	lavamos
laváis	lavais
lavan	lavam

lavar

Spanish	Portuguese
lavo	lavo
lavas	lavas
lava	lava
lavamos	lavamos
laváis	lavais
lavan	lavam

Cross-lingual Inflection

- Use a (trainable) language embedding
- Multi-task training on multiple languages
- No architectural changes necessary

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{(k, w_{\ell_t}) \in \mathcal{D}_t} \log p_{\boldsymbol{\theta}}\left(f_k[w_{\ell_t}] \mid w_{\ell_t}, t_k, \lambda_{\ell_t}\right) + \sum_{(k, w_{\ell_s}) \in \mathcal{D}_s} \log p_{\boldsymbol{\theta}}(f_k[w_{\ell_s}] \mid w_{\ell_s}, t_k, \lambda_{\ell_s})$$

EN plural c a t → **c a t s**

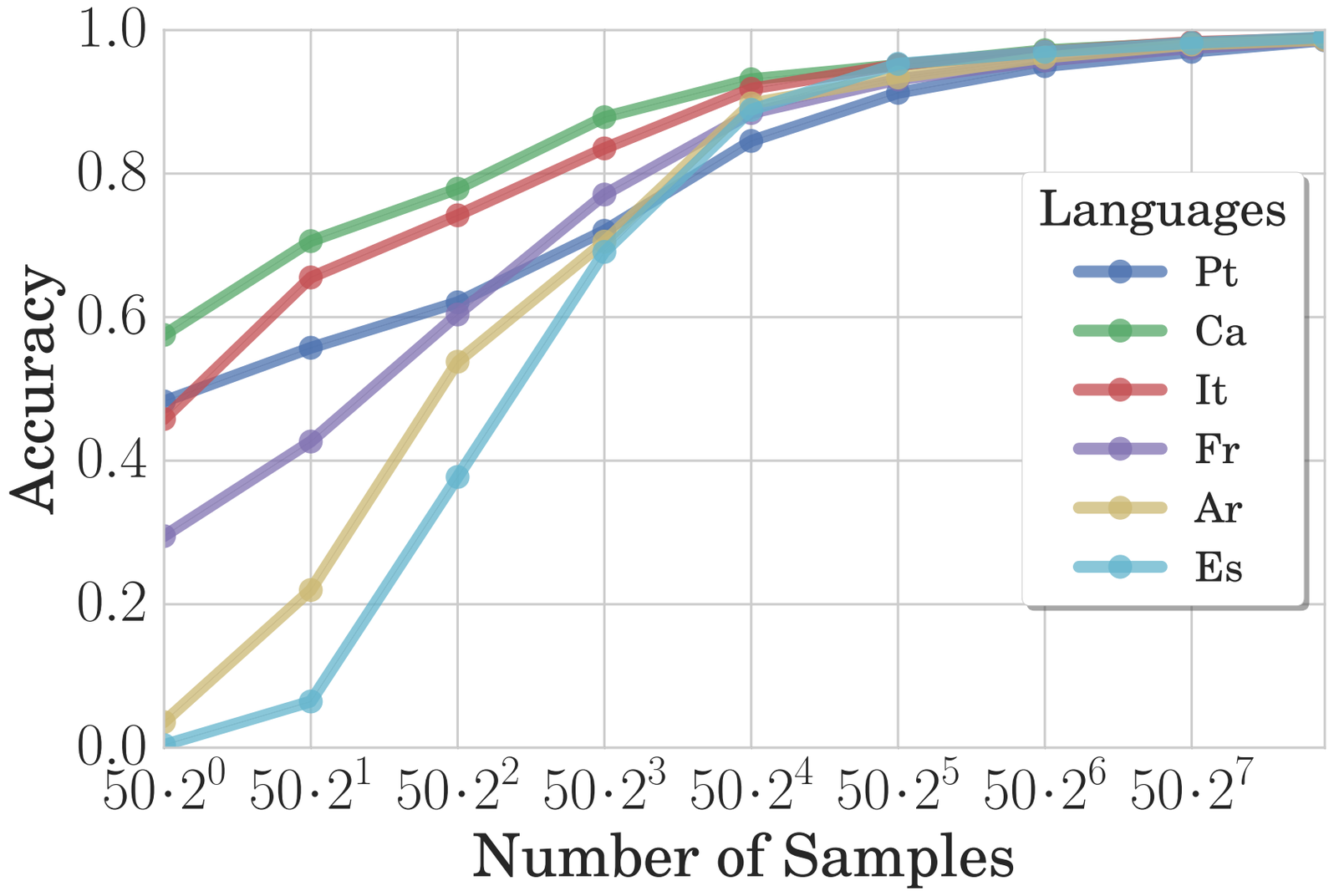
Experiments

- **Q1**: Is cross-lingual transfer possible for morphological inflection?
- **Q2**: How much annotated data do we need in the low-resource language?
- **Q3**: How closely related do source and target language have to be?

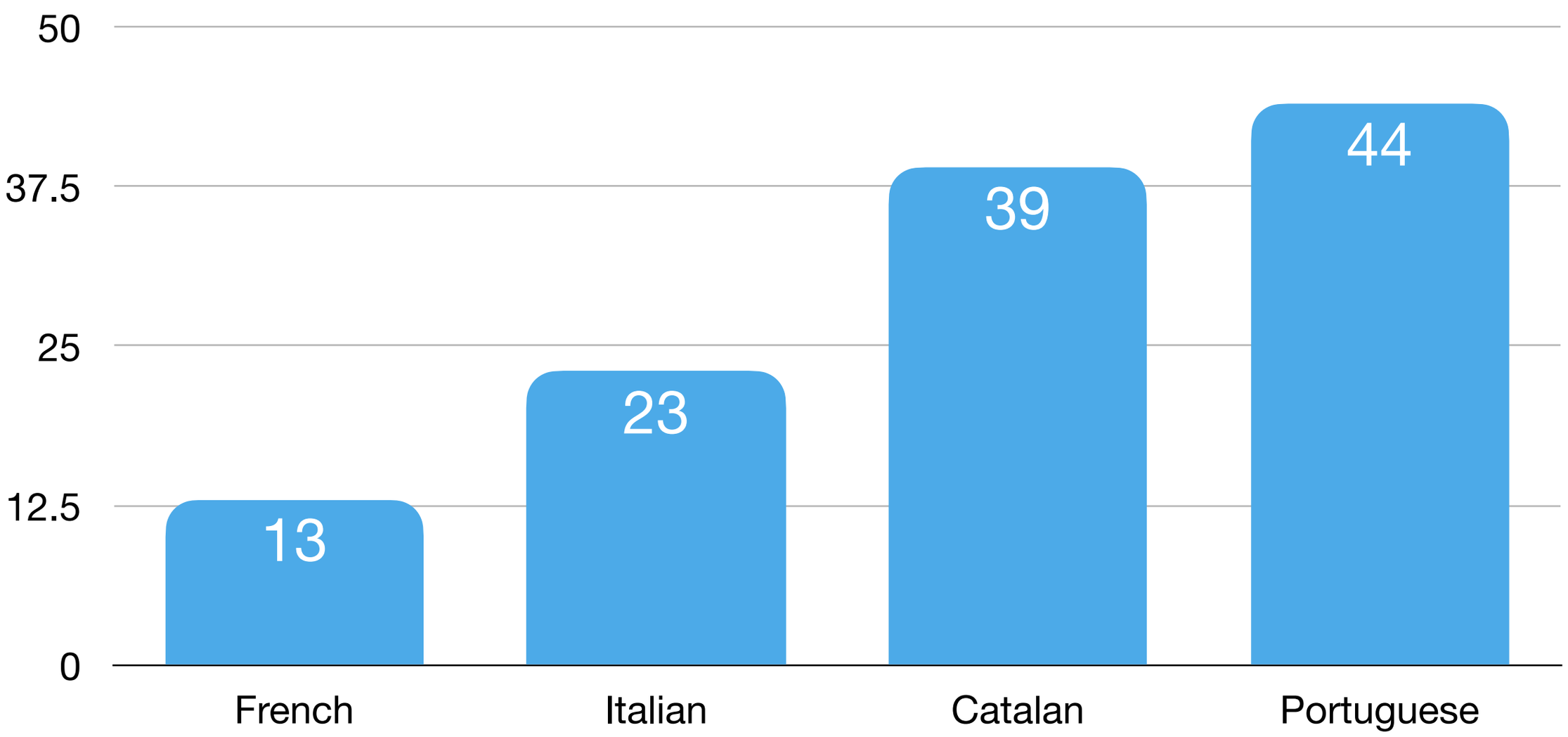
Experiments

Lexical similarity:

Portuguese:	89%
Catalan:	85%
Italian:	82%
French:	75%



One-Shot Results



Research question 3:

What can we do if data from related languages isn’t available, either?

Kann and Schütze, SCLeM 2017

Semi-supervised Training

- Make use of (large) unlabeled corpora
- Learn properties of language on general text

Semi-supervised Training

- Make use of (large) unlabeled corpora
- Learn properties of language on general text

For morphological inflection:

1) Define an autoencoding auxiliary task
2) Create examples from corpus data
3) Add a task embedding
4) Maximize the joint loss

***AE* s u n** → **s u n**

Semi-supervised Training

- Make use of (large) unlabeled corpora
- Learn properties of language on general text

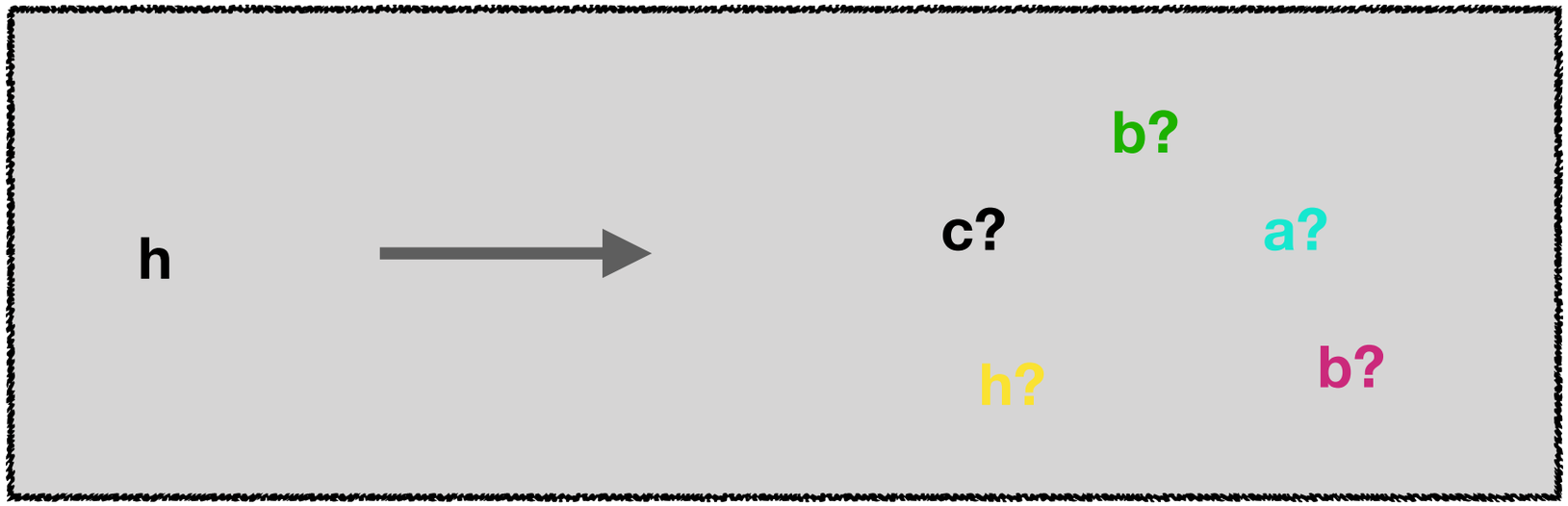
For morphological inflection:

1) Define an autoencoding auxiliary task
2) Create examples from corpus data
3) Add a task embedding
4) Maximize the joint loss

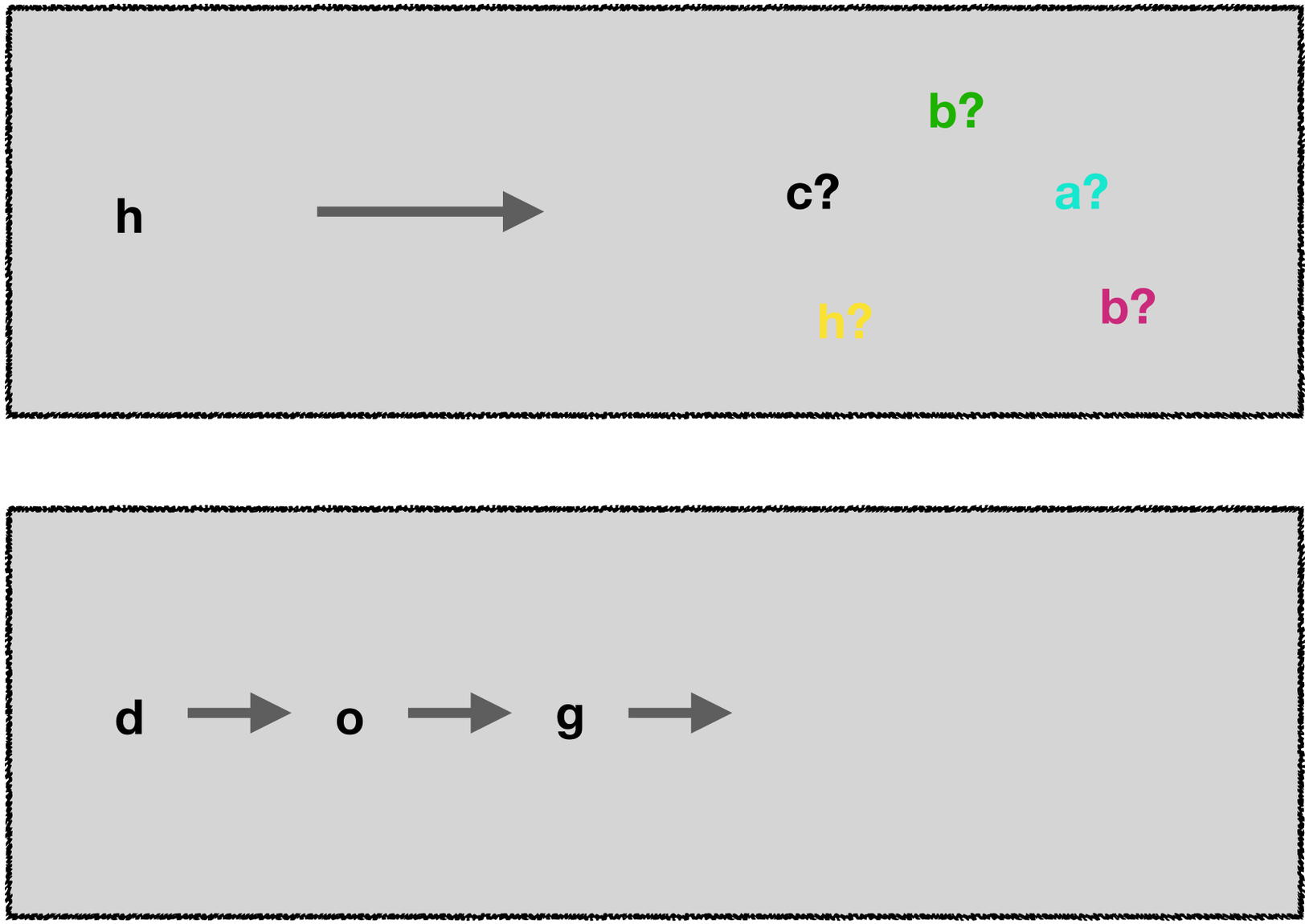
***AE* s u n** → **s u n**

+ do the same for random strings

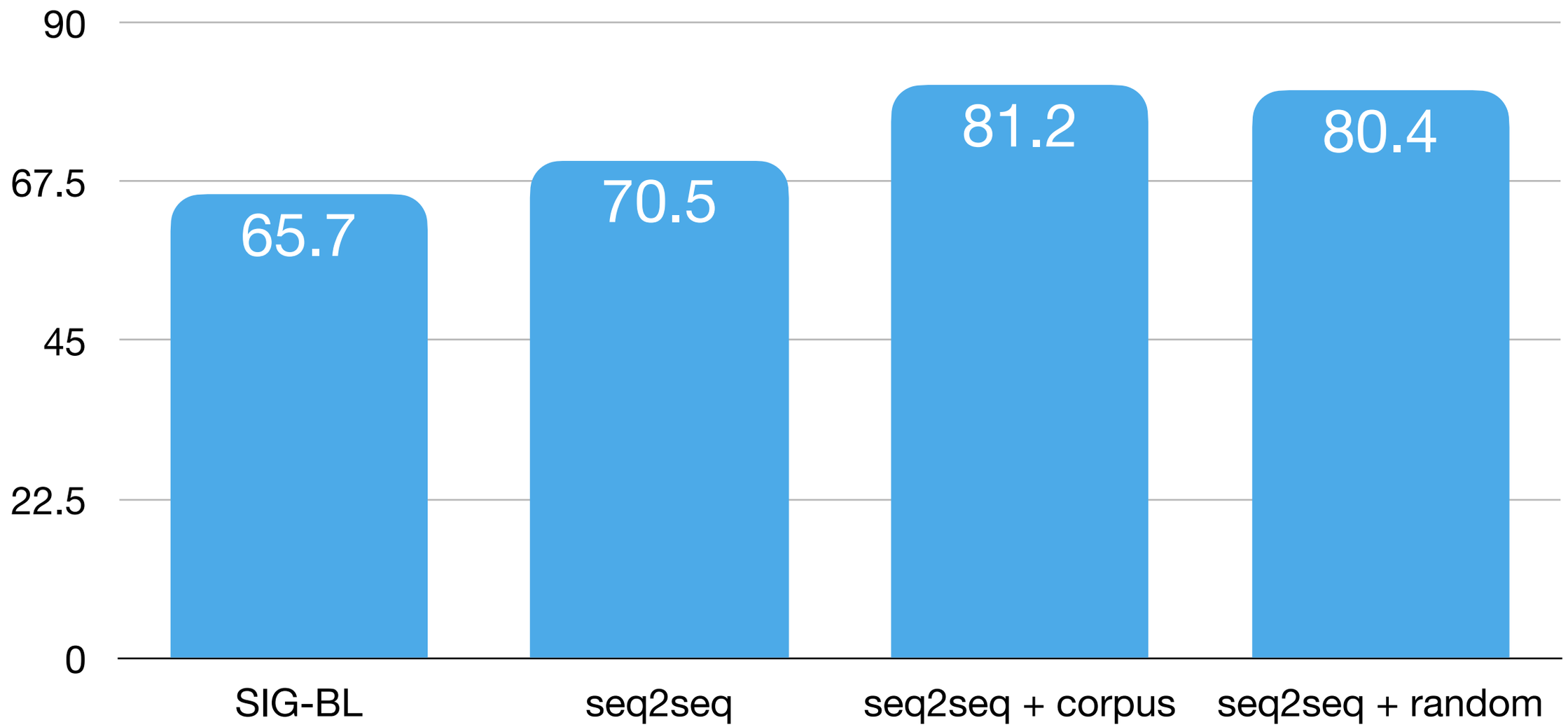
Semi-supervised Training



Semi-supervised Training



Experiments



Work Needed for Low-Resource Languages

Learning from Limited Resources

- Efficient machine learning models
- Data augmentation approaches

Language-Dependent Preprocessing

- Word segmentation
- Morphological segmentation
- Morphological inflection and analysis

Datasets

- Adaptation of existing datasets to new languages
- Creation of new datasets for known tasks in new languages
- Creation of datasets for language-specific tasks

Knowledge Transfer

- Multi-task learning/multi-task training
- Cross-lingual transfer
- Domain adaptation
- Pre-training

Q&A Session!