Deep Contextualized Word Representations (ELMo)

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer, NAACL 2018.

Anish Khazane & Yang Jingfeng

Outline

- Refresher – What's problematic with Word2Vec/GloVe?
- Quick overview on language modeling
- Initial attempts at tackling contextual awareness (CoVe)
- Introduce ELMo
- Discuss Evaluation Tasks (Semantic Role Labeling, NER, etc)
- Analyzing ELMo: What information is captured?
- Advantages of using ELMo
- Disadvantages of using ELMo / Concluding Thoughts

Issues with Word2Vec/GLoVe

Context-Insensitive Approaches

*"I went to the **play** earlier this afternoon"*

*"I want to **play** the new video game."*

How do we differentiate an embedding representation for **play** from the previous two sentences?

- In W2V, GLoVe, F(**play**) is the same for each sentence, despite the semantics being different
- W2V, GLoVe unable to represent words **outside of a fixed vocabulary**.
- Previous approaches did not take **word-order** into account (context-insensitive)

What do we mean by Context?

- Context can be defined by surrounding words in a sentence, paragraph, document, and so on.
- For this work, we'll define it as surrounding sentence. The meaning of individual words can completely change depending on the sentence.
- In a **context-aware** embedding model we augment the embedding for a word with the context in which it appears

$$y = F(\text{word}, \text{context})$$

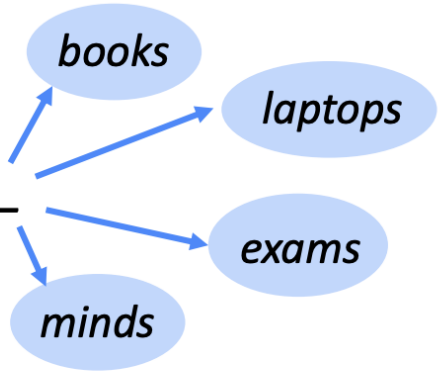
- We call these **contextualized word embeddings.** So, how do we define F? Use language modeling!

Basics of Language Modeling

What is a language model (LM)?

- **Language Modeling** is the task of predicting what word comes next.

the students opened their _____



- More formally: given a sequence of words $x^{(1)}, x^{(2)}, \ldots, x^{(t)}$, compute the probability distribution of the next word $x^{(t+1)}$:

$$P(\boldsymbol{x}^{(t+1)} | \ \boldsymbol{x}^{(t)}, \ldots, \boldsymbol{x}^{(1)})$$

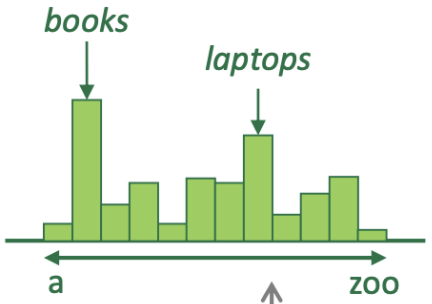
where $x^{(t+1)}$ can be any word in the vocabulary $V = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{|V|}\}$

- A system that does this is called a **Language Model**.

*Adapted from CS224n slides at Stanford

A RNN Language Model

$\hat{\boldsymbol{y}}^{(4)} = P(\boldsymbol{x}^{(5)}|\text{the students opened their})$



output distribution

$$\hat{\boldsymbol{y}}^{(t)} = \text{softmax}\left(\boldsymbol{U}\boldsymbol{h}^{(t)} + \boldsymbol{b}_2\right) \in \mathbb{R}^{|V|}$$

hidden states

$$\boldsymbol{h}^{(t)} = \sigma\left(\boldsymbol{W}_h\boldsymbol{h}^{(t-1)} + \boldsymbol{W}_e\boldsymbol{e}^{(t)} + \boldsymbol{b}_1\right)$$

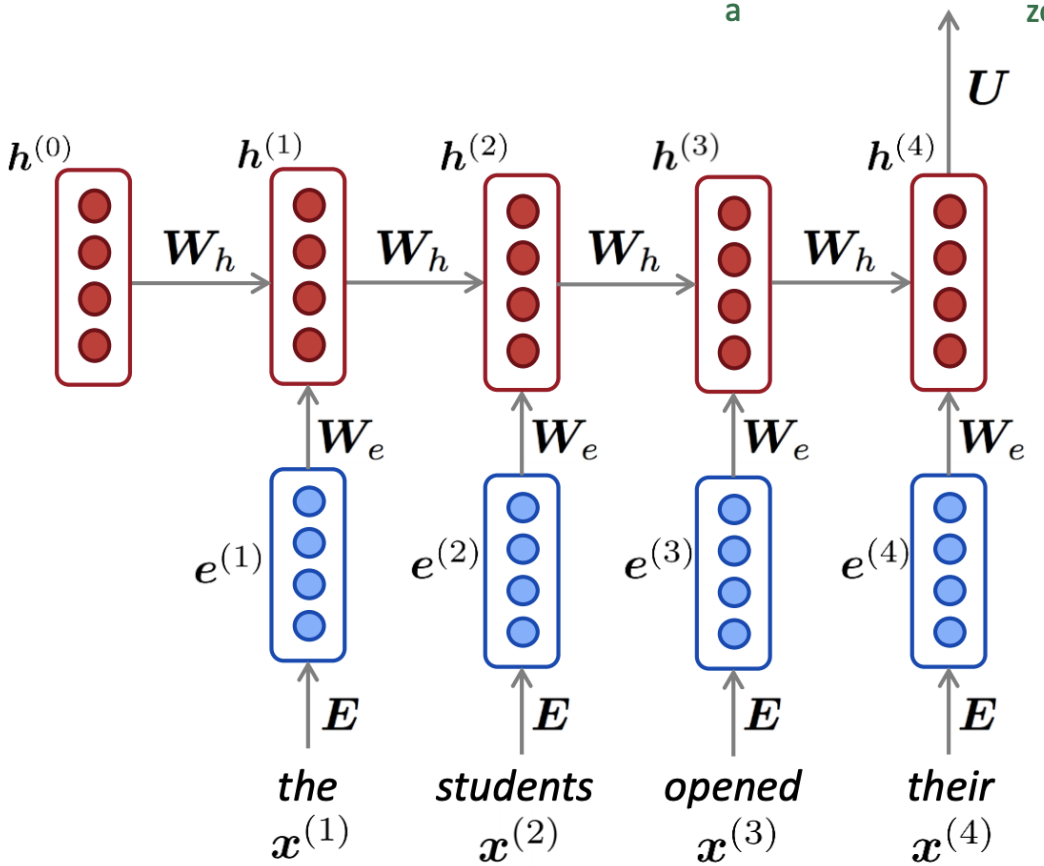
$\boldsymbol{h}^{(0)}$ is the initial hidden state

word embeddings

$$\boldsymbol{e}^{(t)} = \boldsymbol{E}\boldsymbol{x}^{(t)}$$

words / one-hot vectors

$$\boldsymbol{x}^{(t)} \in \mathbb{R}^{|V|}$$

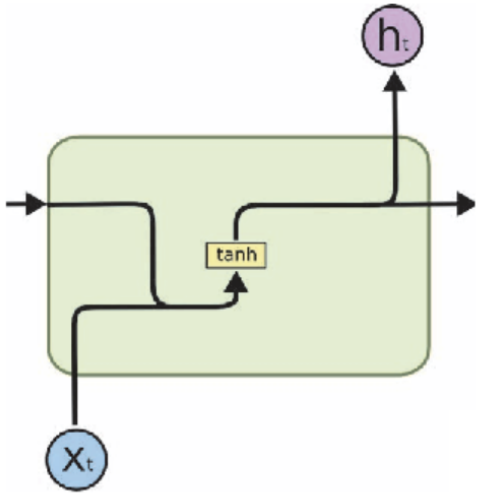


***Note**: this input sequence could be much longer, but this slide doesn't have space!*

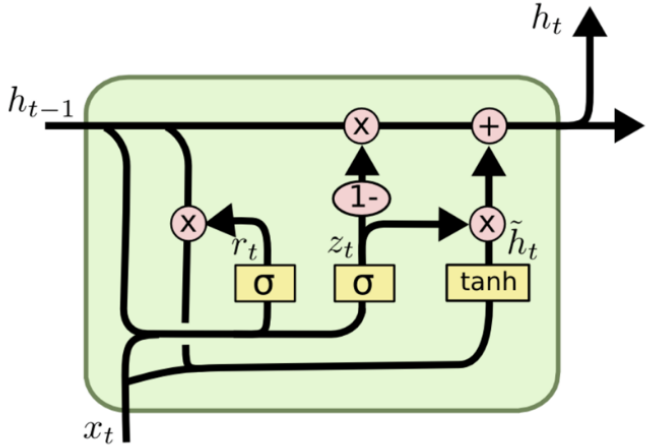
*Adapted from CS224n slides at Stanford

More complicated modules (LSTMs, GRUs)

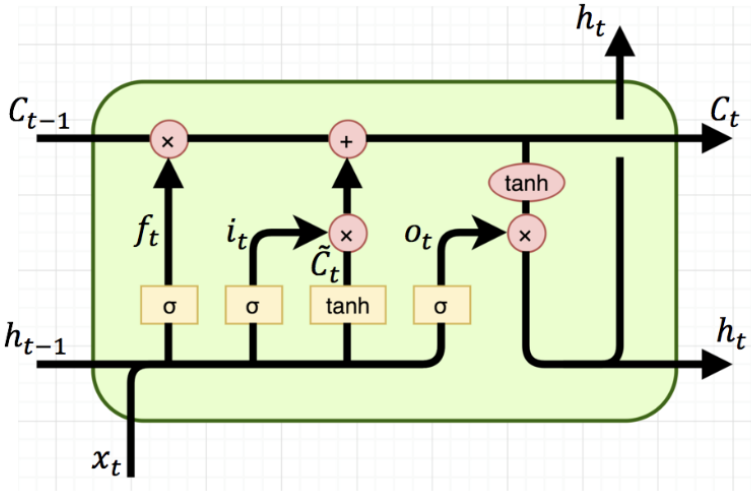
Basic RNN module



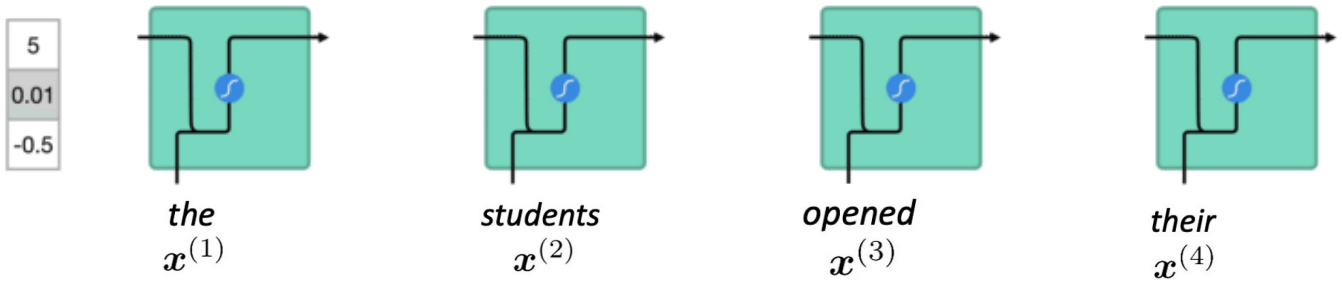
Basic GRU module



Basic LSTM module



Each module creates representations based off <u>contextual relationships</u>

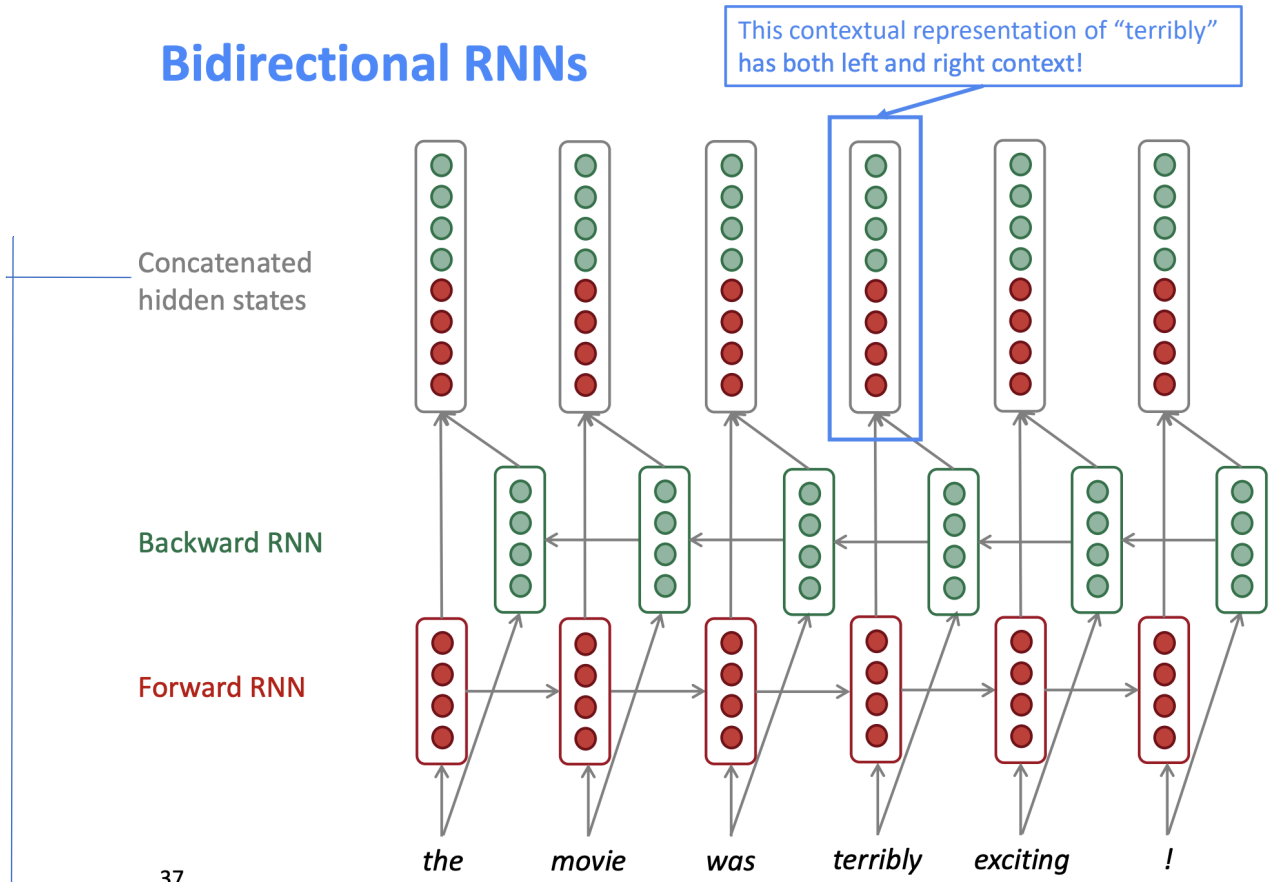


Prediction of next word

*Image from "*Simple RNN vs GRU vs LSTM :- Differences Lies in More Flexible Control*"

We can even stack these units! (bi-LSTMs, bi-RNNs, etc)

Bidirectional RNNs



Backward LSTM objective

$$\sum_{k=1}^{N} p(x_k | x_{k+1}, x_{k+2}, \ldots, x_N; \overleftarrow{\theta}_{LSTM})$$

Forward LSTM objective

$$\sum_{k=1}^{N} p(x_k | x_0, x_1, \ldots, x_{k-1}; \vec{\theta}_{LSTM})$$

Concatenate forward and backward representations for every token, *k*.

$$[(\vec{h}_0; \overleftarrow{h}_0), (\vec{h}_1; \overleftarrow{h}_1), \ldots, (\vec{h}_N; \overleftarrow{h}_N)]$$

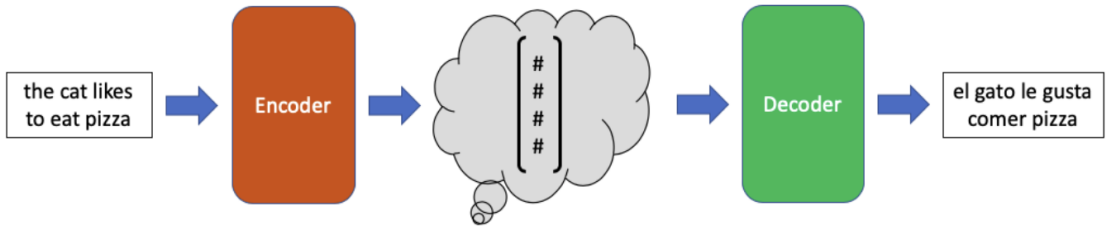
$$y_k = (\vec{h}_k; \overleftarrow{h}_k)$$

*Adapted from CS224n slides at Stanford

Previous Applications of Language Models

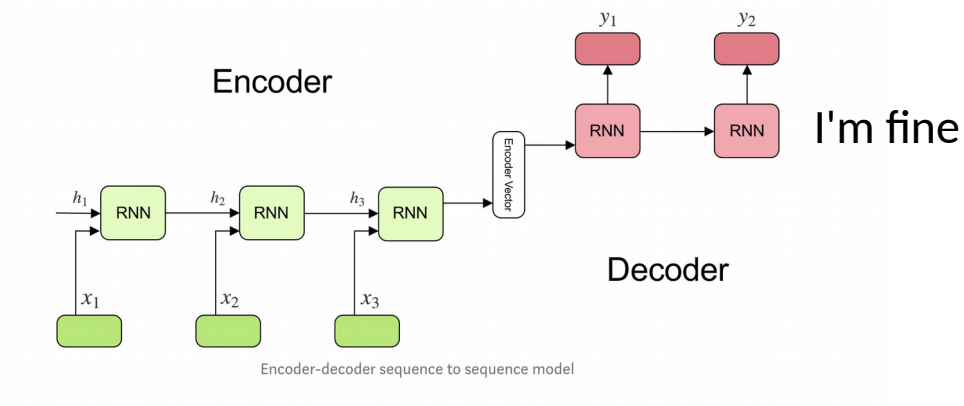
Applications of RNN, LSTM type structures

Machine Translation



Natural Language Generation

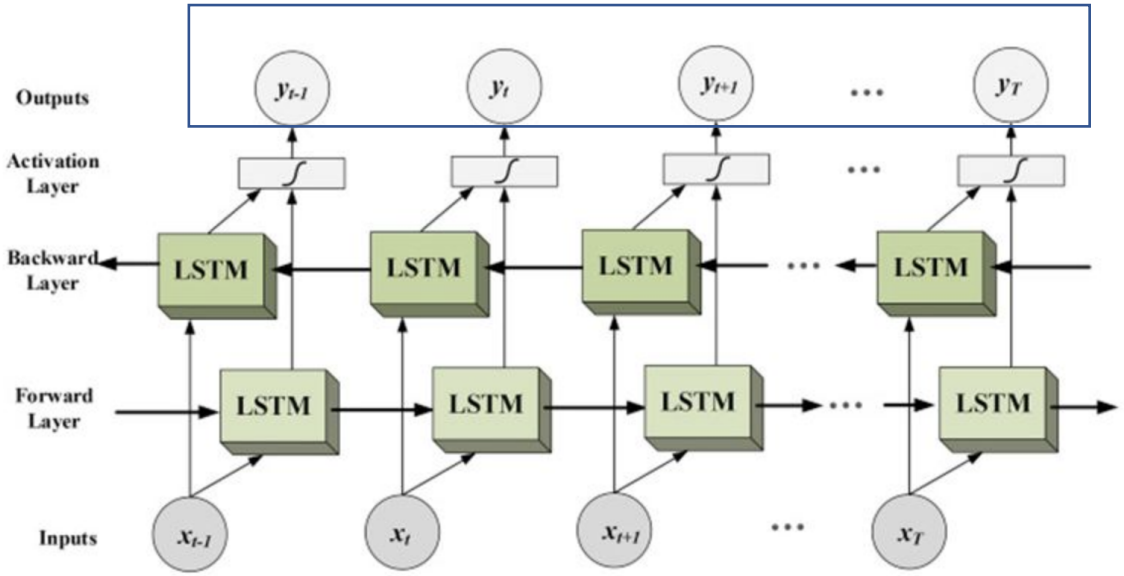
How are you?



*Adapted from CS224n slides at Stanford

CoVe - Contextualized Word Vectors

- *Learned in Translation: Contextual Word Vectors* (McCann, et al. 2017).
- Pre-train bi-LSTM on Machine Translation (supervised) dataset
- Throw away decoder
- Use the final internal hidden states of bi-LSTM as CoVe embeddings!



→ These are the CoVe embeddings for every input, *x*.

DRAWBACKS

- Must pre-train on supervised corpora (not so abundant)
- Authors only take the last hidden layer's representation....why not experiment with more than one?

*Adapted from "*Deep Dive into Bidirectional LSTM*"

ELMo: Embeddings from Language Models

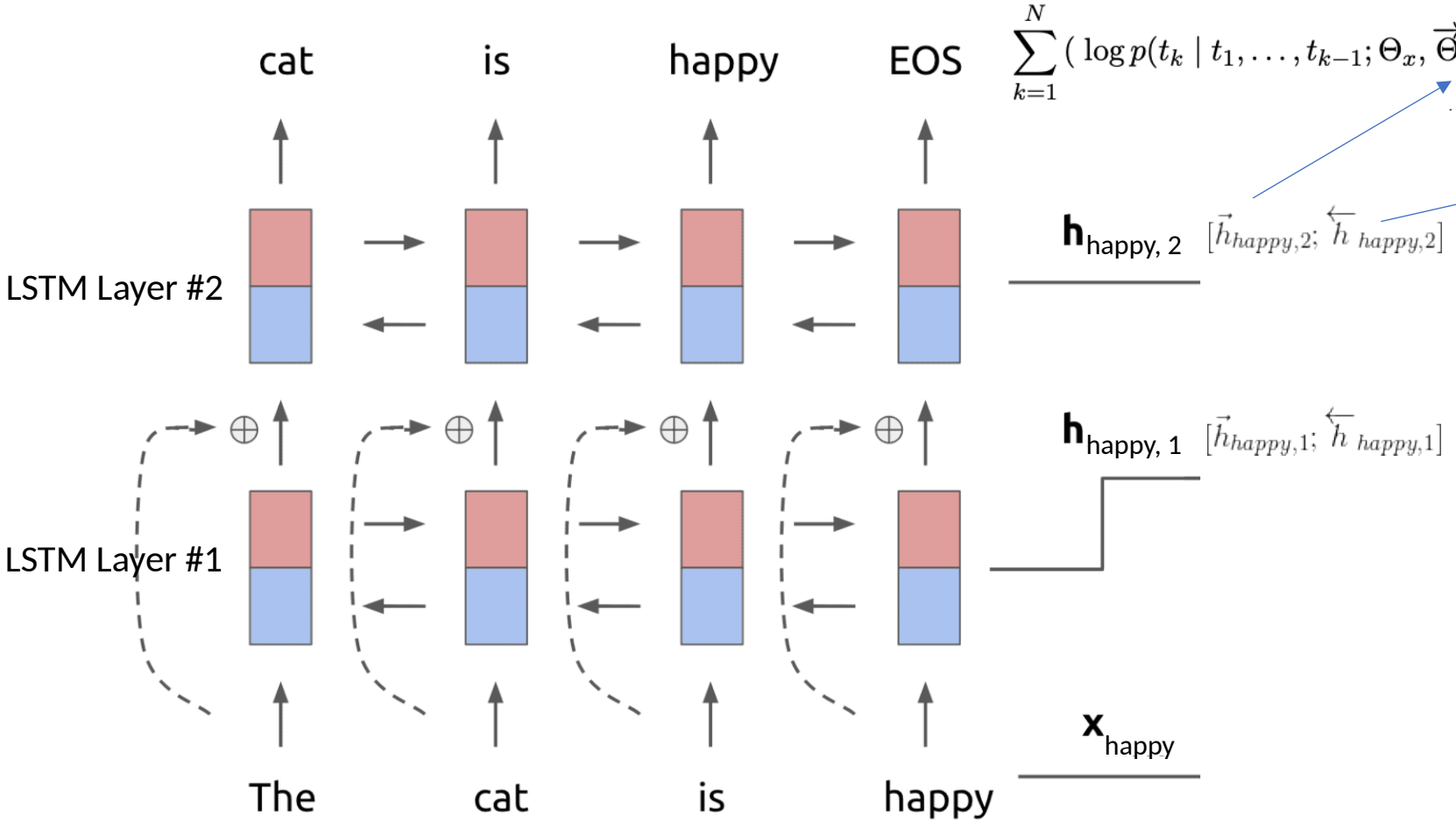
ELMo at a Glance

- Starting from a standard language model architecture…
  - Pre-train on unlabeled data (optimize language modeling objective)
  - Use multiple layers of bi-LSTMs in architecture
  - For any downstream task, create the task-specific embeddings as a linear combination of all the internal layer representations

ELMo Architecture: Stage 1 - Pretrain

Language Modeling Objective

$$\sum_{k=1}^{N} \left(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s \quad + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \right).$$



The representation for input token k is now

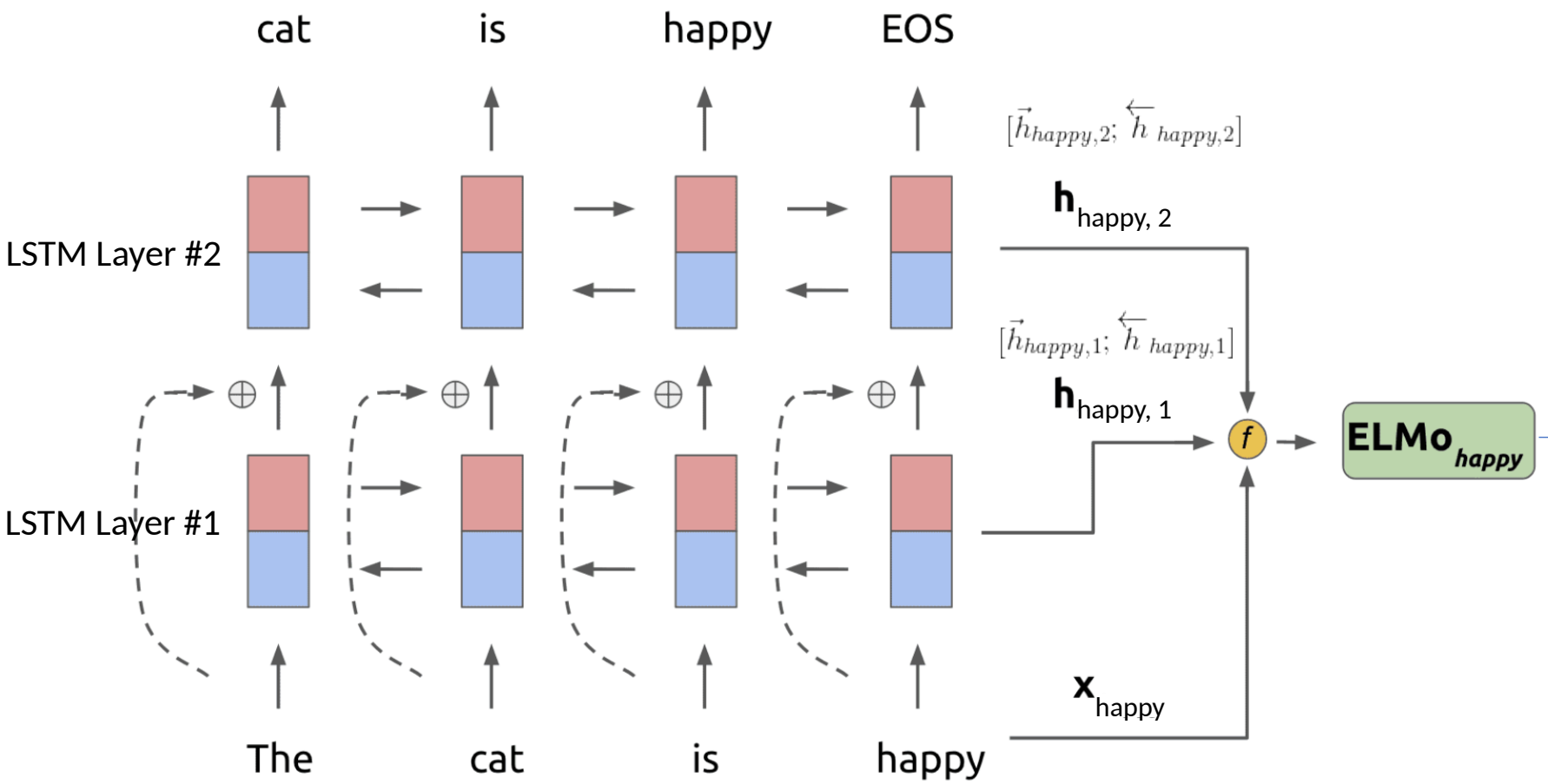
$$R_k = \{x_k, \vec{h}_{k,j}, \overleftarrow{h}_{k,j} \mid j = 1, \ldots, L\}$$

$$= \{h_{k,j} \mid j = 0, \ldots, L\}$$

given $h_{k,0} = x_k$ and $h_{k,j} = [\vec{h}_{k,j}; \overleftarrow{h}_{k,j}]$ otherwise

*Adapted from *"Deep Contextualized Word Representations with ELMo"*

ELMo Architecture: Stage 2: Combine Layer Representations



The representation for input token k is now

$$R_k = \{x_k, \vec{h}_{k,j}, \overleftarrow{h}_{k,j} \mid j = 1, \dots, L\}$$

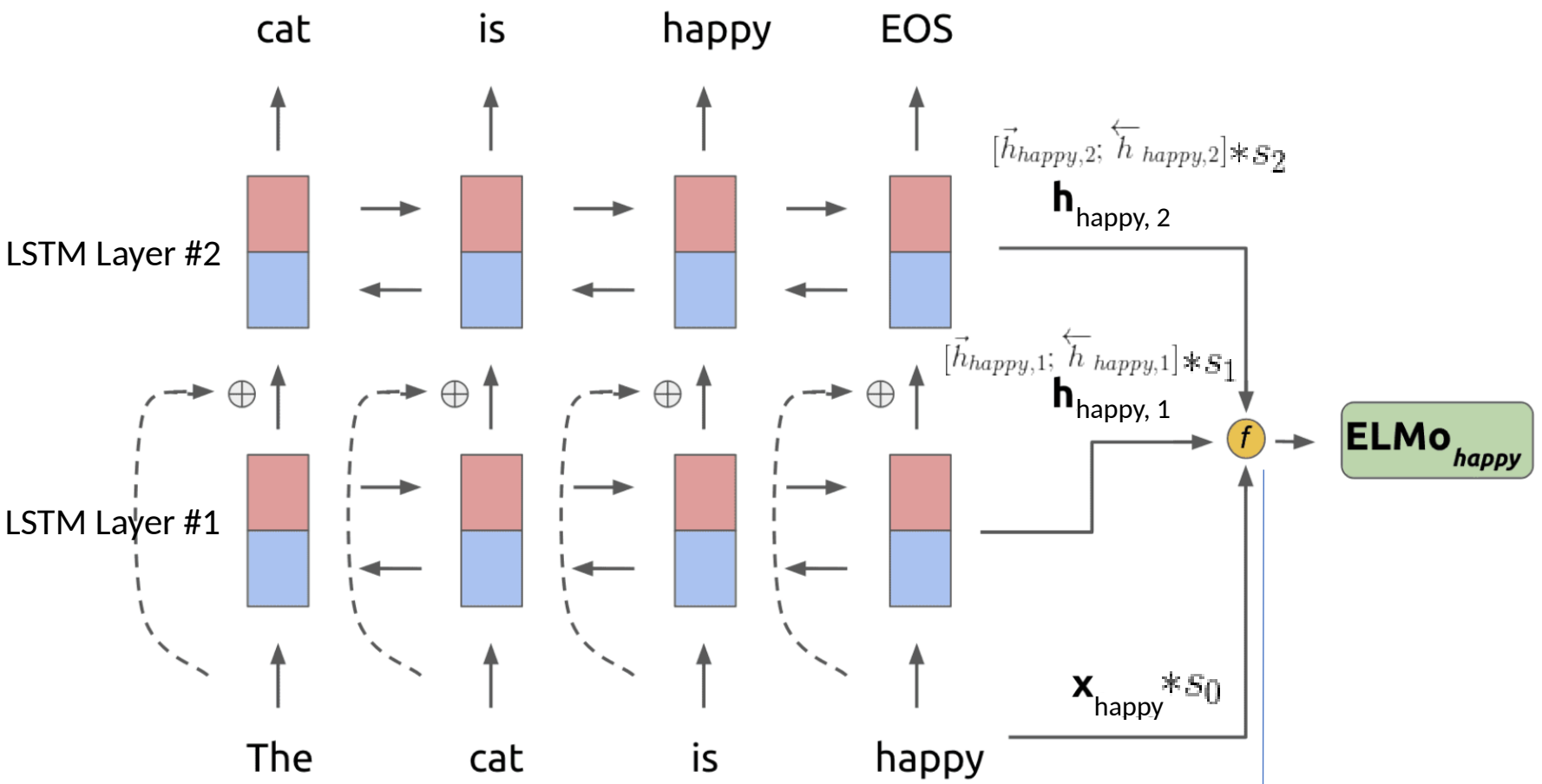
$$= \{h_{k,j} \mid j = 0, \dots, L\}$$

given $h_{k,0} = x_k$ and $h_{k,j} = [\vec{h}_{k,j}; \overleftarrow{h}_{k,j}]$ otherwise

Input to Downstream Task

*Adapted from *"Deep Contextualized Word Representations with ELMo"*

ELMo Architecture: Stage 2: Combine Layer Representations



The representation for input token k is now

$$R_k = \{x_k, \vec{h}_{k,j}, \overleftarrow{h}_{k,j} | \, j = 1, \ldots, L\}$$

$$= \{h_{k,j} | j = 0, \ldots, L\}$$

given $h_{k,0} = x_k$ and $h_{k,j} = [\vec{h}_{k,j}; \overleftarrow{h}_{k,j}]$ otherwise

We can combine the internal representations as a (trainable, weighted) linear combination

$$\text{ELMo}_k^{task} = \gamma^{task} \sum_{j=0}^{L} s_j^{task} h_{k,j}$$

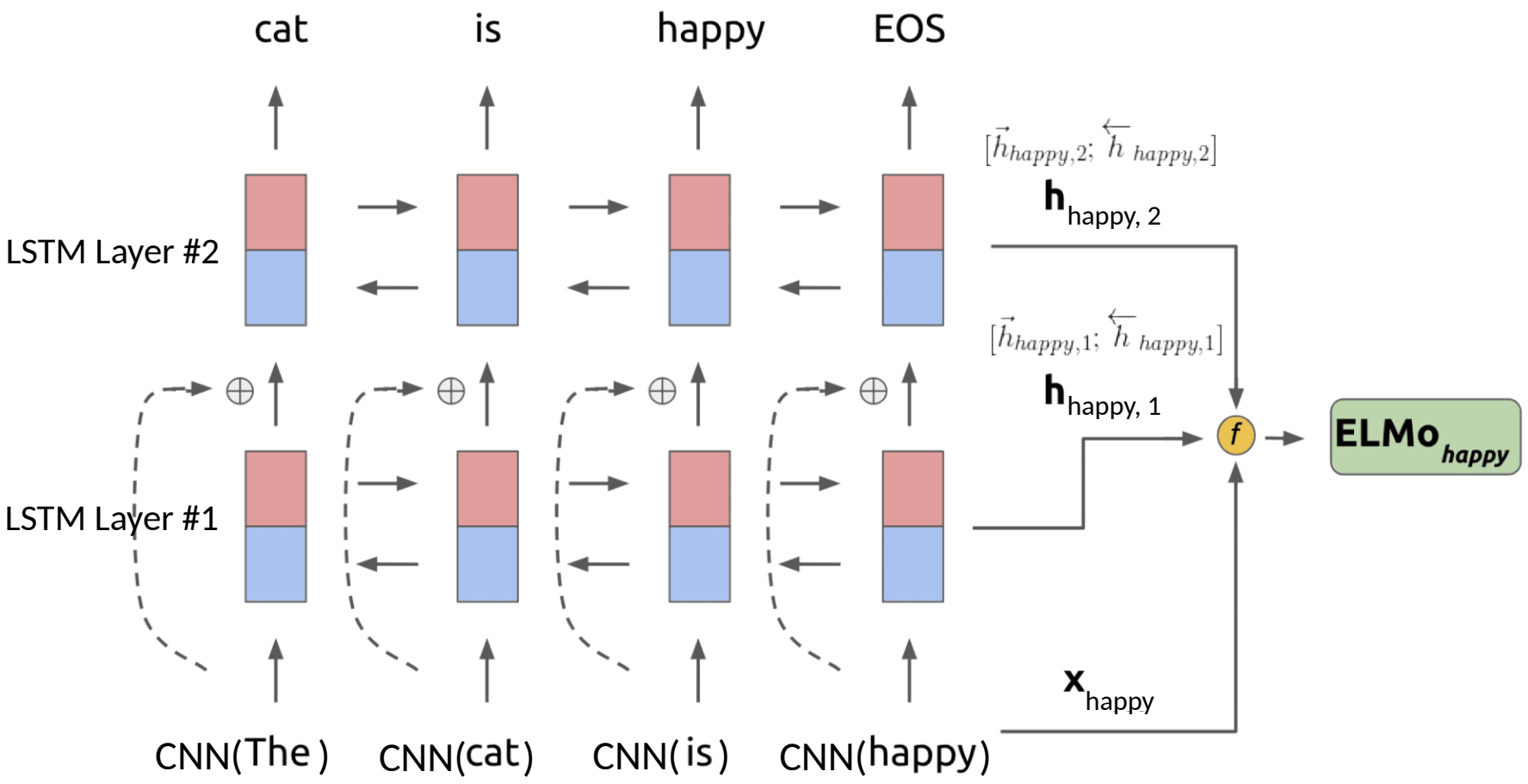
s^{task} are softmax-normalized weights

Task-Specific Trained Parameters

$$ELMo_k^{task} = \gamma_k \cdot (s_0^{task} \cdot x_k + s_1^{task} \cdot h_{\text{k, 1}} + s_2^{task} \cdot h_{\text{k, 2}})$$

*Adapted from *"Deep Contextualized Word Representations with ELMo"*

(extra) ELMo Architecture – charCNNs



The representation for input token k is now

$$R_k = \{x_k, \vec{h}_{k,j}, \overleftarrow{h}_{k,j} | \, j = 1, \ldots, L\}$$

$$= \{h_{k,j} | j = 0, \ldots, L\}$$

given $h_{k,0} = x_k$ and $h_{k,j} = [\vec{h}_{k,j}; \overleftarrow{h}_{k,j}]$ otherwise

We can combine the internal representations as a (trainable, weighted) linear combination

$$\mathrm{ELMo}_k^{task} = \gamma^{task} \sum_{j=0}^{L} s_j^{task} h_{k,j}$$

s^{task} are softmax-normalized weights

charCNN allows us to represent any word, by simply operating over character embeddings

*Adapted from *"Deep Contextualized Word Representations with ELMo"*

Review ELMo Training Steps

1. Pre-train ELMo on 1B Word Benchmark dataset, optimize typical language modeling objective to train bi-LSTM weights
2. Freeze ELMo bi-LSTM weights, train conjunctly with downstream model to train task-specific weights
3. After 2 finishes training, we have task-specific ELMo embeddings

- Summary of Parameters in ELMo model
  - Pre-train: bi-LSTM weights, (charCNN wes_j^{task}s, character embeddγ^{task} using charCNNs)
  - Post-train: Softmax-normalized weights and Global Weight

Evaluating ELMo

Evaluation (Overall)

Task	Previous SOTA		Our baseline	ELMo + baseline	Increase (absolute/ relative)
SQuAD	Liu et al. (2017)	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al. (2017)	88.6	88.0	88.7 ± 0.17	0.7 / 5.8%
SRL	He et al. (2017)	81.7	81.4	84.6	3.2 / 17.2%
Coref	Lee et al. (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al. (2017)	91.93 ± 0.19	90.15	92.22 ± 0.10	2.06 / 21%
SST-5	McCann et al. (2017)	53.7	51.4	54.7 ± 0.5	3.3 / 6.8%

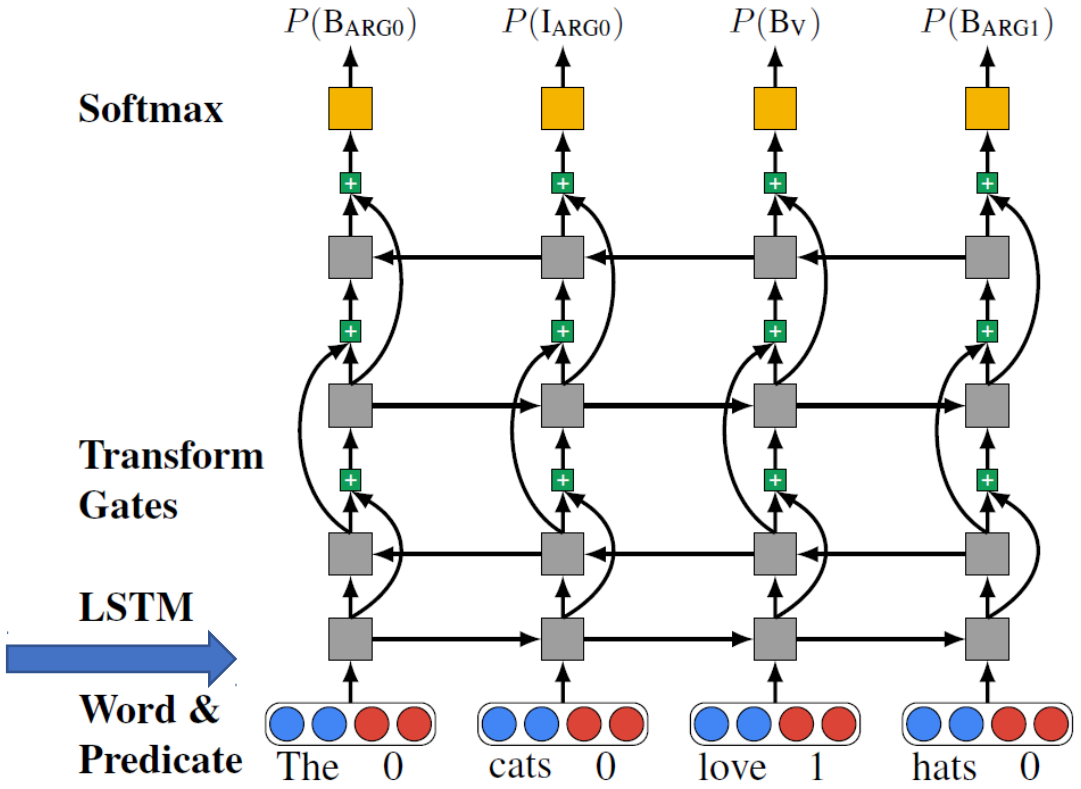
*Table from *"Deep Contextualized Word Representations" (Peters, 2018)*

Semantic Role Labeling

Model	F_1
Pradhan et al. (2013)	77.5
Zhou and Xu (2015)	81.3
He et al. (2017), single	81.7
He et al. (2017), ensemble	83.4
He et al. (2017), our impl.	81.4
He et al. (2017) + ELMo	**84.6**

Table 10: SRL CoNLL 2012 test set F_1.

Add ELMo vectors to the input of the contextual GRU layer



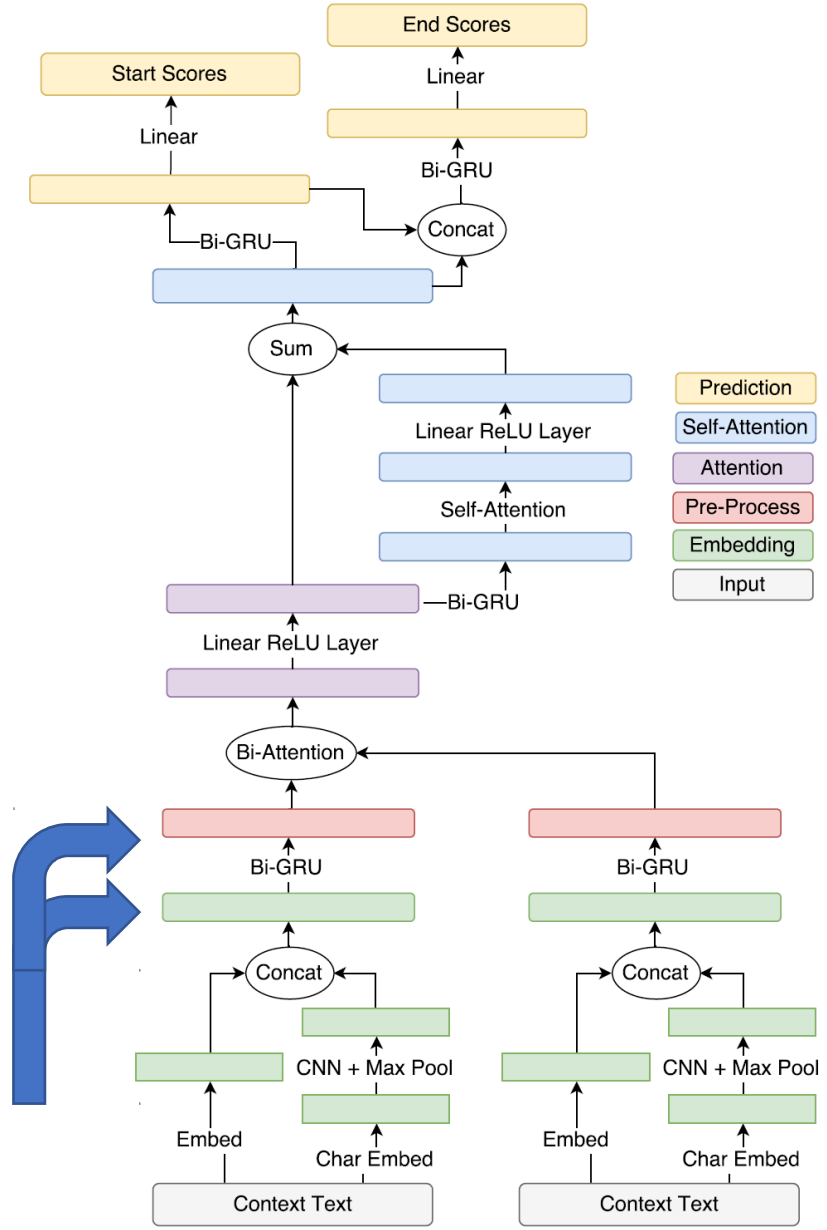
Luheng He, Kenton Lee, Mike Lewis, and Luke S. Zettlemoyer. 2017. Deep semantic role labeling: What works and what's next. In ACL.

*Table from *"Deep Contextualized Word Representations" (Peters, 2018)*

Question Answering

The Stanford Question Answering Dataset (SQuAD)

Model	EM	F_1
BiDAF (Seo et al., 2017)	68.0	77.3
BiDAF + Self Attention	72.1	81.1
DCN+	75.1	83.1
Reg-RaSoR	75.8	83.3
FusionNet	76.0	83.9
r-net (Wang et al., 2017)	76.5	84.3
SAN (Liu et al., 2017)	76.8	84.4
BiDAF + Self Attention + ELMo	**78.6**	**85.8**
DCN+ Ensemble	78.9	86.0
FusionNet Ensemble	79.0	86.0
Interactive AoA Reader+ Ensemble	79.1	86.5
BiDAF + Self Attention + ELMo Ensemble	**81.0**	**87.4**



Add ELMo vectors to both the input and output of the contextual GRU layer

*Table from *"Deep Contextualized Word Representations" (Peters, 2018)*

Christopher Clark and Matthew Gardner. 2017. Simple and effective multi-paragraph reading comprehension. CoRR abs/1710.10723.

Model	Acc.
Feature based (Bowman et al., 2015)	78.2
DIIN (Gong et al., 2018)	88.0
BCN+Char+CoVe (McCann et al., 2017)	88.1
ESIM (Chen et al., 2017)	88.0
ESIM+TreeLSTM (Chen et al., 2017)	88.6
ESIM+ELMo	**88.7** ± 0.17
DIIN ensemble (Gong et al., 2018)	88.9
ESIM+ELMo ensemble	**89.3**

Table 8: SNLI test set accuracy.[3]Single model results occupy the portion, with ensemble results at the bottom.

Textual Entailment

Coreference Resolution

Model	Average F_1
Durrett and Klein (2013)	60.3
Wiseman et al. (2016)	64.2
Clark and Manning (2016)	65.7
Lee et al. (2017) (single)	67.2
Lee et al. (2017) (ensemble)	68.8
Lee et al. (2017) + ELMo	**70.4**

Table 11: Coreference resolution average F_1 on the test set from the CoNLL 2012 shared task.

Name Entity Recognition

Model	F_1 ± std.
Collobert et al. (2011)♣	89.59
Lample et al. (2016)	90.94
Ma and Hovy (2016)	91.2
Chiu and Nichols (2016)♣,♢	91.62 ± 0.33
Peters et al. (2017)♢	91.93 ± 0.19
biLSTM-CRF + ELMo	**92.22** ± **0.10**

Table 12: Test set F_1 for CoNLL 2003 NER task. Models with ♣ included gazetteers and those with ♢ used both the train and development splits for training.

Sentiment Classification

Model	Acc.
DMN (Kumar et al., 2016)	52.1
LSTM-CNN (Zhou et al., 2016)	52.4
NTI (Munkhdalai and Yu, 2017)	53.1
BCN+Char+CoVe (McCann et al., 2017)	53.7
BCN+ELMo	**54.7**

Table 13: Test set accuracy for SST-5.

*Tables from *"Deep Contextualized Word Representations" (Peters, 2018)*

Analyzing ELMo

What information is captured?

- Case example of Word sense

	Source	Nearest Neighbors
GloVe	play	playing, game, games, played, players, plays, player, Play, football, multiplayer
biLM	Chico Ruiz made a spectacular play on Alusik 's grounder {...}	Kieffer , the only junior in the group , was commended for his ability to hit in the clutch , as well as his all-round excellent play .
	Olivia De Havilland signed to do a Broadway play for Garson {...}	{...} they were actors who had been handed fat roles in a successful play , and had talent enough to fill the roles competently , with nice understatement .

Table 4: Nearest neighbors to "play" using GloVe and the context embeddings from a biLM.

*Table from *"Deep Contextualized Word Representations" (Peters, 2018)*

Word sense disambiguation

In our house, everybody has a career and none of them includes washing **dishes**.

I'm looking for a restaurant that serves vegetarian **dishes**.

- Most words have multiple senses
- Task: determine which of various senses of a word are invoked in context

Model	**F_1**
WordNet 1st Sense Baseline	65.9
Raganato et al. (2017a)	69.9
Iacobacci et al. (2016)	**70.1**
CoVe, First Layer	59.4
CoVe, Second Layer	*64.7*
biLM, First layer	67.4
biLM, Second layer	*69.0*

Table 5: All-words fine grained WSD F_1. For CoVe and the biLM, we report scores for both the first and second layer biLSTMs.

*Table from *"Deep Contextualized Word Representations" (Peters, 2018)*

Part-of-speech (POS) tagging

Model	Acc.
Collobert et al. (2011)	97.3
Ma and Hovy (2016)	97.6
Ling et al. (2015)	**97.8**
CoVe, First Layer	*93.3*
CoVe, Second Layer	92.8
biLM, First Layer	*97.3*
biLM, Second Layer	96.8

Table 6: Test set POS tagging accuracies for PTB. For CoVe and the biLM, we report scores for both the first and second layer biLSTMs.

*Table from *"Deep Contextualized Word Representations" (Peters, 2018)*

Advantages / Disadvantages of ELMo

Advantages

- Contextualized: more expressive
- Bidirectional: capturing both sides of contextual information
- "Deep": combination of multiple layers
- Unsupervised pretraining (LM), do not need parallel corpus like with CoVe.
- Can represent out-of-vocabulary words due to characater CNNs + contextual information
- Sample efficiency (transfer learning)

Sample efficiency

- Number of parameter updates to reach state-of-the-art performance ⬇
- The training set size ⬇

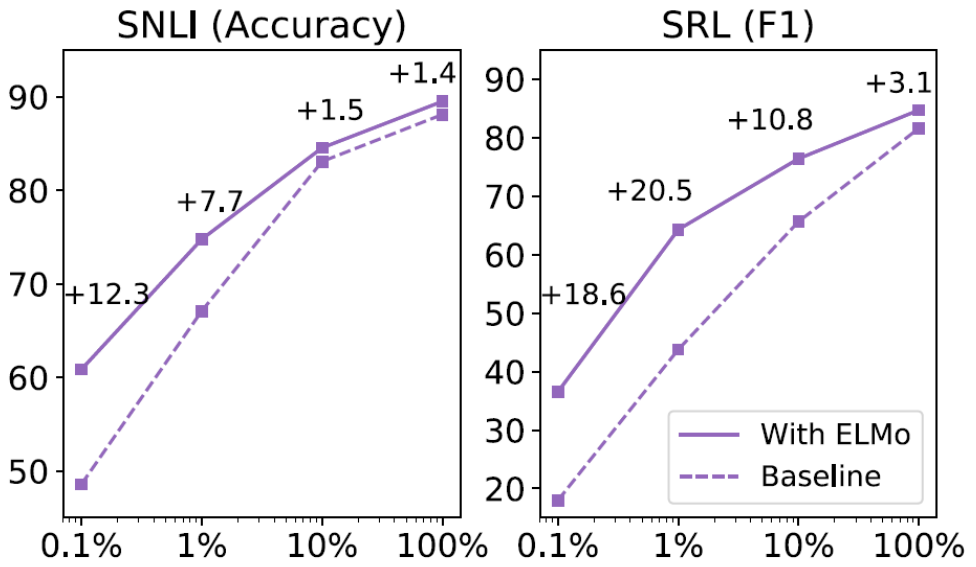


Figure 1: Comparison of baseline vs. ELMo performance for SNLI and SRL as the training set size is varied from 0.1% to 100%.

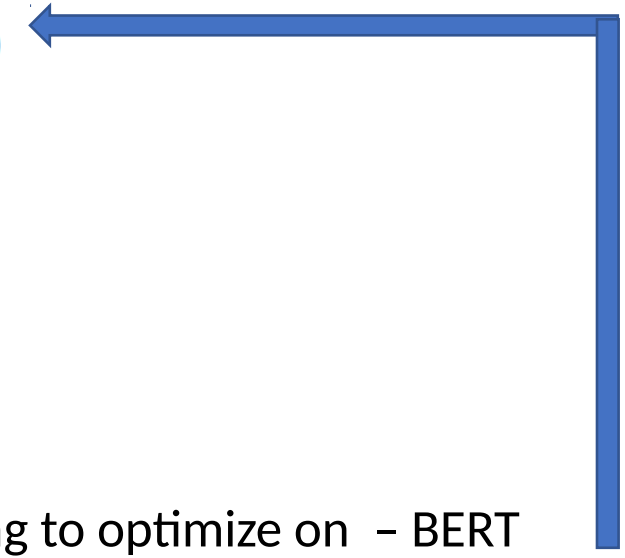
*Graphs from *"Deep Contextualized Word Representations" (Peters, 2018)*

Disadvantages / Concluding Thoughts

- Intrinsic drawbacks of traditional language modeling:

  Unable to learn bidirectional contexts

$$p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t \mid \mathbf{x}_{<t})$$



- Use LSTM instead of Transformer

How to improve?

- Choose an objective (Cloze task) other than language modeling to optimize on – BERT
- Maximize the expected likelihood over all permutations of the factorization order – XLNet
- Use Transformer -- GPT

References

- Christopher Clark and Matthew Gardner. 2017. *Simple and effective multi-paragraph reading comprehension*. CoRR abs/1710.10723.
  - https://arxiv.org/pdf/1710.10723.pdf
- Luheng He, Kenton Lee, Mike Lewis, and Luke S. Zettlemoyer. 2017. *Deep semantic role labeling: What works and what's next*. In ACL.
  - https://www.aclweb.org/anthology/P17-1044
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer. 2018. *Deep Contextualized Word Representations*. NAACL 2018
  - https://arxiv.org/pdf/1802.05365.pdf
- *Deep Contextualized Word Representations with ELMo*
  - https://medium.com/@saurabh.rathor092/simple-rnn-vs-gru-vs-lstm-difference-lies-in-more-flexible-control-5f33e07b1e57
- CS 224n slides "Deep Learning for NLP" at Stanford
  - Lecture 6: http://web.stanford.edu/class/cs224n/slides/cs224n-2019-lecture06-rnnlm.pdf
  - Lecture 8: https://web.stanford.edu/class/cs224n/slides/cs224n-2019-lecture08-nmt.pdf
- CS 124 slides "From Languages to Information" at Stanford
  - Lecture on Language Modeling: https://web.stanford.edu/class/cs124/lec/languagemodeling.pdf
- *Simple RNN vs GRU vs LSTM :- Differences Lies in More Flexible Control*
  - https://medium.com/@saurabh.rathor092/simple-rnn-vs-gru-vs-lstm-difference-lies-in-more-flexible-control-5f33e07b1e57
- *Deep Dive into Bi-directional LSTMs*
  - https://www.i2tutorials.com/technology/deep-dive-into-bidirectional-lstm/