# Deep Contextual Word Representations (ELMo)

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

Presented by Debapriyo Majumdar
debapriyo@isical.ac.in

# Contextual word embeddings

- Word embeddings: *learned* vector representations
  - Models complex characteristics of syntax and semantics of the words
- Challenge: polysemy
  - Words need different representation depending on the context
  - Example 1:
    - One **apple** a day keeps the doctor away.
    - iPhone 13 should be the next big launch for **Apple**.
  - Example 2:
    - It was the last of Olivier's appearances in a Shakespeare **play**.
    - Let's **play** football.
- ELMo: Embeddings from Language Models
- Entire input sentence $\mapsto$ embeddings for input token

# Related previous works

- Context independent pre-trained Word embeddings
  - Word2Vec (Mikolov et al., 2013)
  - GloVe (Pennington et al., 2014)
- Learning separate vectors for each word sense
  - Neelakantan et al., 2014
- Context-dependent representations
  - Context2vec (Melamud et al., 2016)
    - Bidirectional LSTM to encode the context around a pivot word
  - CoVe (McCann et al., 2017)
    - Two-layer bidirectional LSTM attentional model for translation
    - Use the encoder to provide context to other NLP tasks
    - Entire input sentence $\mapsto$ embeddings for input token
  - TagLM (Peters et al., 2017)
    - Previous work by the same leading author

# Bidirectional Language Model (biLM)

- Forward language model: given a sequence of $N$ tokens $(t_1, t_2, \ldots, t_N)$, compute

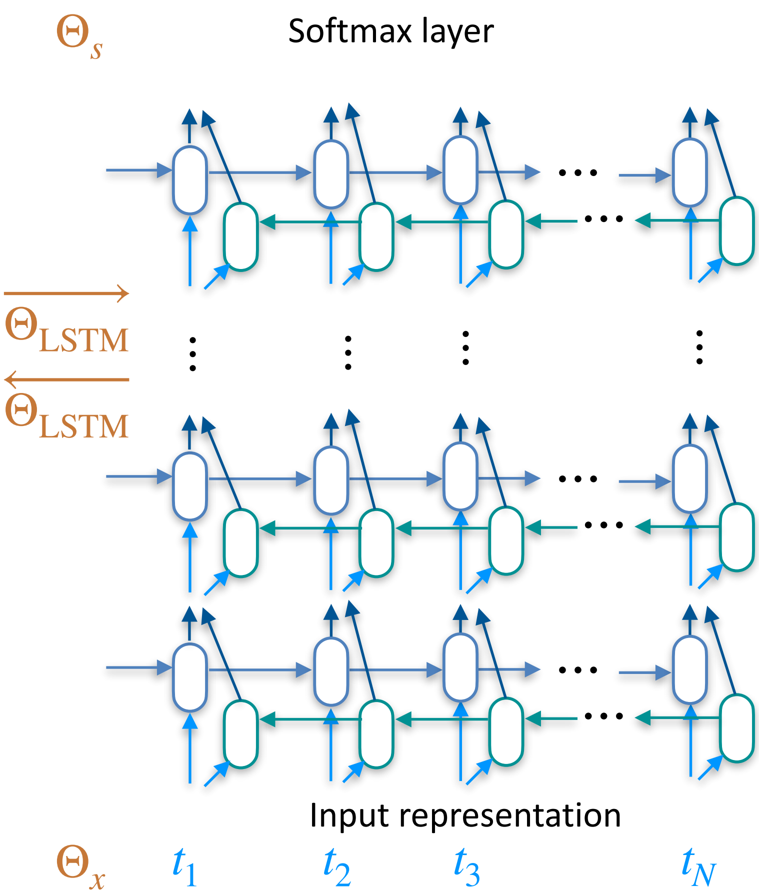
$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{n} p(t_k \mid t_1, t_2, \ldots, t_{k-1})$$

  - Can be learnt by an $L$-layer forward LSTM
- Backward language model: predict the previous token given the future context

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{n} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

  - An $L$-layer backward LSTM

# Bidirectional Language Model (biLM)

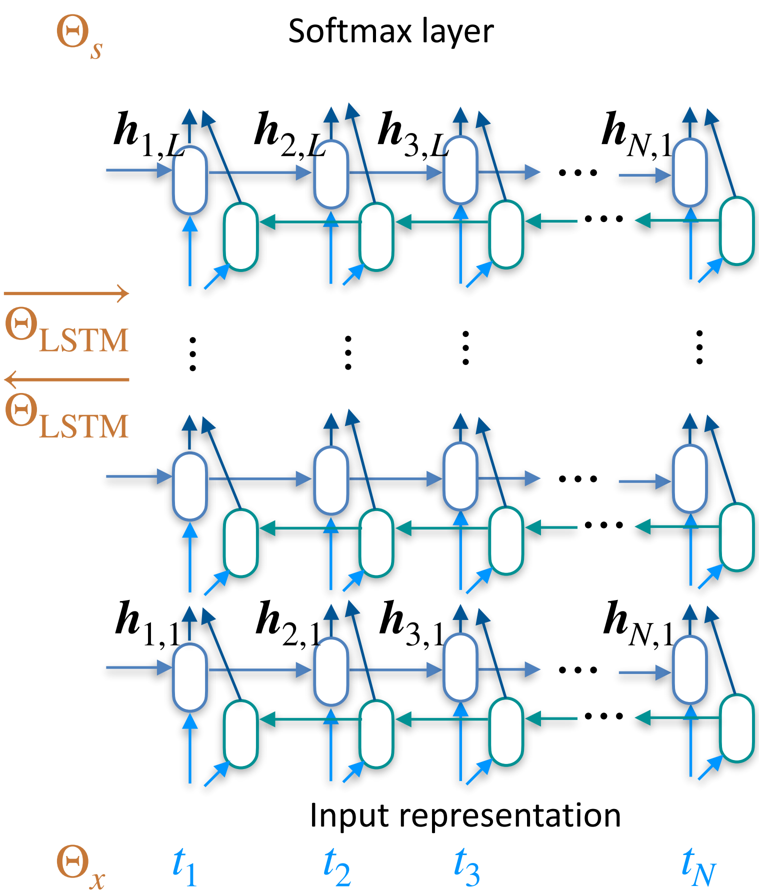



- Combine the forward and the backward LMs to construct the **biLM**
- Jointly maximize the log-likelihood:

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

- The forward and backward LSTMs have separate parameters
- The parameters for the input representation and softmax layers are the same

# Representation of tokens by ELMo




- The representation for the $k$-th token $t_k$

  $$R_k = \{\boldsymbol{x}_k, \overrightarrow{\boldsymbol{h}}_{k,j}, \overleftarrow{\boldsymbol{h}}_{k,j}, j = 1,\ldots,L\}$$
  $$= \{\boldsymbol{h}_{k,j} \,|\, j = 1,\ldots,L\}\,.$$

  Where $\boldsymbol{x}_k = \boldsymbol{h}_{0,j}$ is the input representation

- ELMo embedding: collapse all layers of $R$ into a single vector
  - $\text{ELMo}_k = E(R_k; \Theta_e)$
  - Task specific weighting of the layers

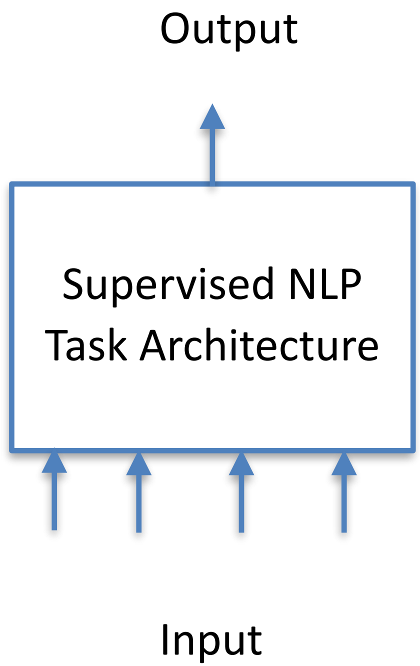
$$\text{ELMo}_k^{\text{task}} = E(R_k; \Theta^{\text{task}}) = \gamma^{\text{task}} \sum_{j=0}^{L} s_j^{\text{task}} \boldsymbol{h}_{k,j}$$

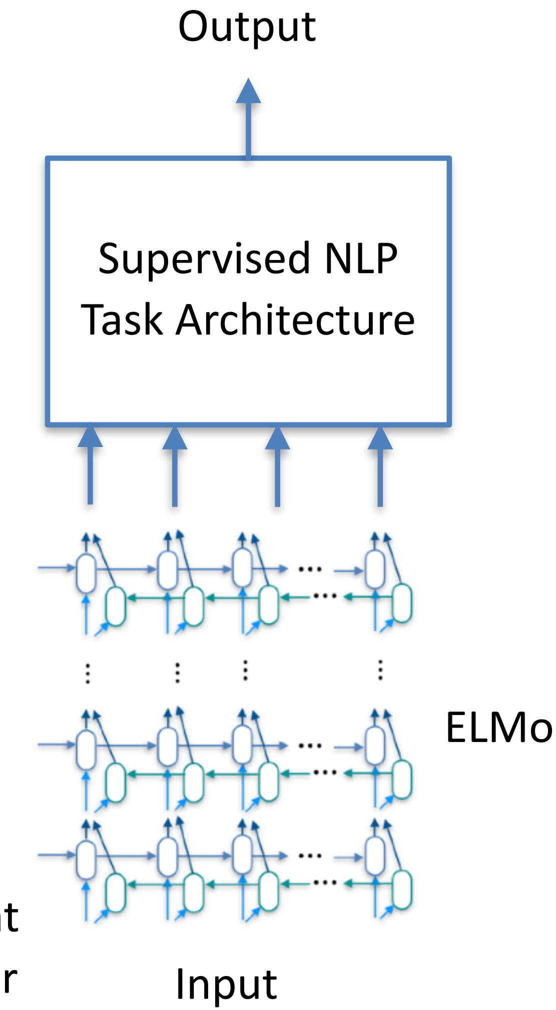
$s_j^{\text{task}}$ are softmax-normalized weights

$\gamma^{\text{task}}$ is important for optimization process

# ELMo in a supervised NLP task

ELMo can be used as an add-on
to any downstream NLP task




For some tasks adding another ELMo at
the output improves the results further

# Experimental results

- SQuAD dataset
  - About 100K+ crowd-sourced QA pairs
  - Answer is a span in a Wikipedia paragraph
  - Baseline: bidirectional attention flow model (Seo et al., 2017)
  - Improvement after adding ELMo: improves from 81.1% to 85.8%
  - CoVe achieved a 1.8% improvement
- SNLI Corpus
  - Textual entailment: determine whether a hypothesis is true, given a premise
  - About 550K hypothesis/premise pairs
  - Baseline: BiLSTM + matrix attention + local inference + BiLSTM composition + pooling (Chen et al., 2017)
  - Improvement after adding ELMo: 0.7% absolute improvement

# Results on other tasks

| 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 | $88.7 \pm 0.17$ | 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) | $91.93 \pm 0.19$ | 90.15 | $92.22 \pm 0.10$ | 2.06 / 21% |
| SST-5 | McCann et al. (2017) | 53.7 | 51.4 | $54.7 \pm 0.5$ | 3.3 / 6.8% |

- Improvement on most standard NLP tasks by adding ELMo to a baseline model

# Contextual embedding: demonstration

| | Source | Nearest Neighbors |
|---|---|---|
| GloVe | play | playing, game, games, played, players, plays, player, Play, football, multiplayer |
| biLM | Chico Ruiz made a spectacular <u>play</u> 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 <u>play</u> . |
| | Olivia De Havilland signed to do a Broadway <u>play</u> for Garson {. . . } | {. . . } they were actors who had been handed fat roles in a successful <u>play</u> , 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.