Transfer Learning

ELMo (Peters et al., 2018)

Learning goals

- tbd

CONTEXTUAL EMBEDDINGS

Source: *Jay Alammar*

ELMO

▸ PETERS ET AL., 2018

- Bidirectional language model (LM)
- Combines a forward LM

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

and a backward LM

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

to arrive at the following loglikelihood:

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

ELMO EMBEDDINGS

- Character-based (context-independent) token representations

$$x_k^{LM}$$

- Two-layer biLSTM as main architecture:
  - Two context-dependent token representations *per layer*, i.e.

$$\overrightarrow{\mathbf{h}}_{k,j}^{LM} \text{ \& } \overleftarrow{\mathbf{h}}_{k,j}^{LM} \text{ for the } k\text{-th token in the } j\text{-th layer.}$$

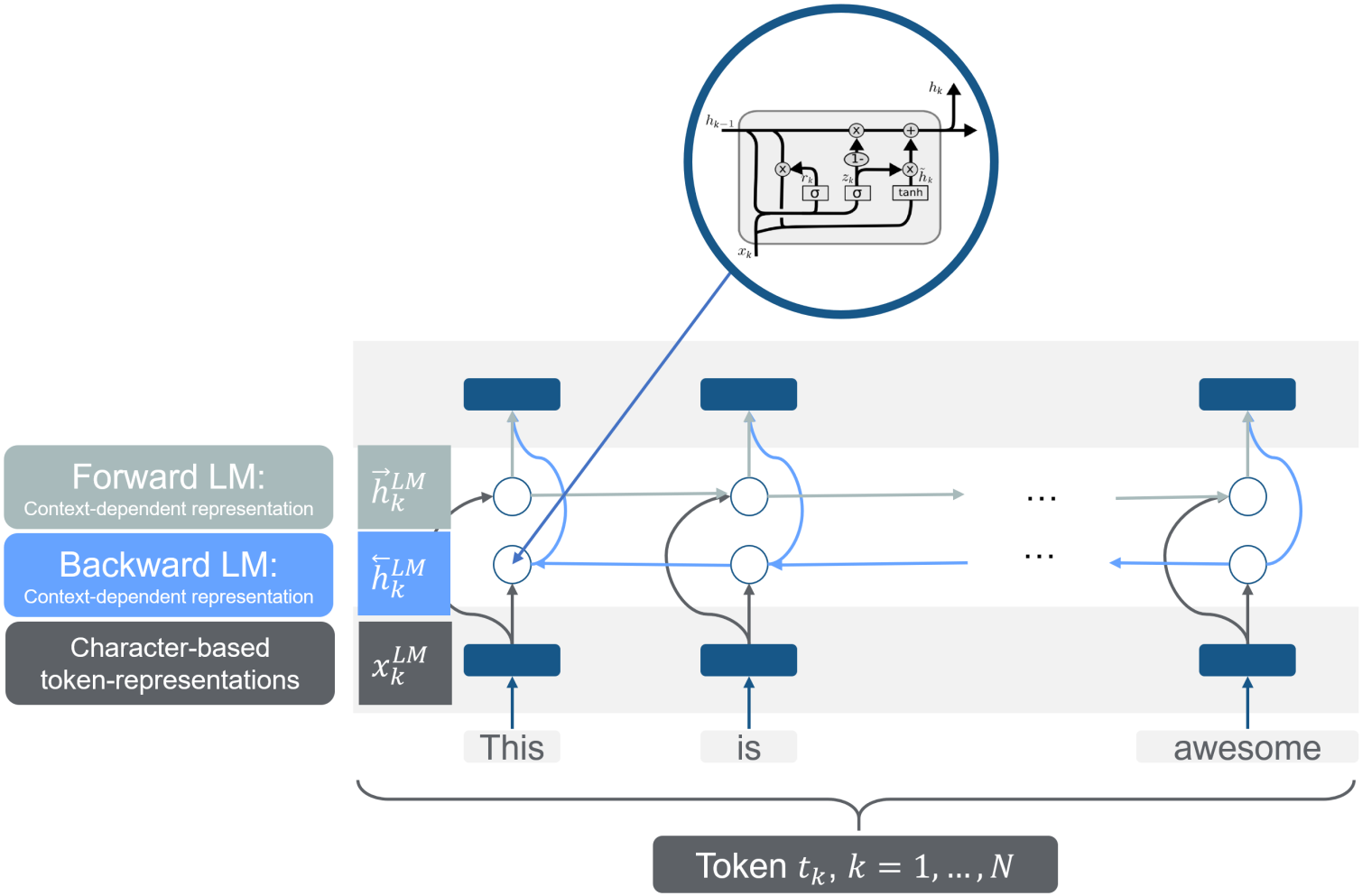
  - Four context-dependent token representations in total:

$$\left\{ \overrightarrow{\mathbf{h}}_{k,j}^{LM}, \overleftarrow{\mathbf{h}}_{k,j}^{LM} | j = 1, 2 \right\}$$

- Five representations per token in total:

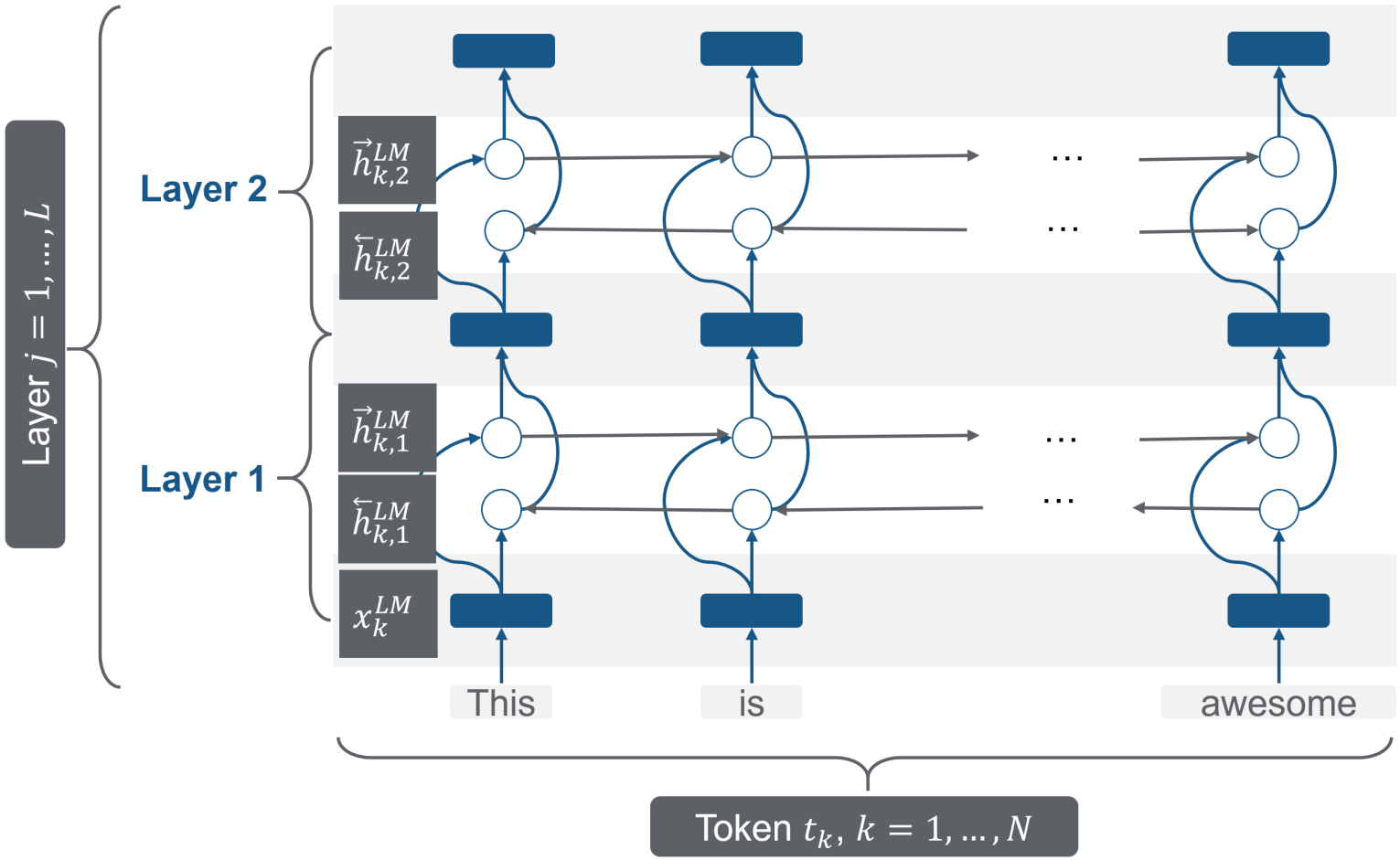
$$\begin{aligned} R_k &= \left\{ \mathbf{x}_k^{LM}, \overrightarrow{\mathbf{h}}_{k,j}^{LM}, \overleftarrow{\mathbf{h}}_{k,j}^{LM} | j = 1, \ldots, L \right\} \\ &= \left\{ \mathbf{h}_{k,j}^{LM} | j = 0, 1, 2 \right\} \end{aligned}$$

GRAPHICAL REPRESENTATION



Source: *Carolin Becker*

GRAPHICAL REPRESENTATION



Source: *Carolin Becker*

TASK ADAPTION

Including ELMo in downstream tasks:

- Calculate task-specific weights of all five representations:

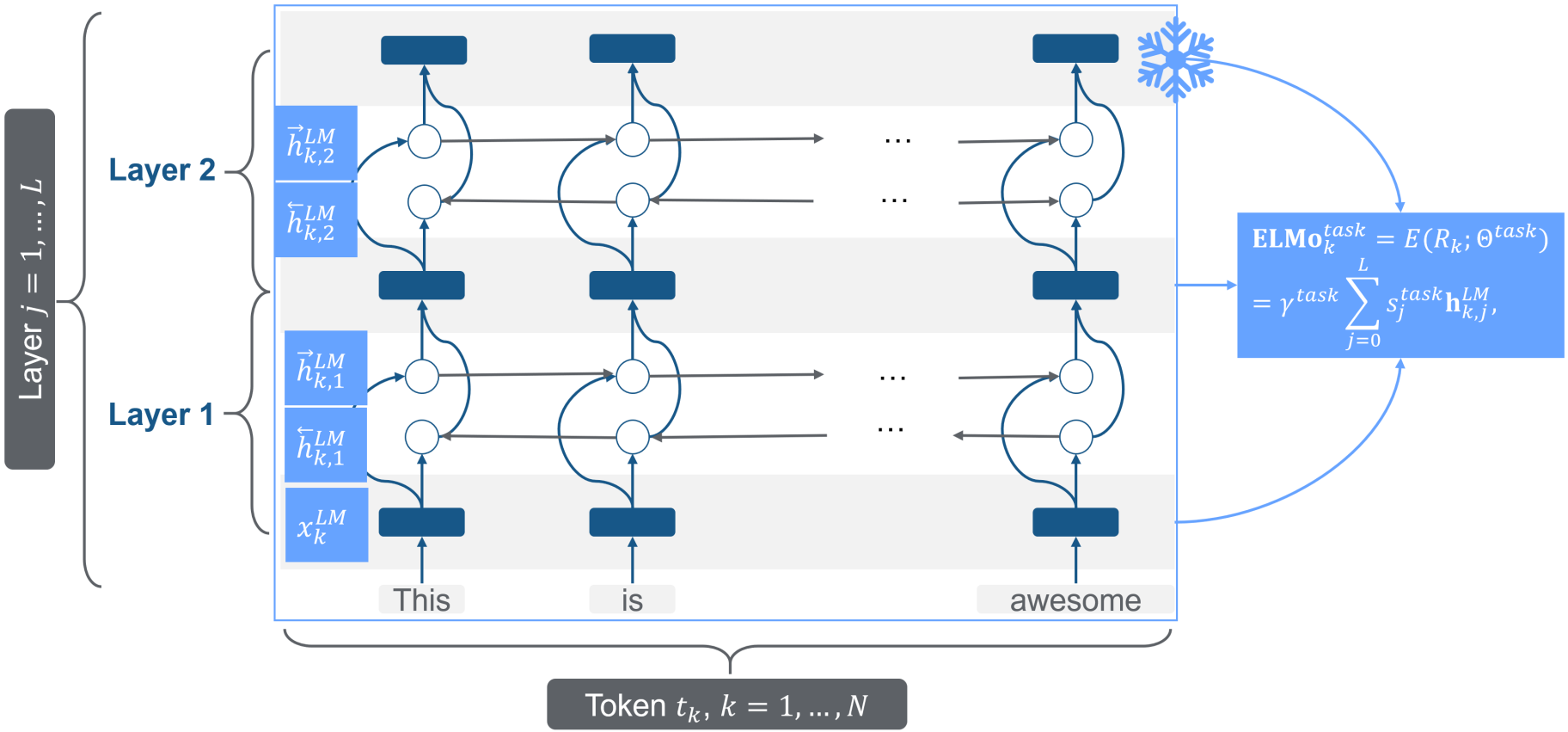
$$\mathbf{ELMo}_k^{task} = E\left(R_k; \Theta^{task}\right) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} \mathbf{h}_{k,j}^{LM},$$

  where the $\mathbf{h}_{k,j}^{LM}$ are **not trainable** anymore.
- Trainable parameters during the adaption:
  - s_j^{task} are trainable (softmax-normalized) weights
  - γ^{task} is a trainable scaling parameter

Advantages over context free-embeddings:

- Task-specific model has access to *multiple* representations of each token
- Model learns to which degree to use the different representations depending on the task at hand

TASK ADAPTION



Source: *Carolin Becker*

FINE-TUNING APPROACH

- Pre-trained on a general domain corpus
- Embeddings are contextualized (as opposed to word2vec)
- Embeddings are not adapted to target domain/task (similar to word2vec)
- Sequential nature of LSTMs:
  - Not fully parallelizable (compared to Transformers, see next chapter)
  - Fails to capture long-range dependency during contextualization