Detecting Local Insights from Global Labels: Supervised & Zero-Shot Sequence Labeling via a Convolutional Decomposition

Allen Schmaltz

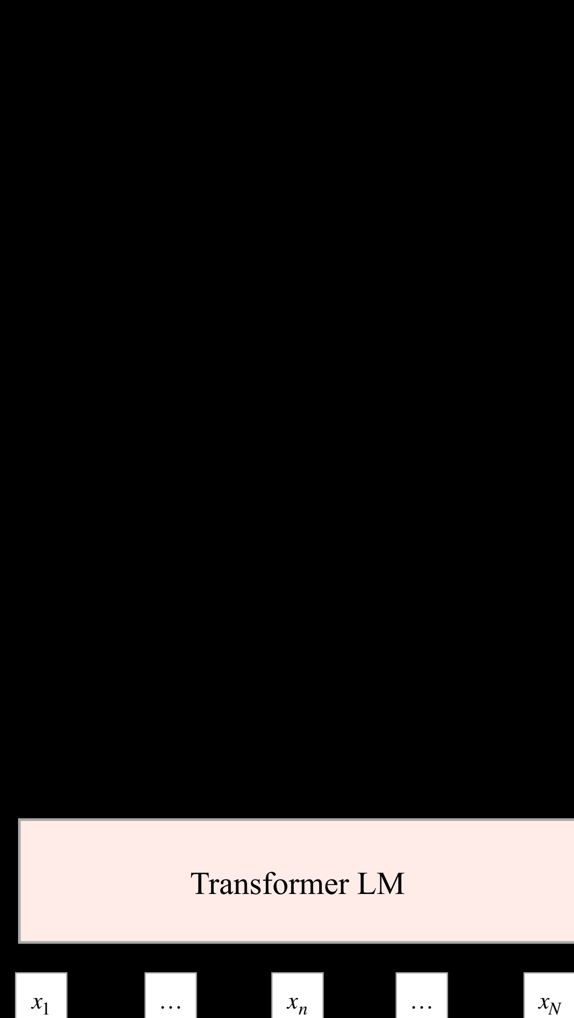
Harvard University

Presented at EMNLP, November 2021

Computational Linguistics (2021)

Version 1.0

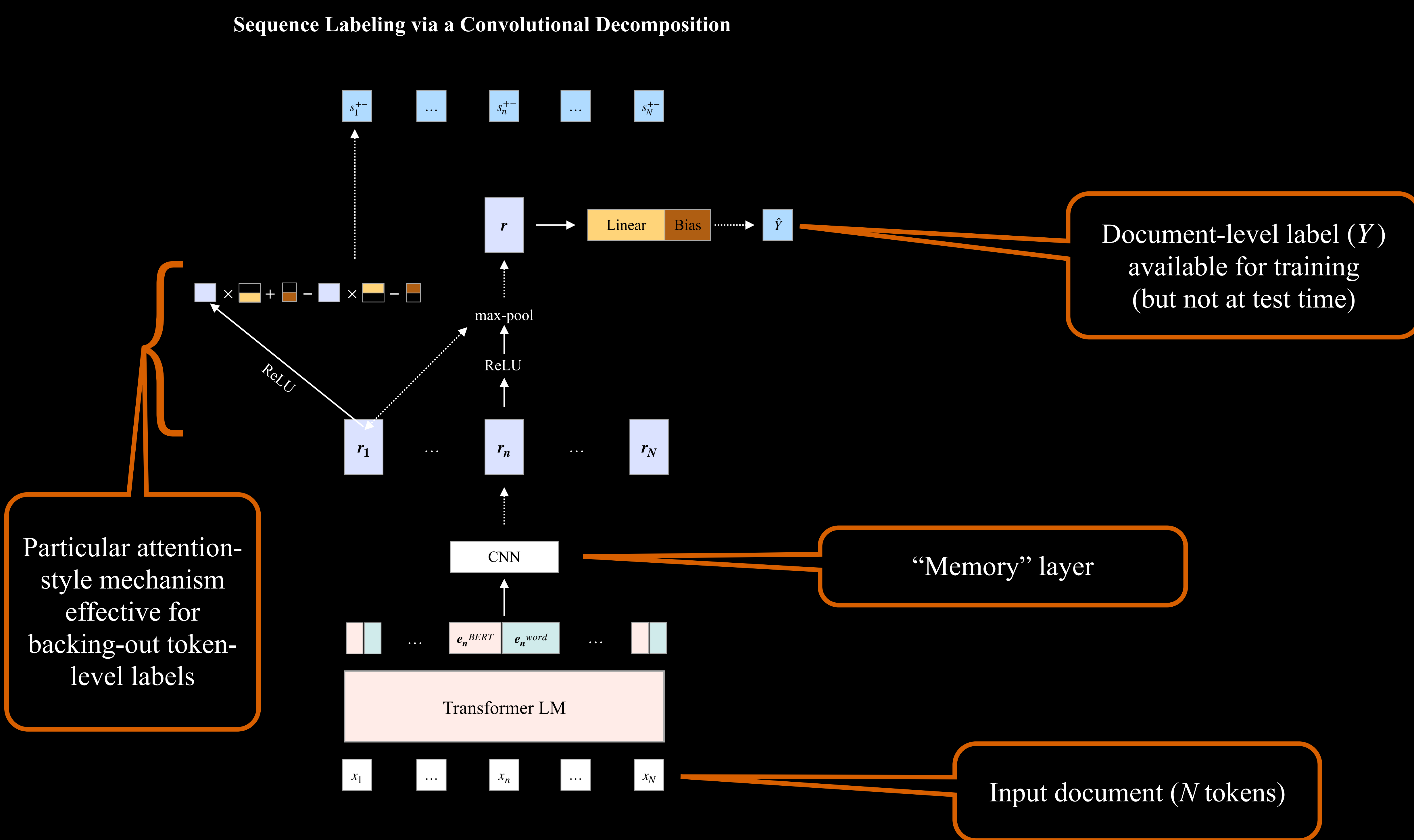
Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions



Document-level label (Y) available for training (but not at test time)

Input document (N tokens)

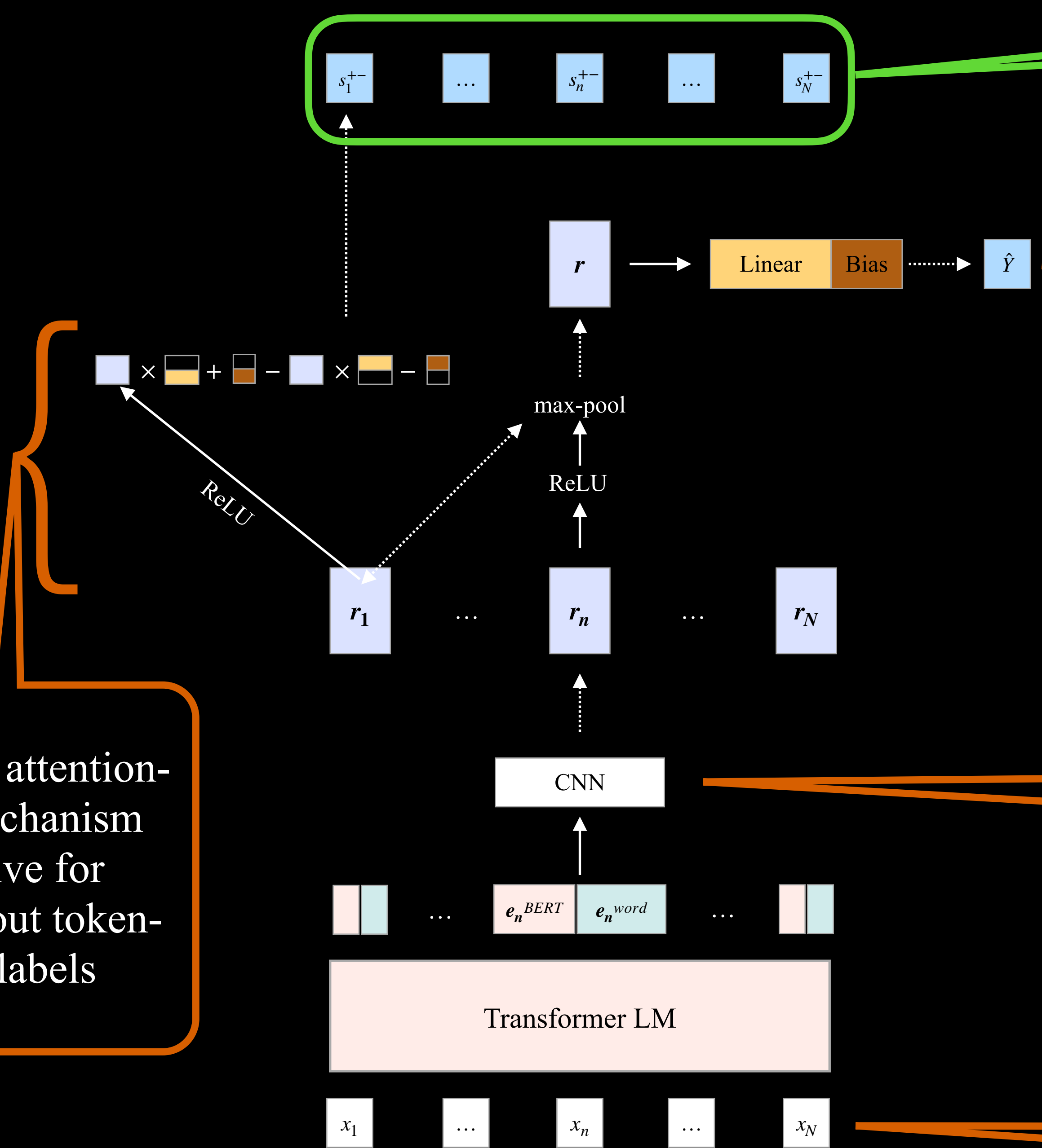
Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions



Allen Schmaltz

Horizontal (across the input) & *Vertical* (across the support set)
Model Decompositions

Sequence Labeling via a Convolutional Decomposition



Predict: token-level labels (y_n)
(Hidden from training and test)

Document-level label (Y)
available for training
(but not at test time)

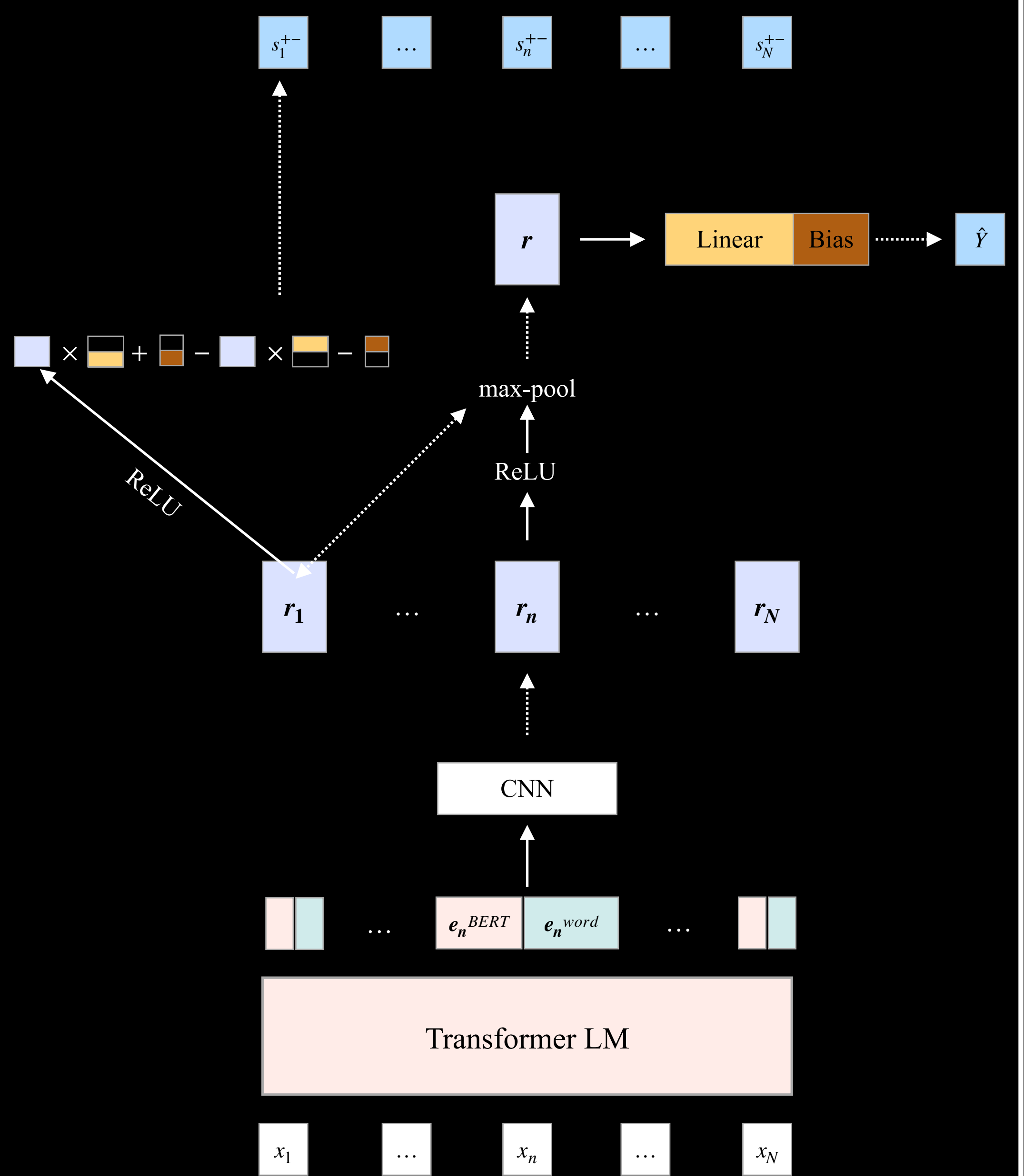
Particular attention-style mechanism effective for backing-out token-level labels

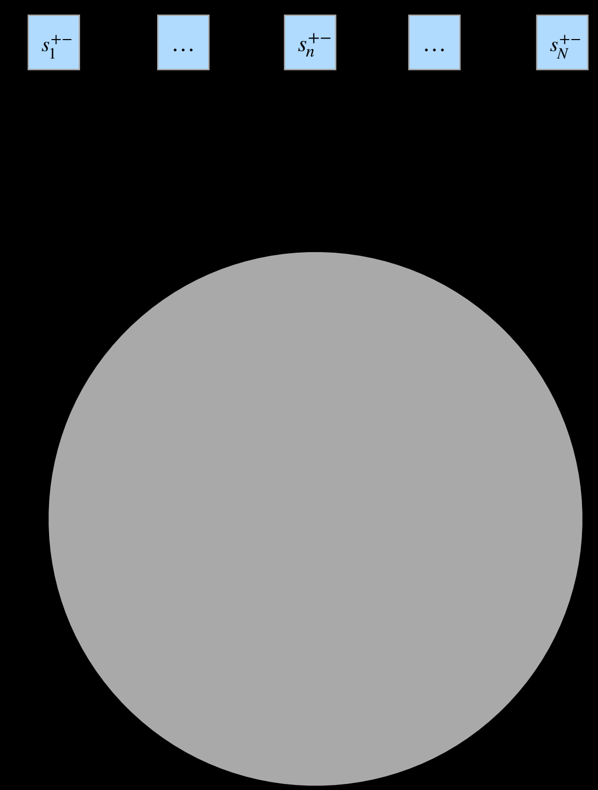
"Memory" layer

Input document (N tokens)

Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions

Sequence Labeling via a Convolutional Decomposition

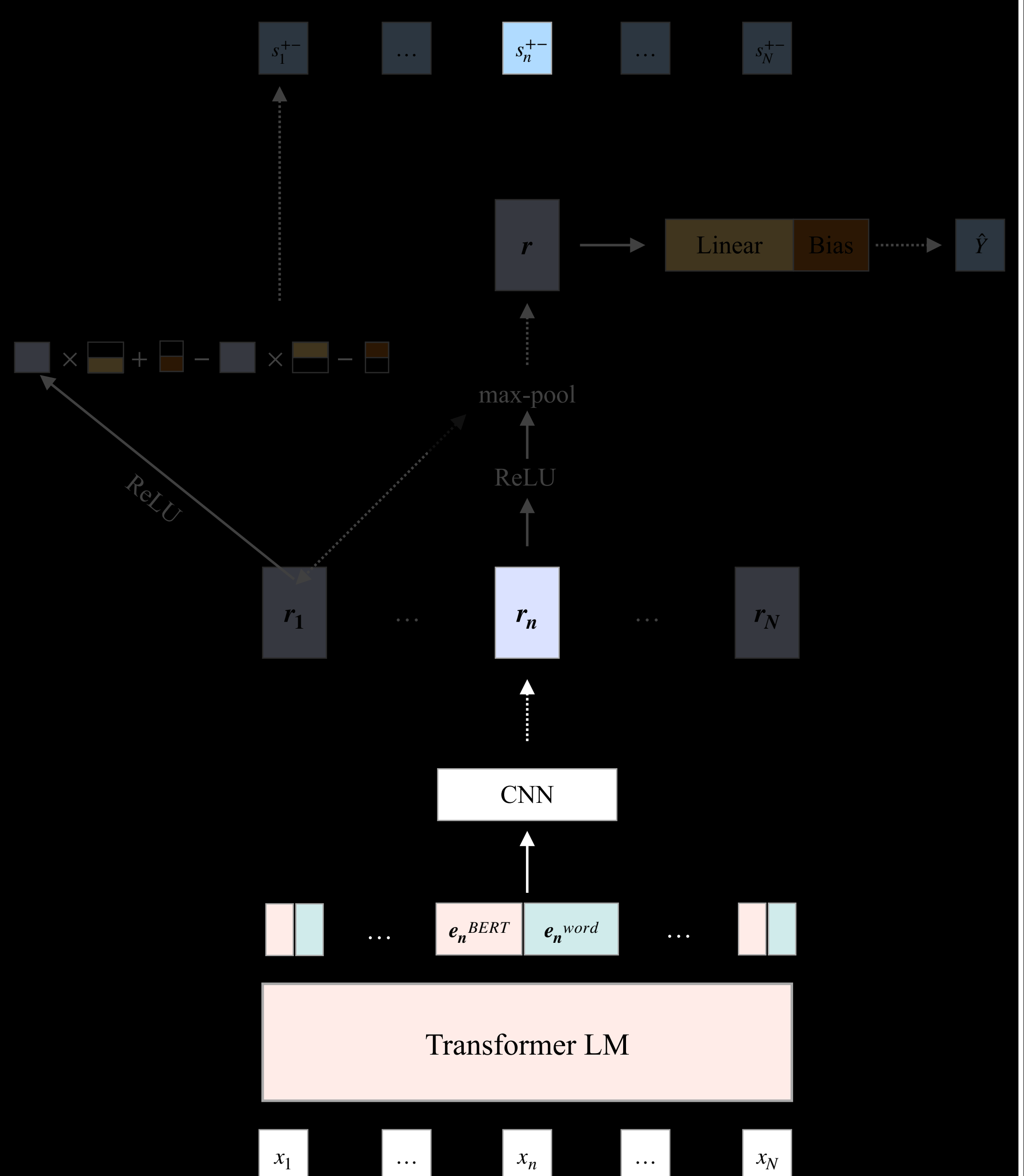


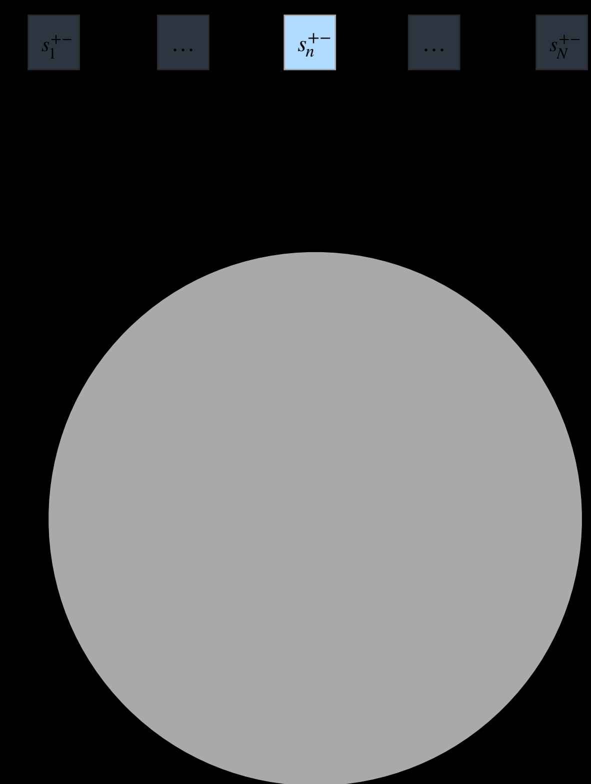


Allen Schmaltz

Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions

Sequence Labeling via a Convolutional Decomposition

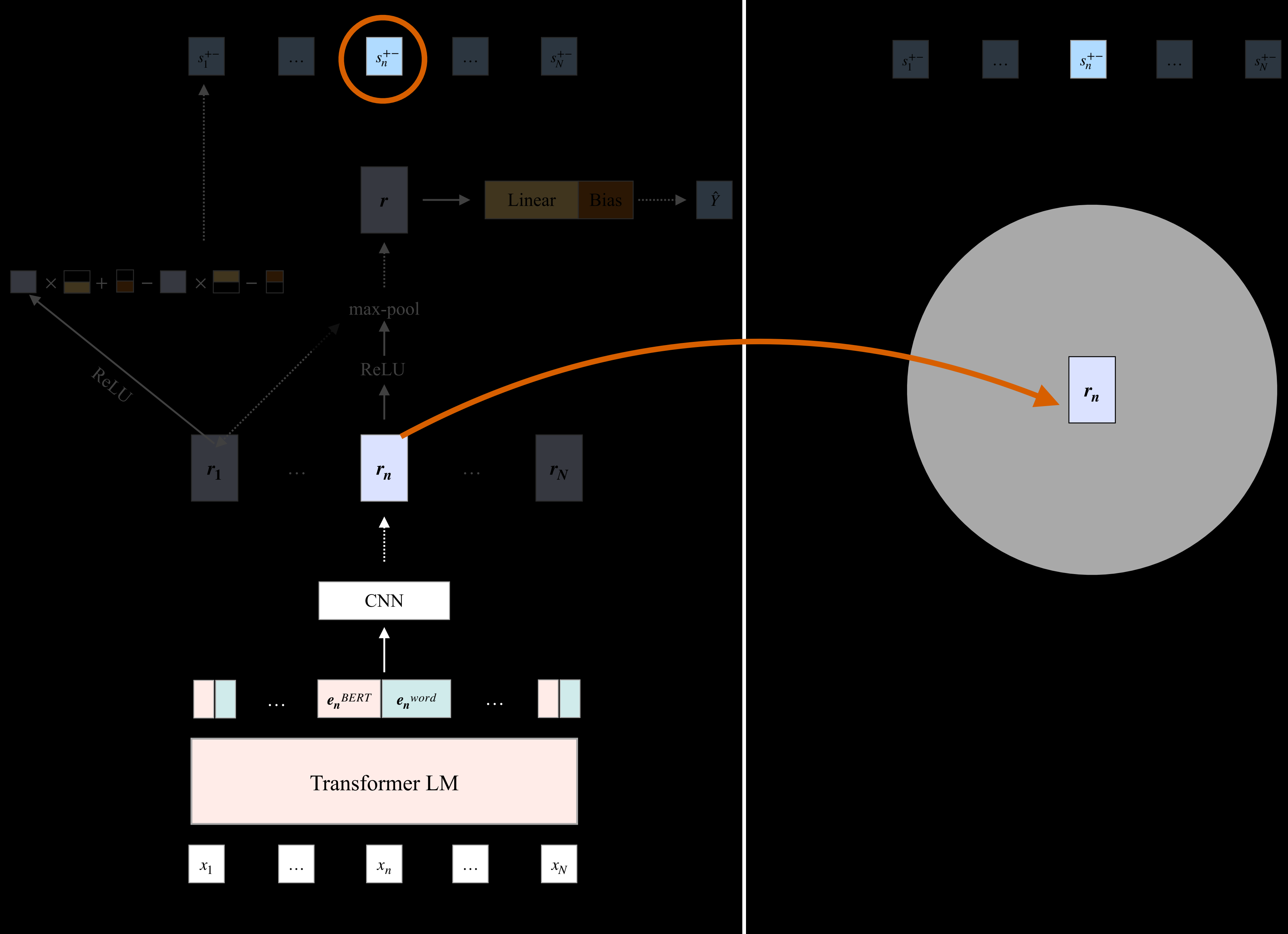




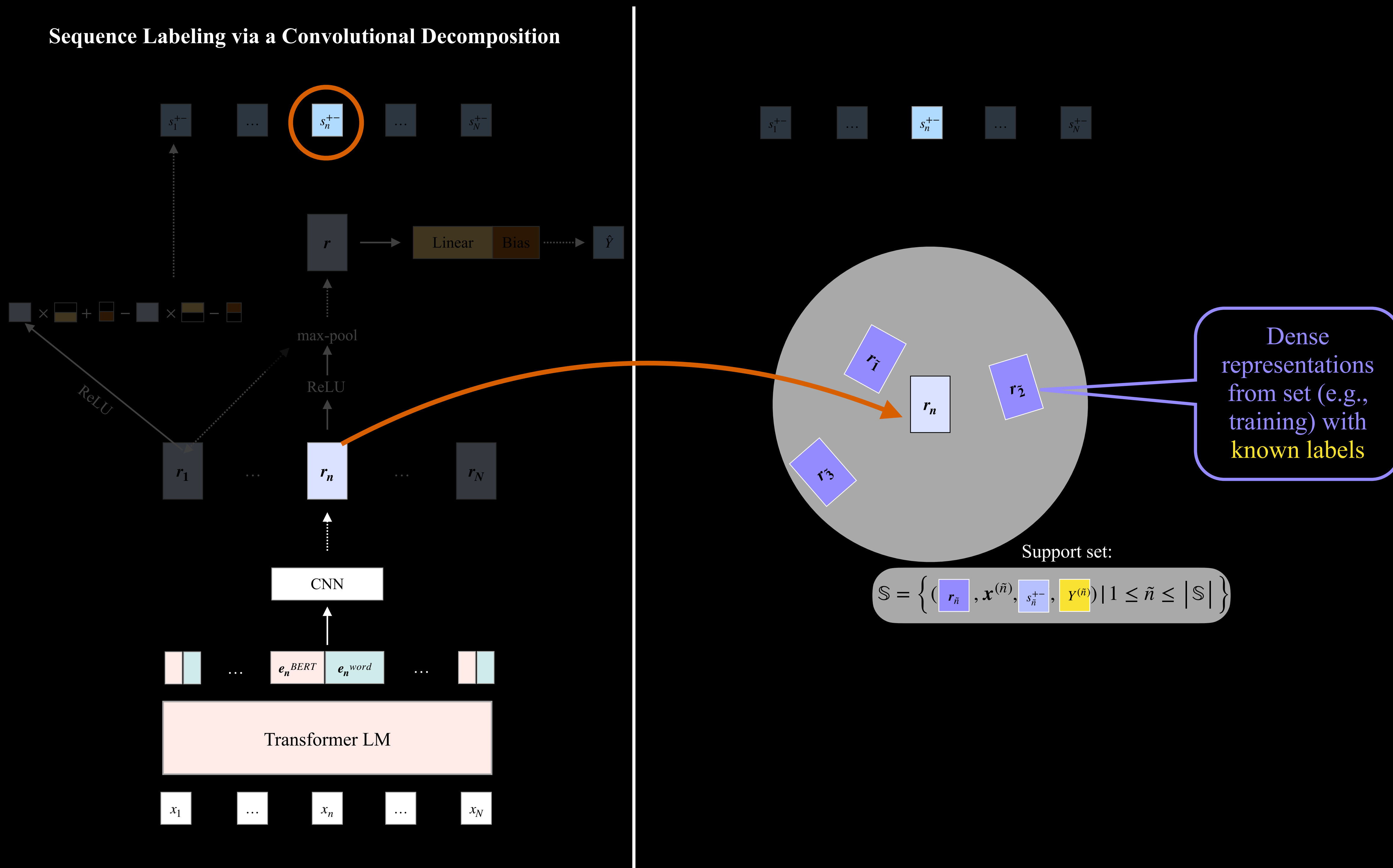
Allen Schmaltz

Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions

Sequence Labeling via a Convolutional Decomposition



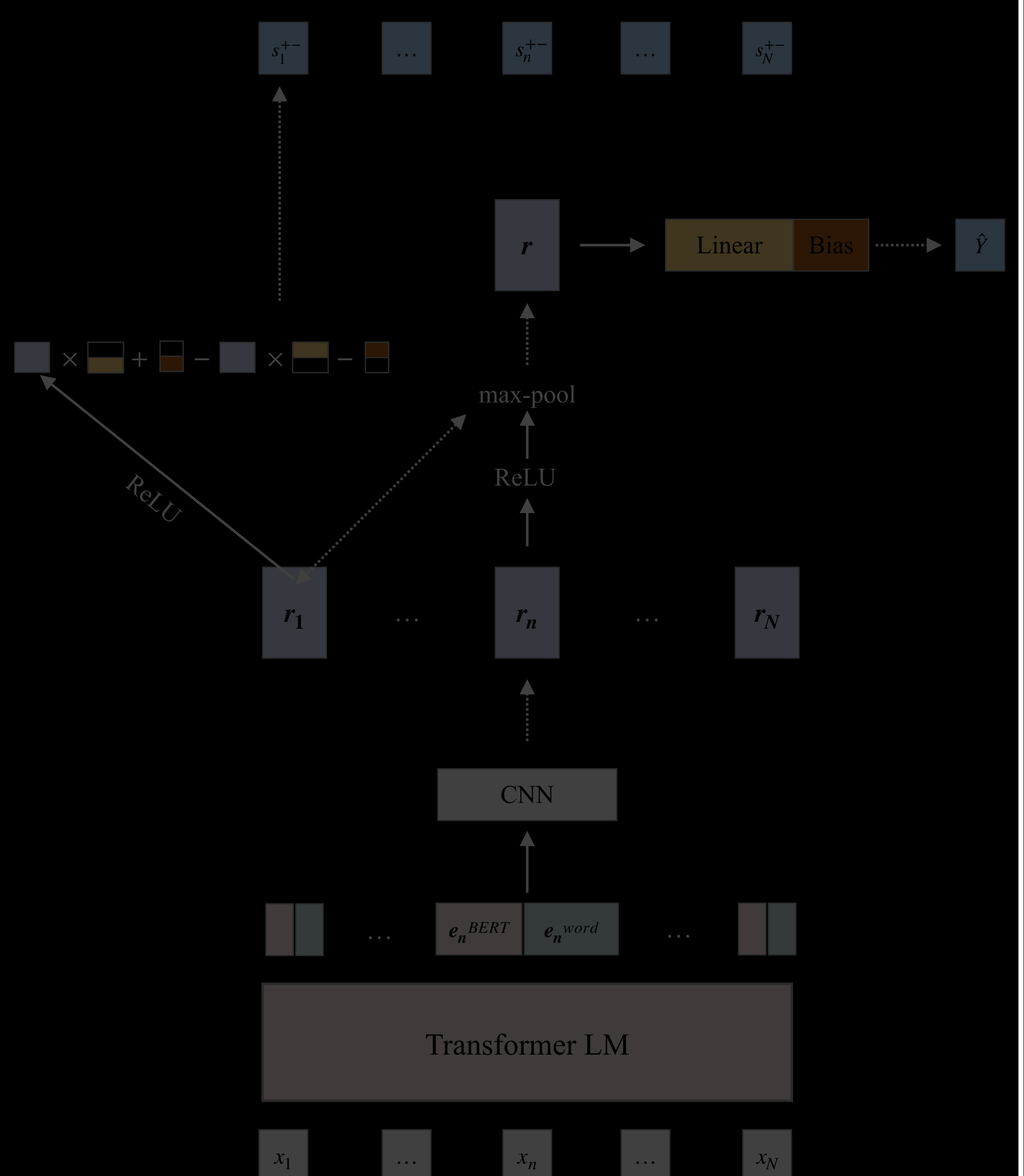
Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions



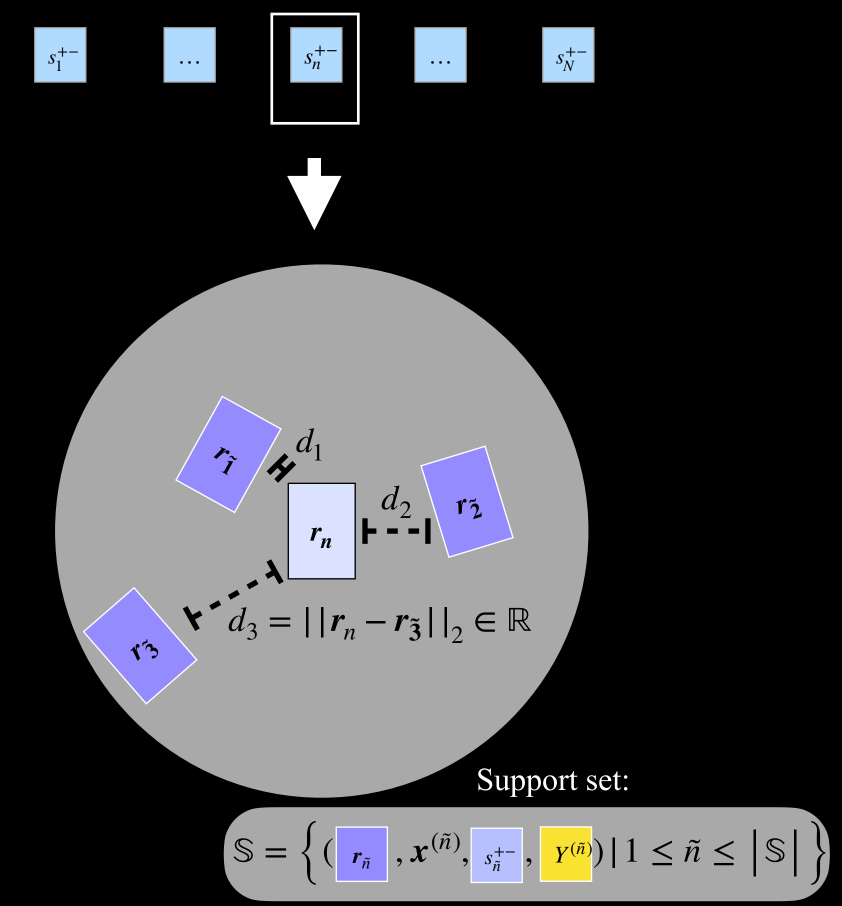
Allen Schmaltz

Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions

Sequence Labeling via a Convolutional Decomposition



K-NN Approximation



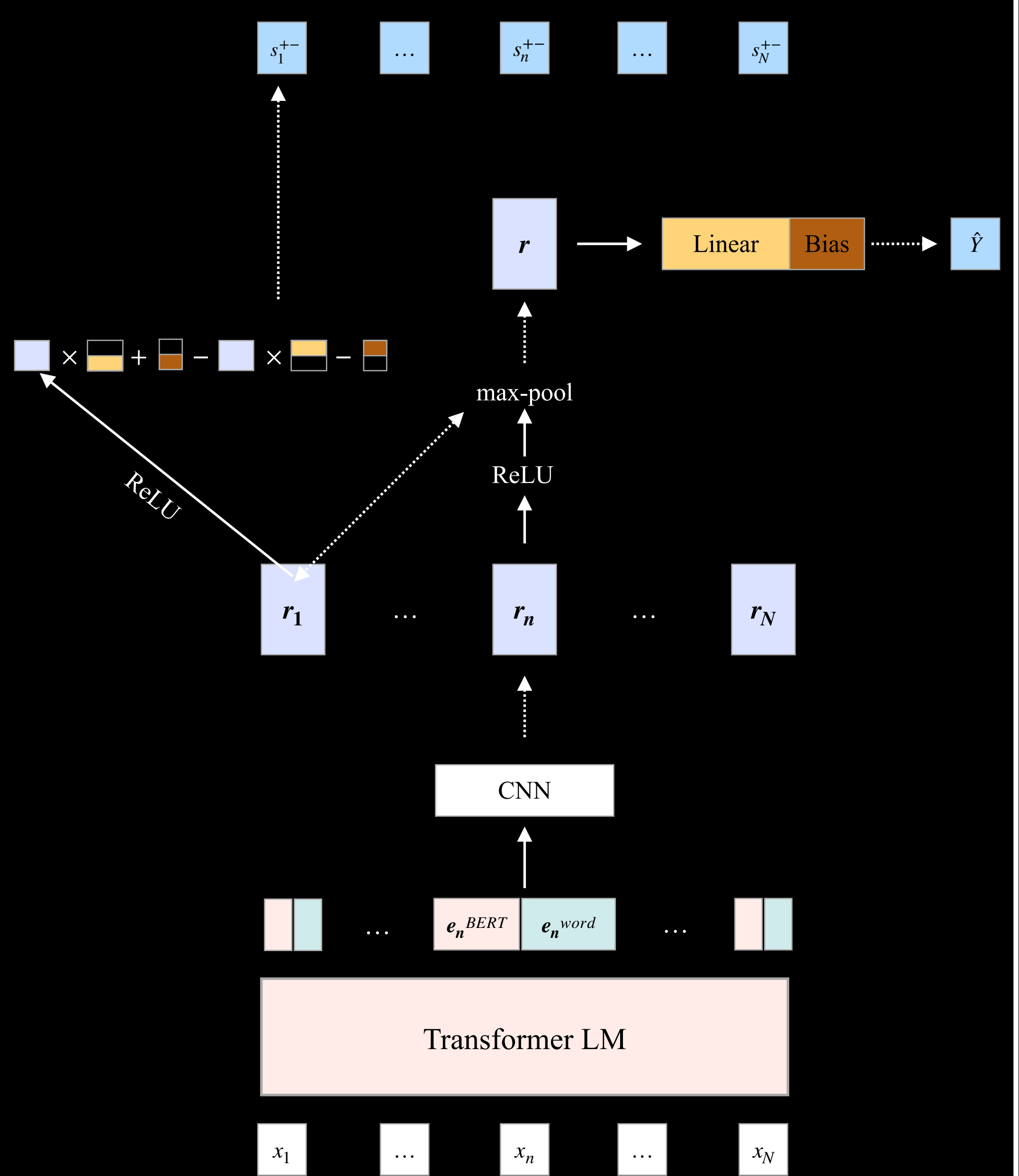
$$\begin{aligned}\boxed{s_n^{+-}} \approx \beta &+ w_1 \cdot \left(\tanh(\boxed{s_{\tilde{1}}^{+-}}) + \gamma \cdot \boxed{Y^{(\tilde{1})}}\right)\\ &+ w_2 \cdot \left(\tanh(\boxed{s_{\tilde{2}}^{+-}}) + \gamma \cdot \boxed{Y^{(\tilde{2})}}\right)\\ &+ w_3 \cdot \left(\tanh(\boxed{s_{\tilde{3}}^{+-}}) + \gamma \cdot \boxed{Y^{(\tilde{3})}}\right)\end{aligned}$$

$$w_k = \frac{\exp(-d_k/\tau)}{\sum_{k'=1}^{3} \exp(-d_{k'}/\tau)}$$

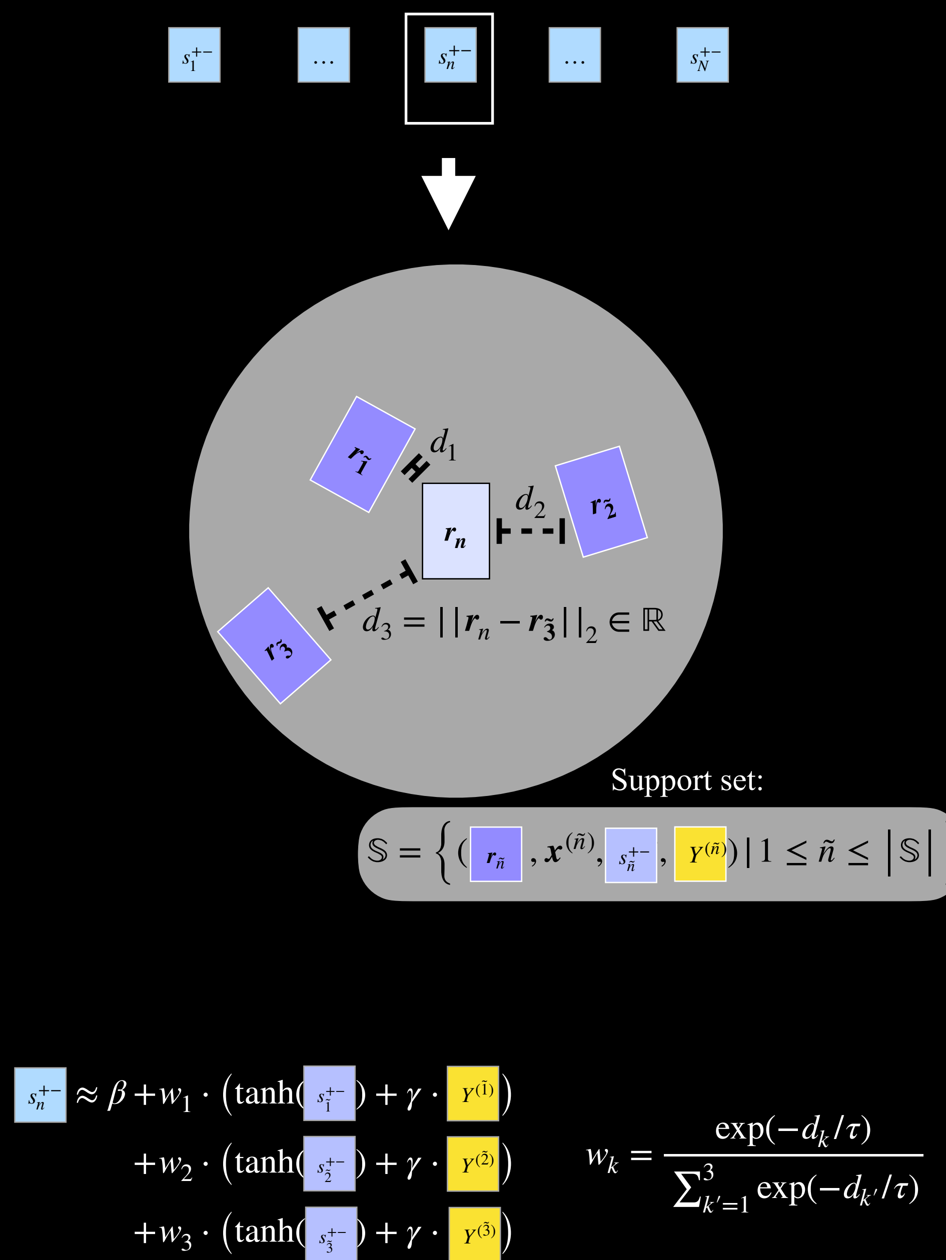
Allen Schmaltz

Horizontal (across the input) & *Vertical* (across the support set) Model Decompositions

Sequence Labeling via a Convolutional Decomposition



K-NN Approximation



Allen Schmaltz

Plan

- Task: Zero-shot sequence labeling
  - Running example: Grammatical errer [1] detection

Plan

- Task: Zero-shot sequence labeling
  - Running example: Grammatical error detection
- Decomposition across the input
- Decomposition across the support set
- Unique properties (added to the standard deep networks):
  - Analyze data at lower resolutions than available labels
  - Out-of-domain (OOD) detection / prediction reliability heuristics
  - Updatability

Task: Zero-Shot Binary Sequence Labeling

Corresponds to "feature detection" for document-level classification models

- Training: $\mathbb{D} = \{(\mathbf{x}_d, Y_d) \mid 1 \leq d \leq |\mathbb{D}|\}$
    - Document of N tokens/words: $\mathbf{x} = x_1, \ldots, x_n, \ldots, x_N$
    - Document-level label: $Y \in \{-1, 1\}$
- Inference:
    - Predict token-level labels: $\hat{\mathbf{y}} = \hat{y}_1, \ldots, \hat{y}_n, \ldots, \hat{y}_N$, where $\hat{y}_n \in \{-1, 1\}$

Task: ~~Zero-Shot~~ Supervised Binary Sequence Labeling

- Training: $\mathbb{D} = \{(\mathbf{x}_d, \cancel{Y_d}) \mid 1 \leq d \leq |\mathbb{D}|\}$ $\mathbf{y}_d$
  - Document of N tokens/words: $\mathbf{x} = x_1, \ldots, x_n, \ldots, x_N$
  - ~~Document-level label: $Y \in \{-1,1\}$~~

    Token-level labels: $\mathbf{y} = y_1, \ldots, y_n, \ldots, y_N$, **where** $y_n \in \{-1,1\}$
- Inference:
  - Predict token-level labels: $\hat{\mathbf{y}} = \hat{y}_1, \ldots, \hat{y}_n, \ldots, \hat{y}_N$, where $\hat{y}_n \in \{-1,1\}$

Inference task is unchanged. Training signal is different.

Allen Schmaltz

Task: Zero-Shot Binary Sequence Labeling

- Zero-Shot Grammatical Error Detection:

$y_1 = -1 \quad y_2 = 1 \quad y_3 = -1 \quad \ldots$

Sentence 1: `The runing example will be grammatical error detection, predicting whether or not each word has a grammatical error.`

$Y = 1$

Sentence 2: `See the paper for additional datasets and tasks.`

$Y = -1$

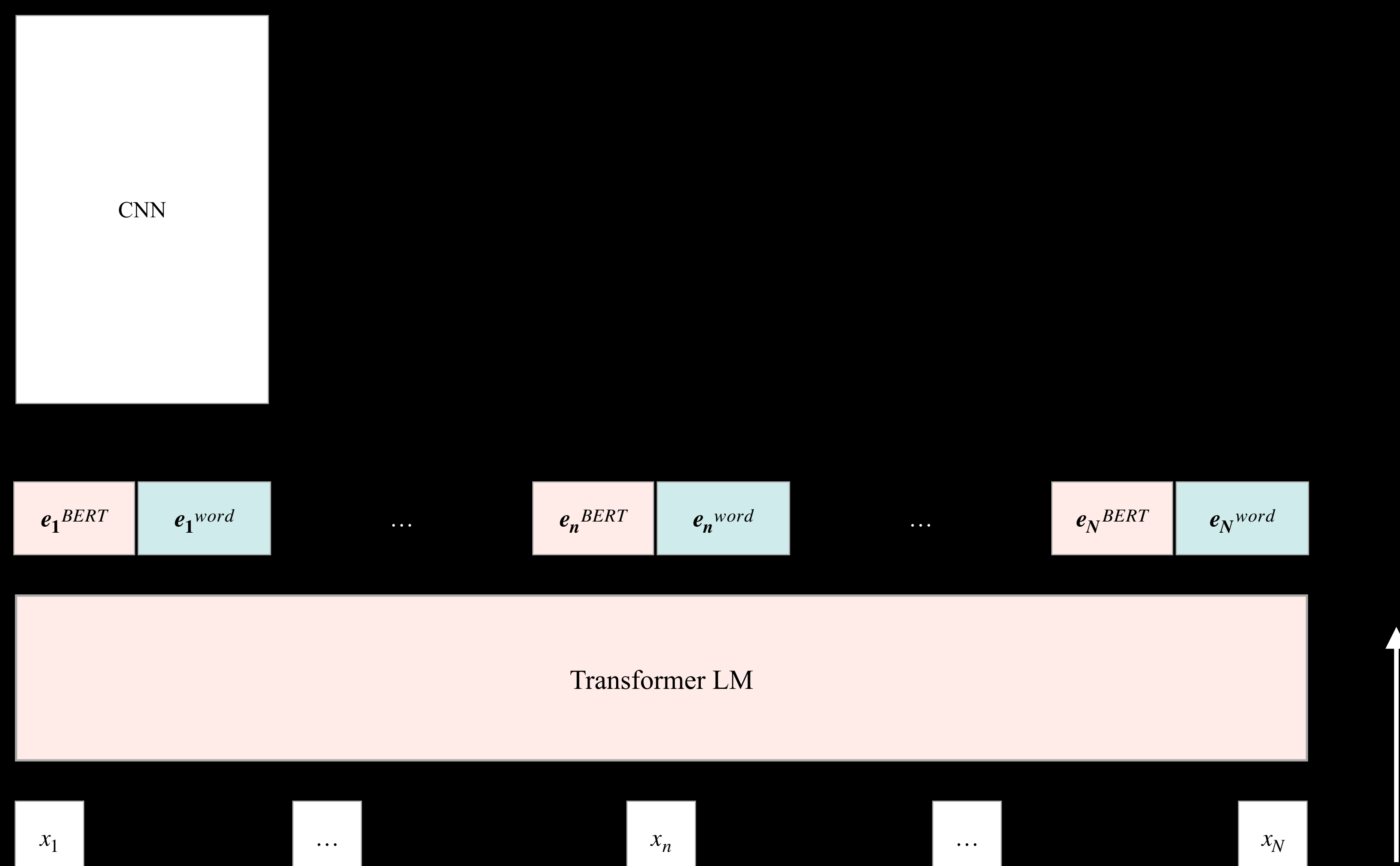
Intrinsic Challenges for Zero-Shot Labeling

- Multiple annotation schemes could be consistent with the document-level label
    - Need to think carefully about the inductive bias
    - Need some facility for adaptability to available priors
- Parameters of the network are not identifiable
    - Will instead aim for *instrospectable and updatable constraints against the observed data*

Mirrors the challenges with neural network interpretability, more generally

Horizontal (across the input) Model Decomposition

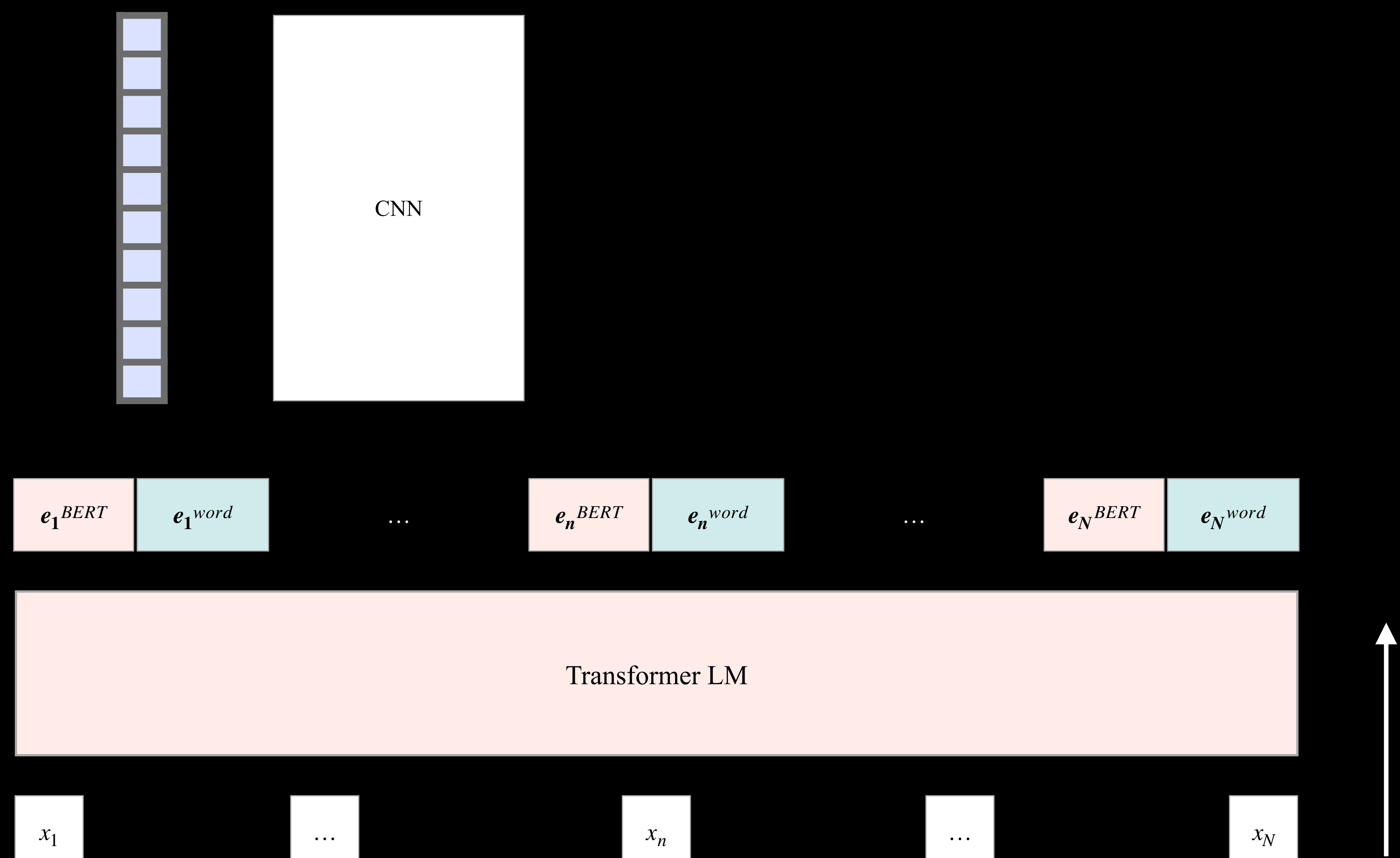
$M = 1000$ kernel-width 1 filters



Allen Schmaltz

Horizontal (across the input) Model Decomposition

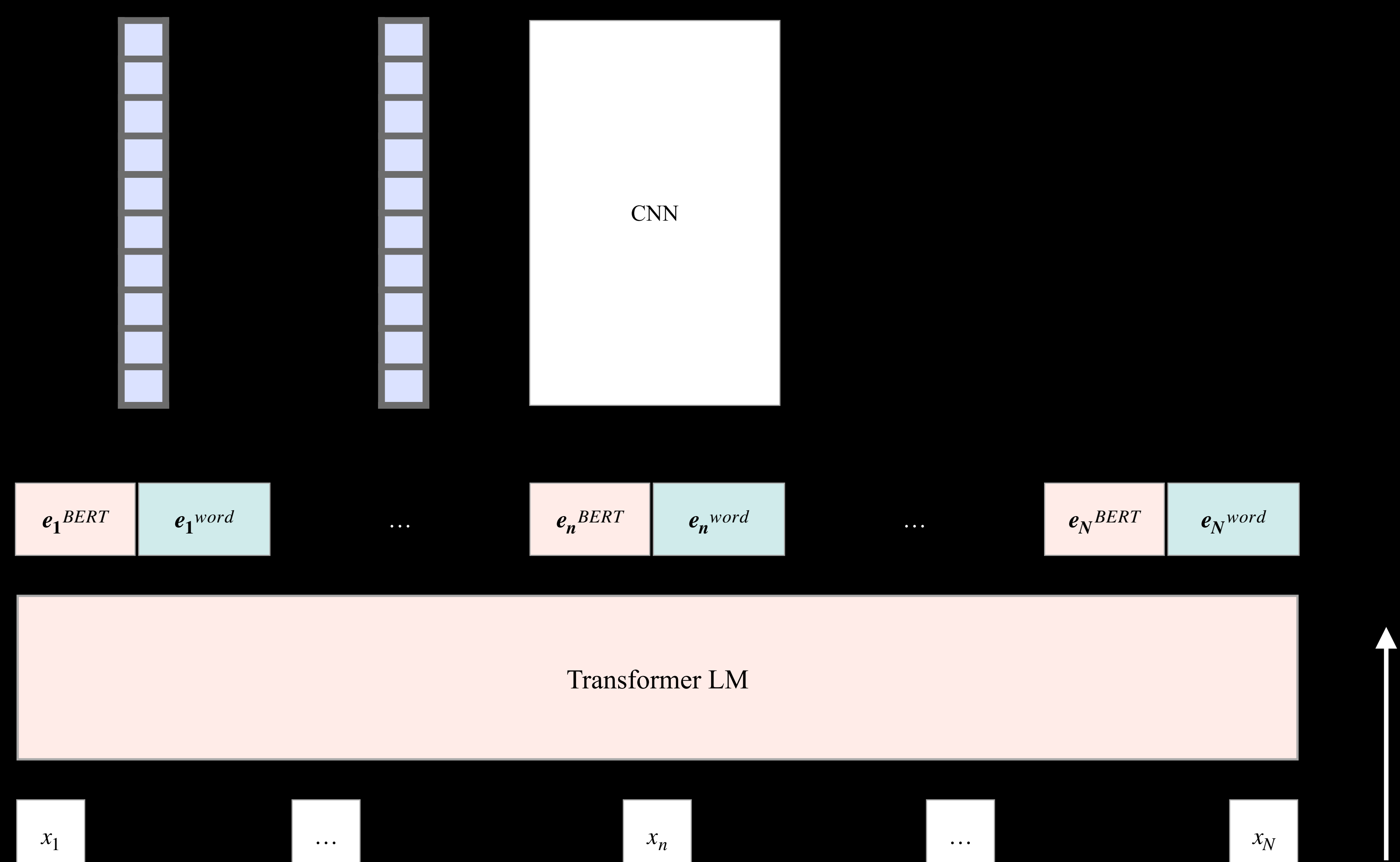
$M = 1000$ kernel-width 1 filters



Allen Schmaltz

Horizontal (across the input) Model Decomposition

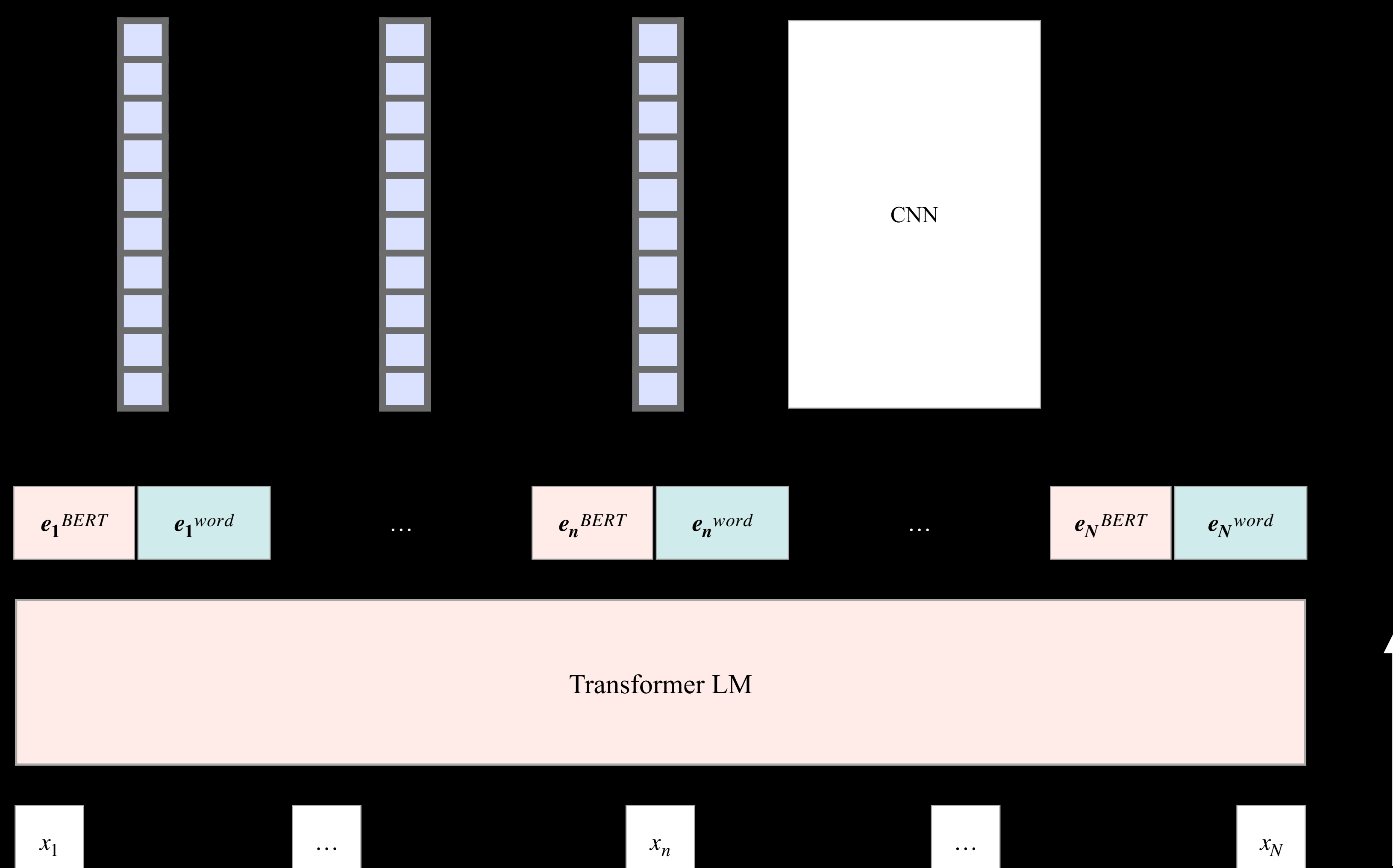
$M = 1000$ kernel-width 1 filters



Allen Schmaltz

Horizontal (across the input) Model Decomposition

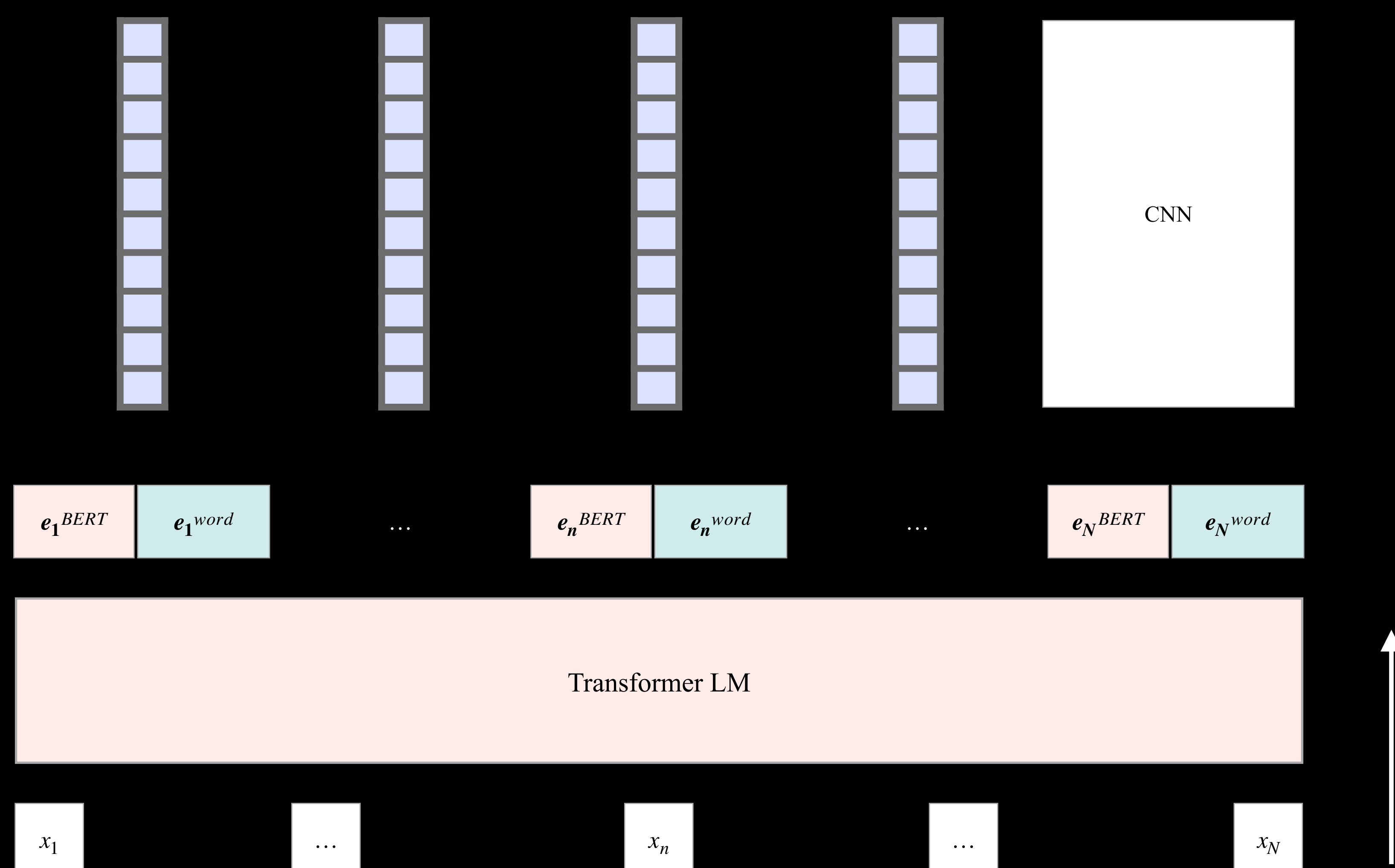
$M = 1000$ kernel-width 1 filters



Allen Schmaltz

Horizontal (across the input) Model Decomposition

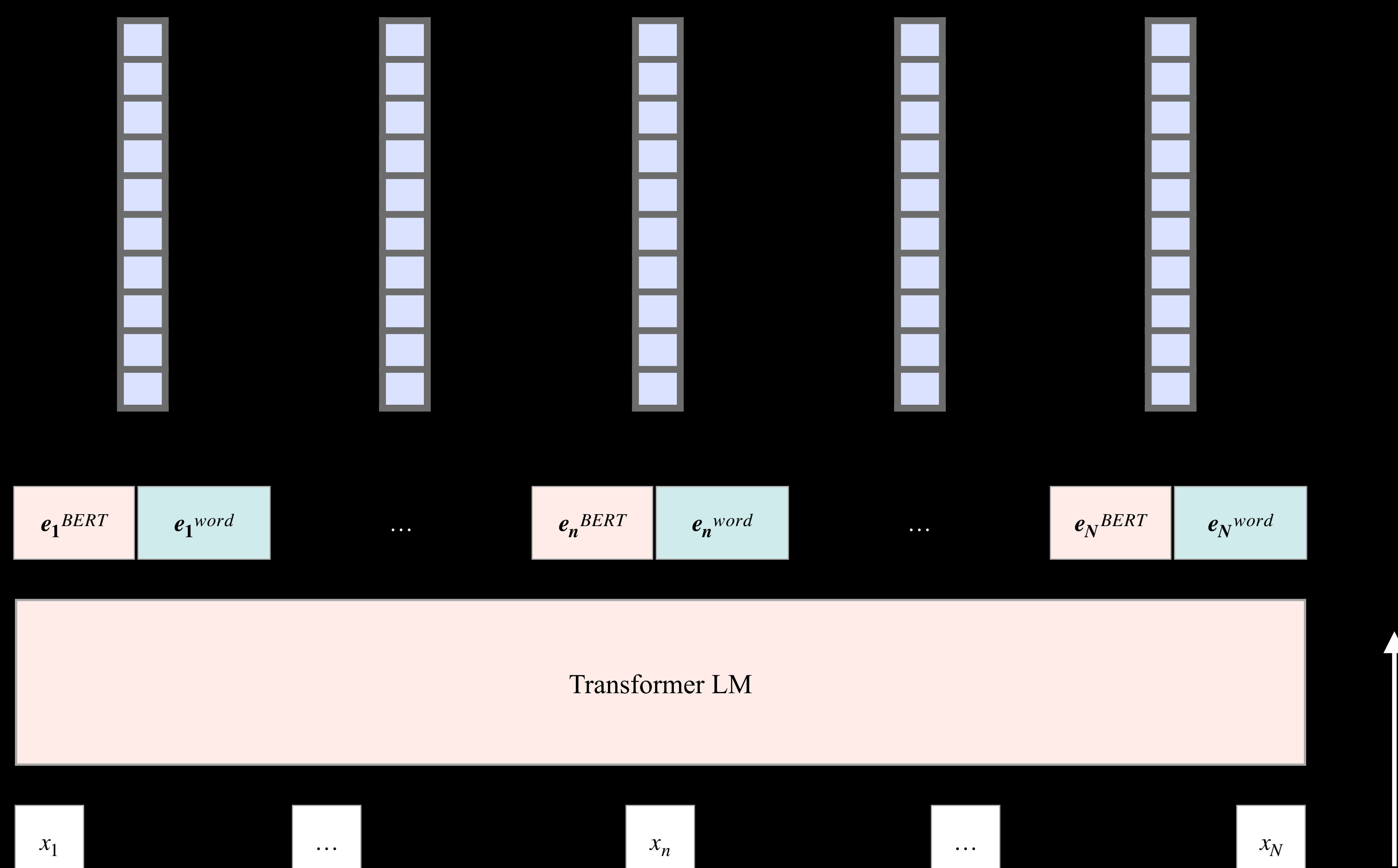
$M = 1000$ kernel-width 1 filters



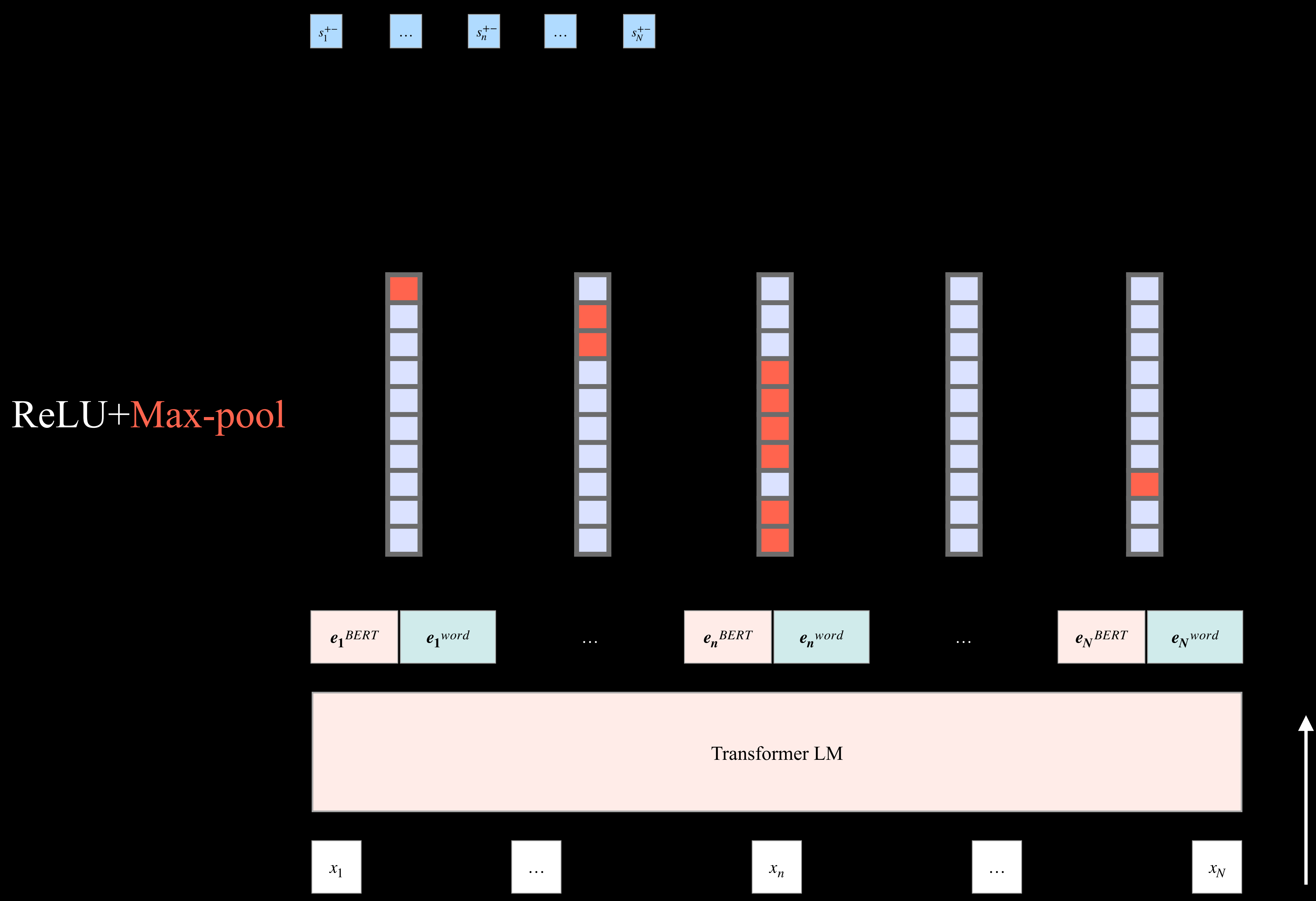
Allen Schmaltz

Horizontal (across the input) Model Decomposition

$M = 1000$ kernel-width 1 filters

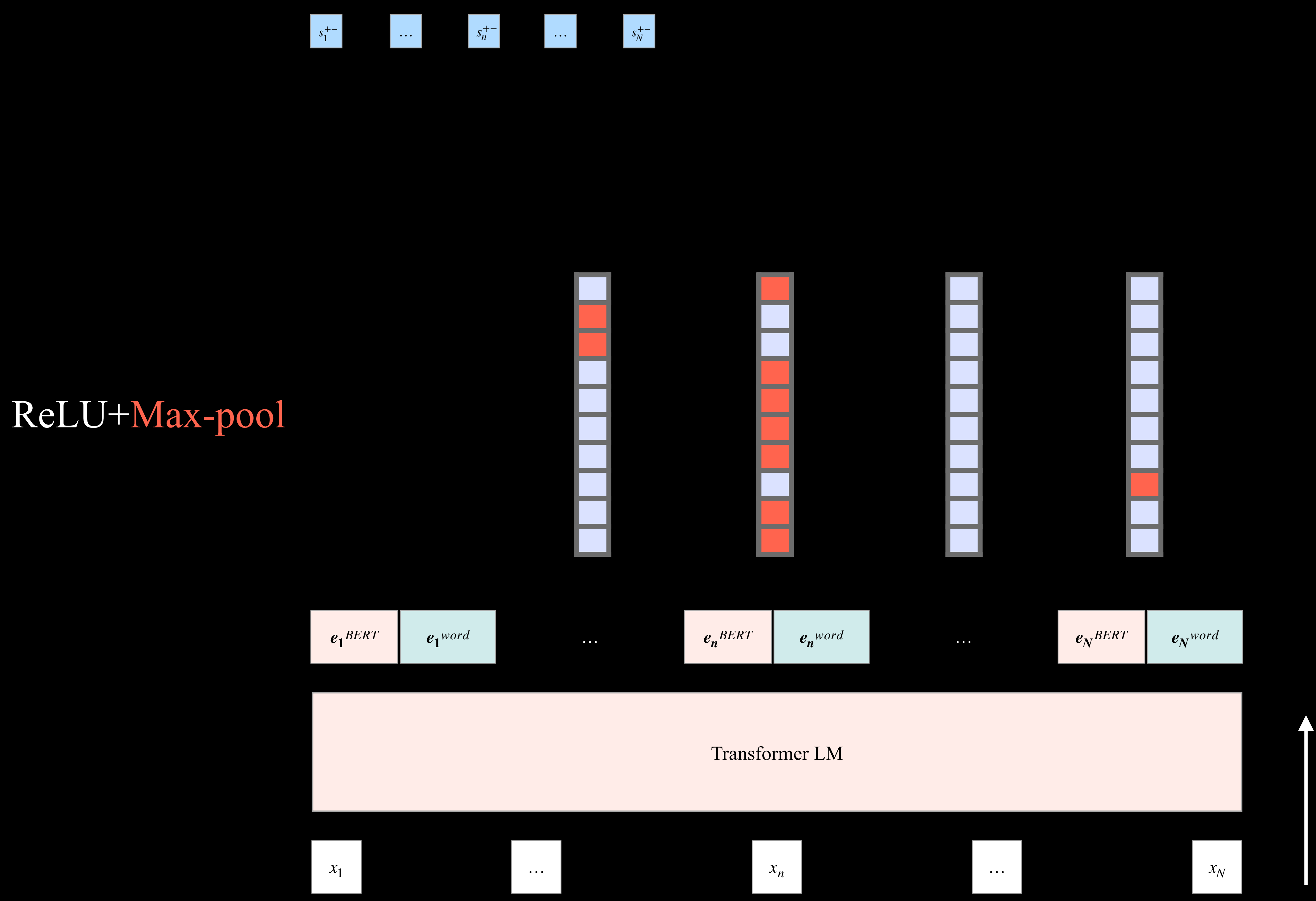


Horizontal (across the input) Model Decomposition



Allen Schmaltz

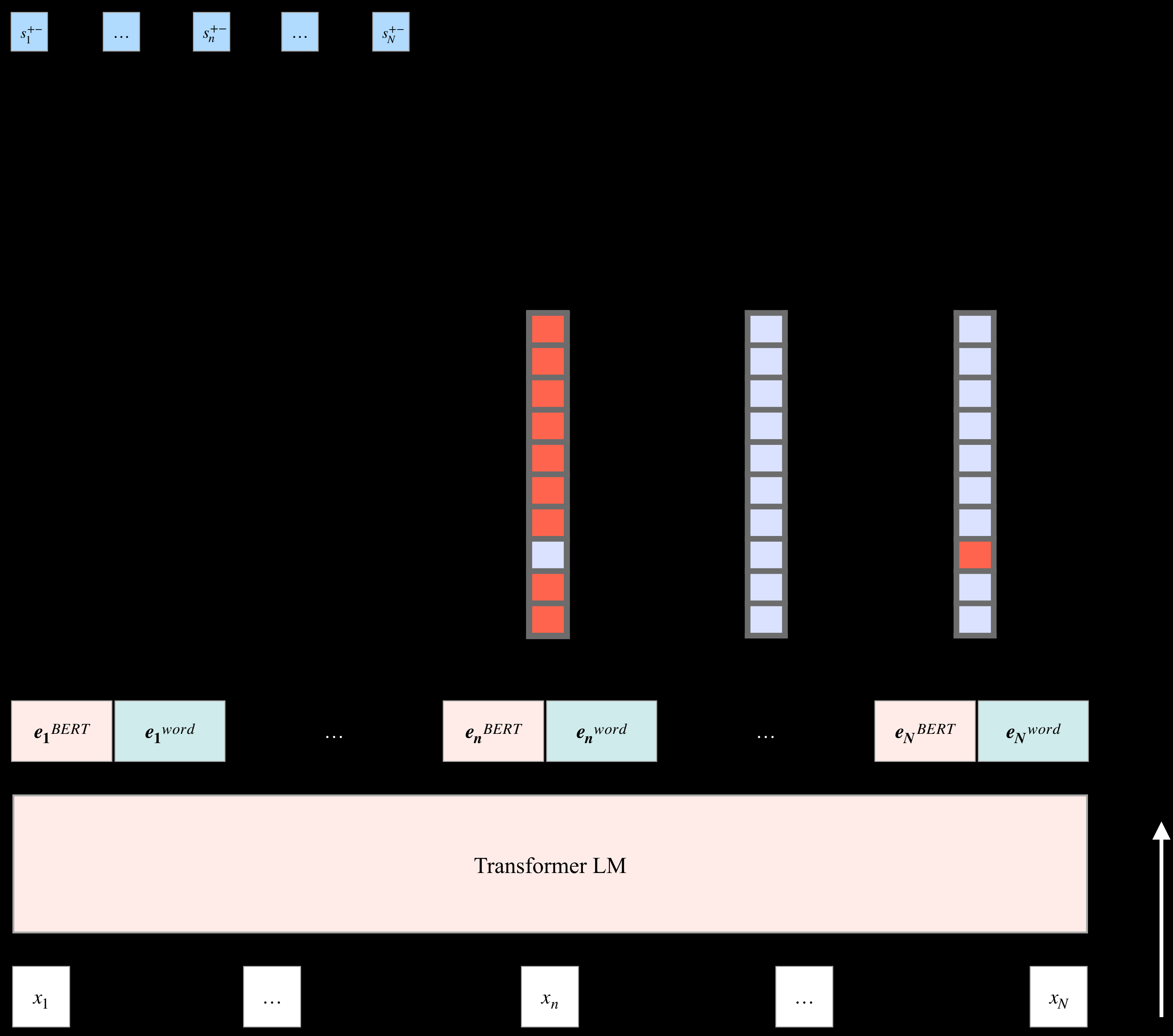
Horizontal (across the input) Model Decomposition



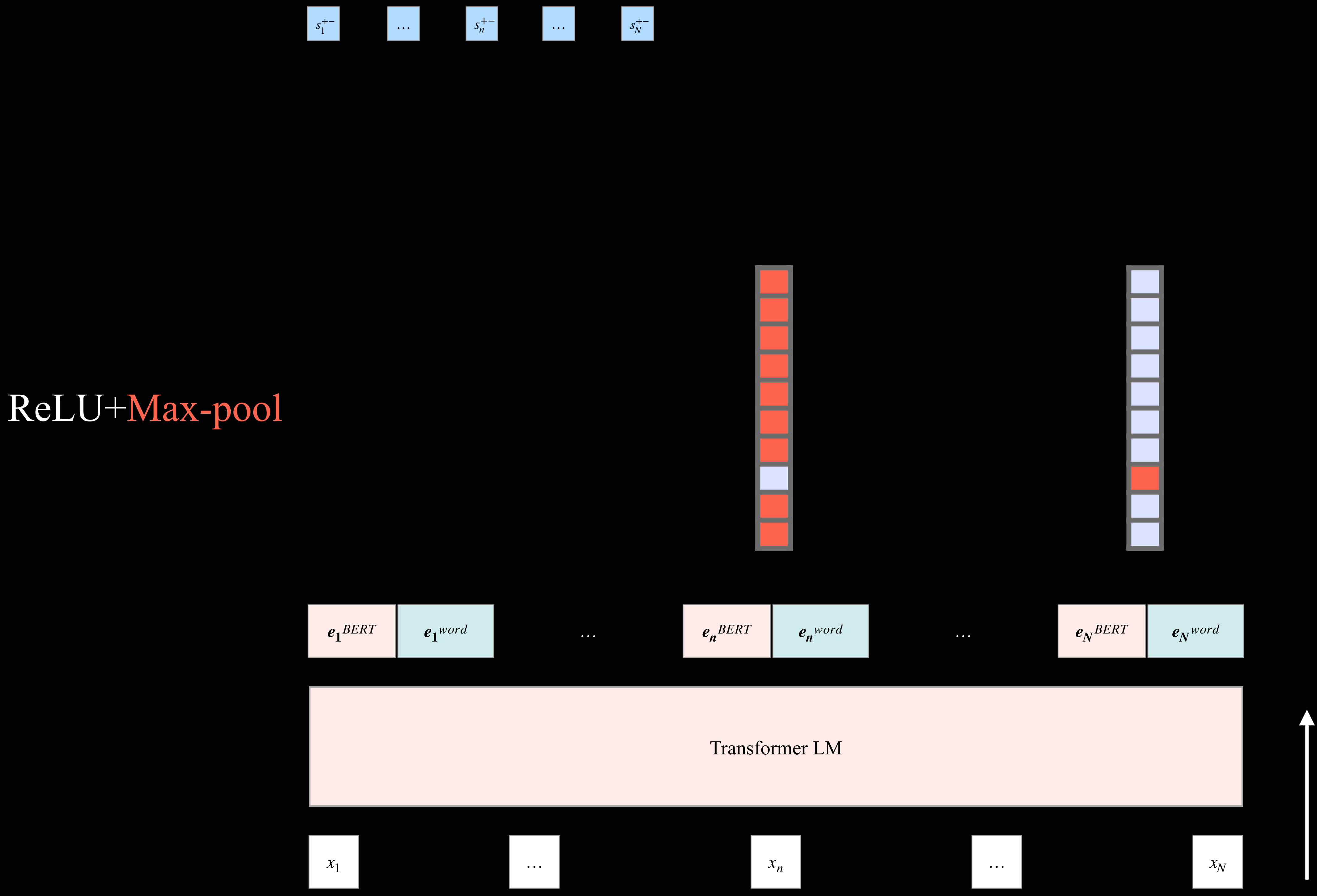
Allen Schmaltz

Horizontal (across the input) Model Decomposition

ReLU+Max-pool

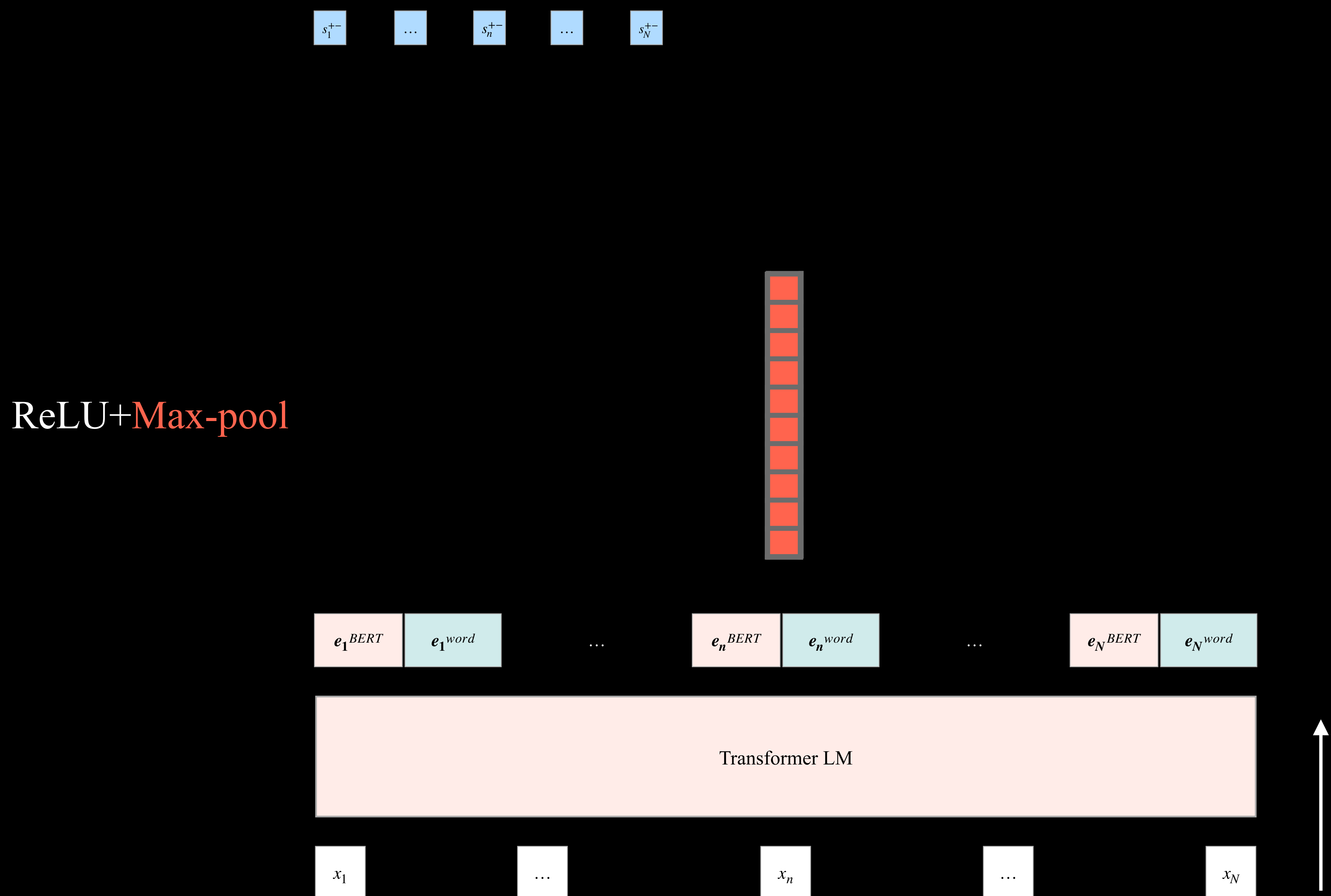


Horizontal (across the input) Model Decomposition

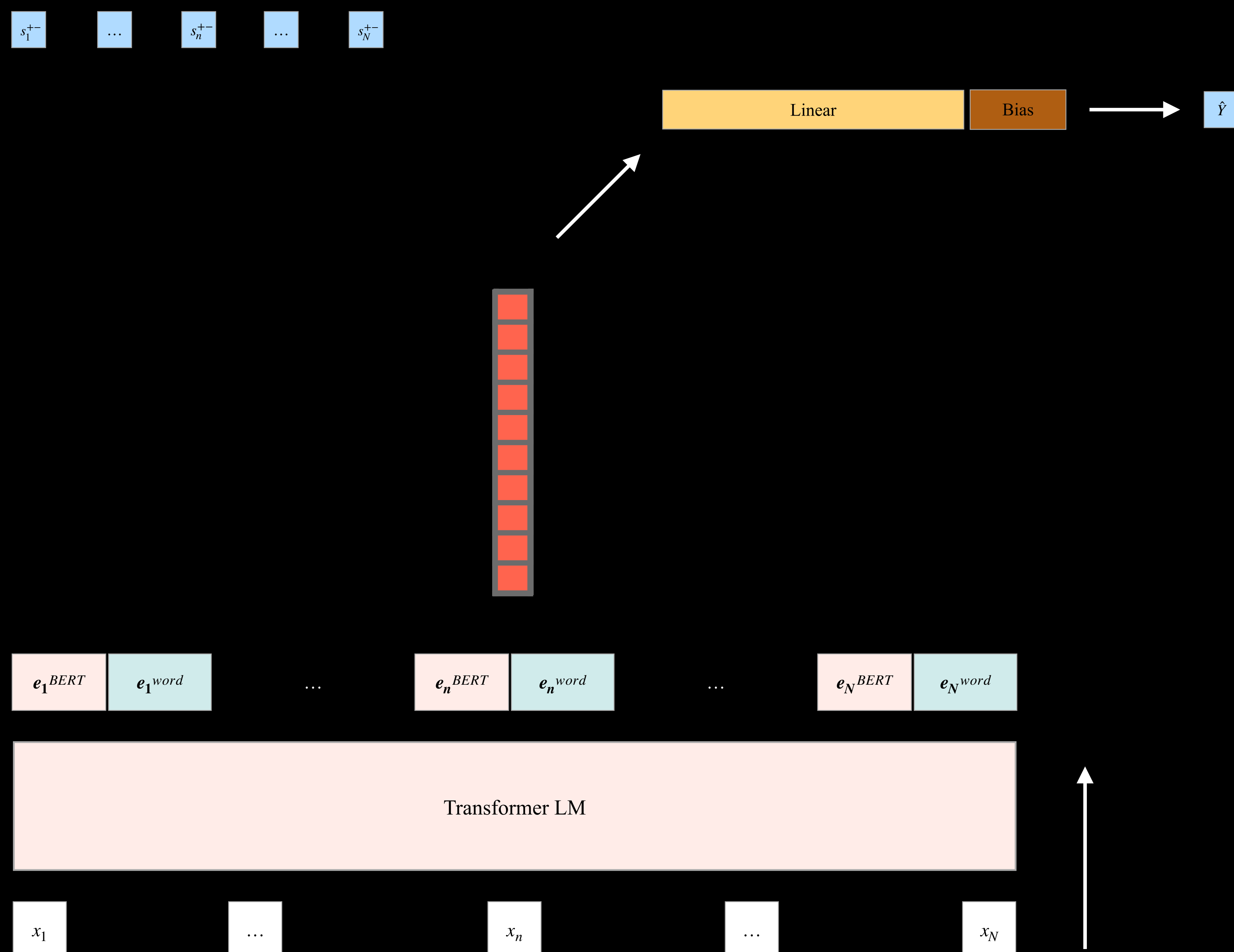


Allen Schmaltz

Horizontal (across the input) Model Decomposition

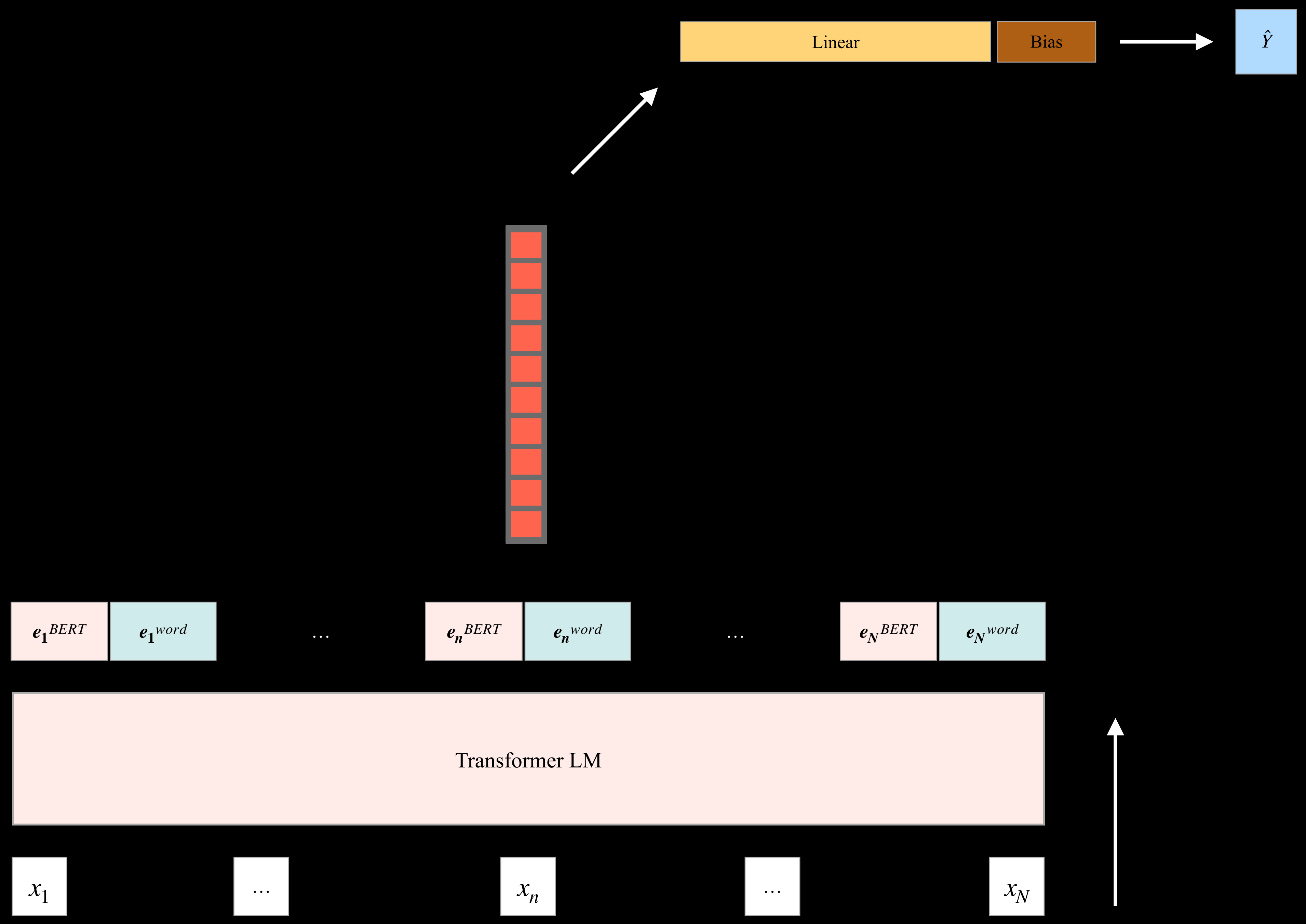


Horizontal (across the input) Model Decomposition



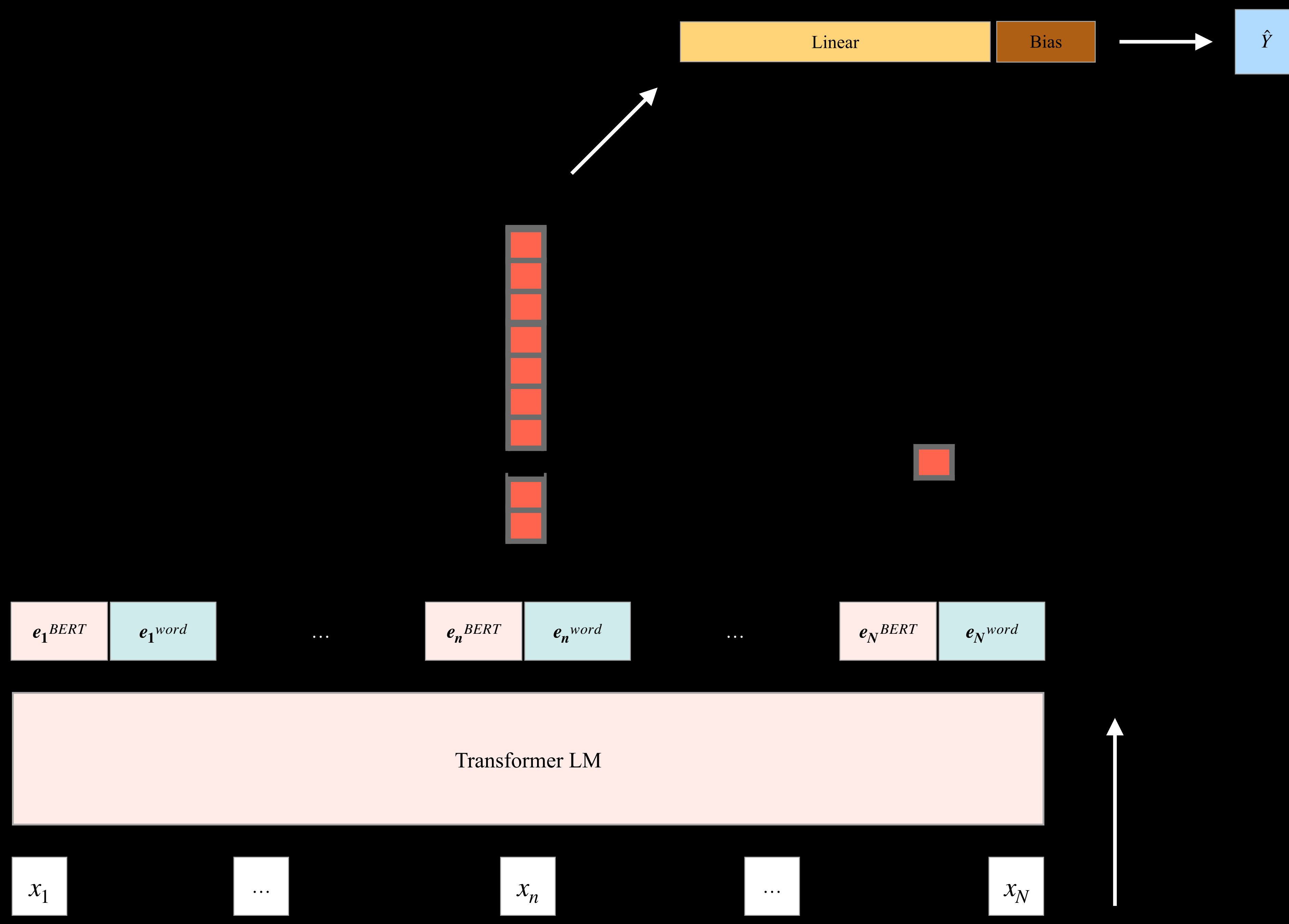
ReLU+Max-pool

Horizontal (across the input) Model Decomposition



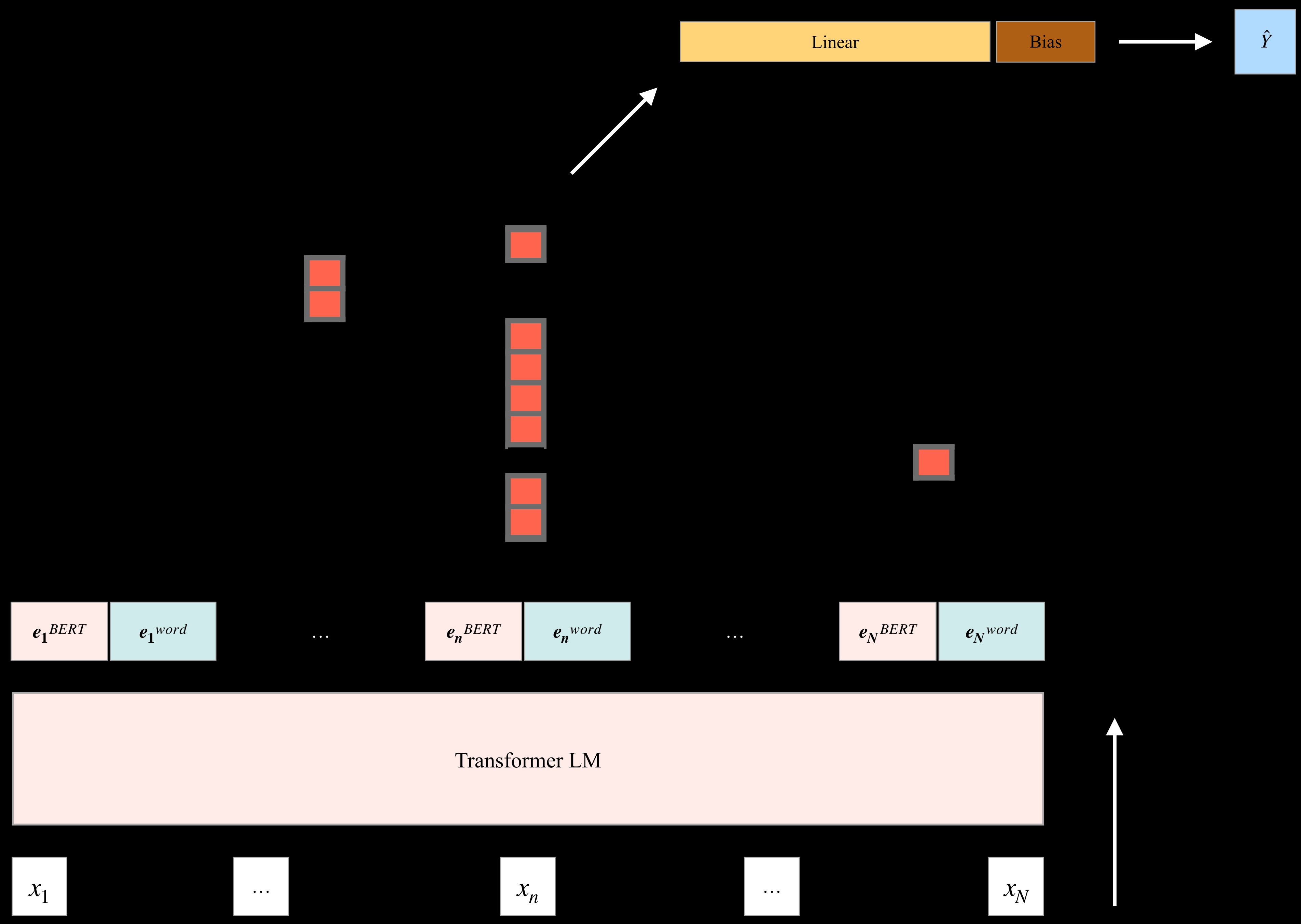
Allen Schmaltz

Horizontal (across the input) Model Decomposition



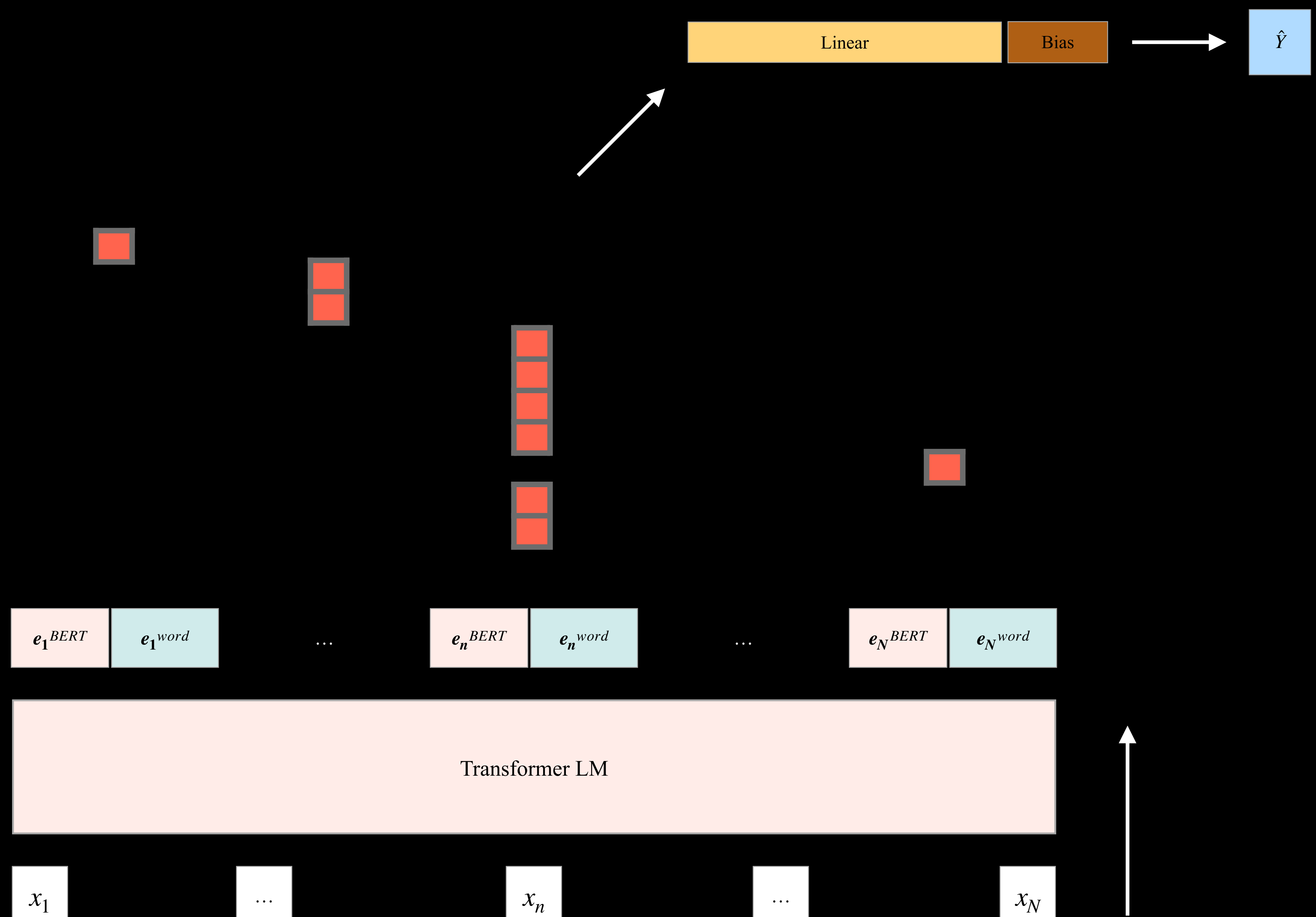
Allen Schmaltz

Horizontal (across the input) Model Decomposition



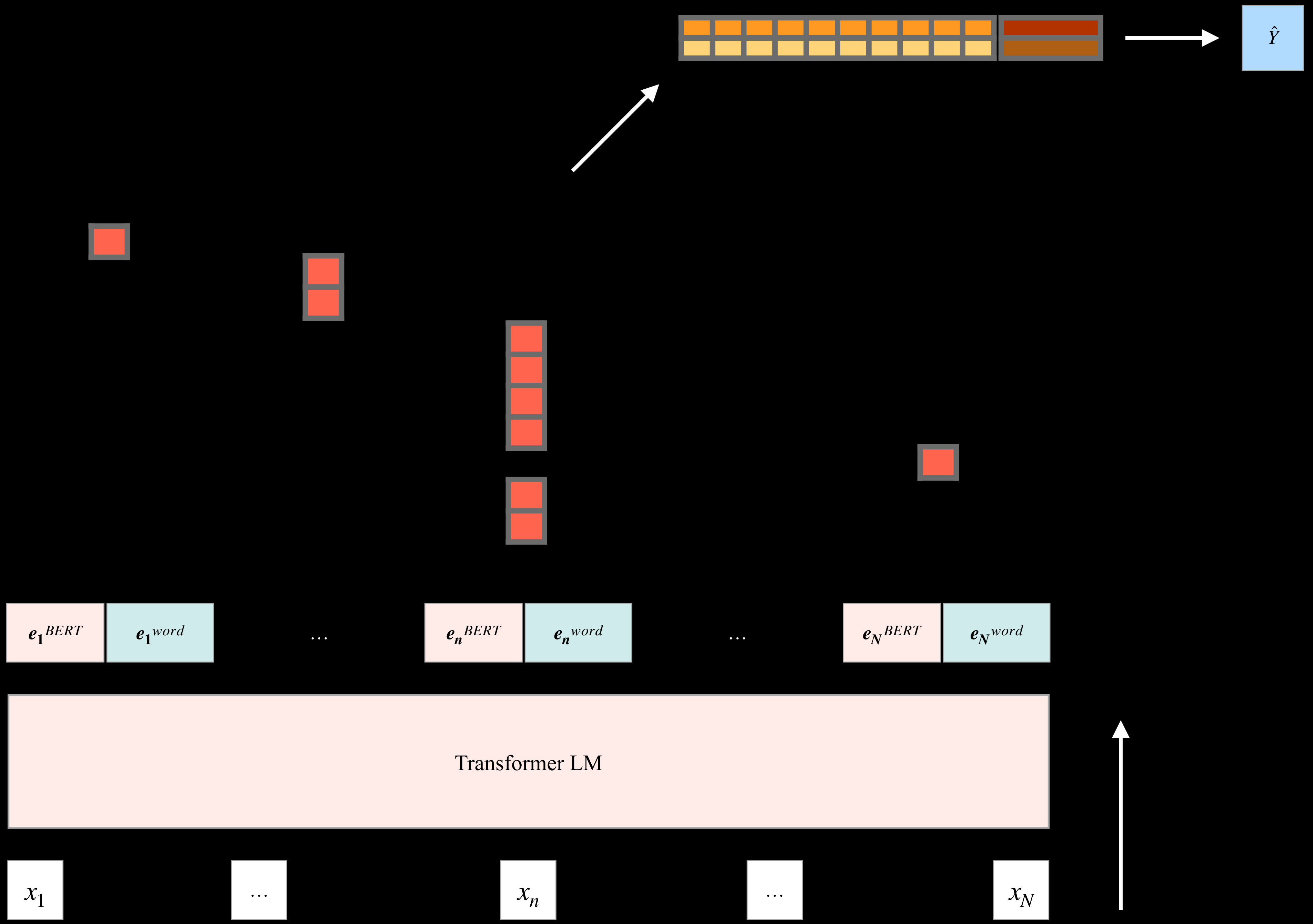
Allen Schmaltz

Horizontal (across the input) Model Decomposition



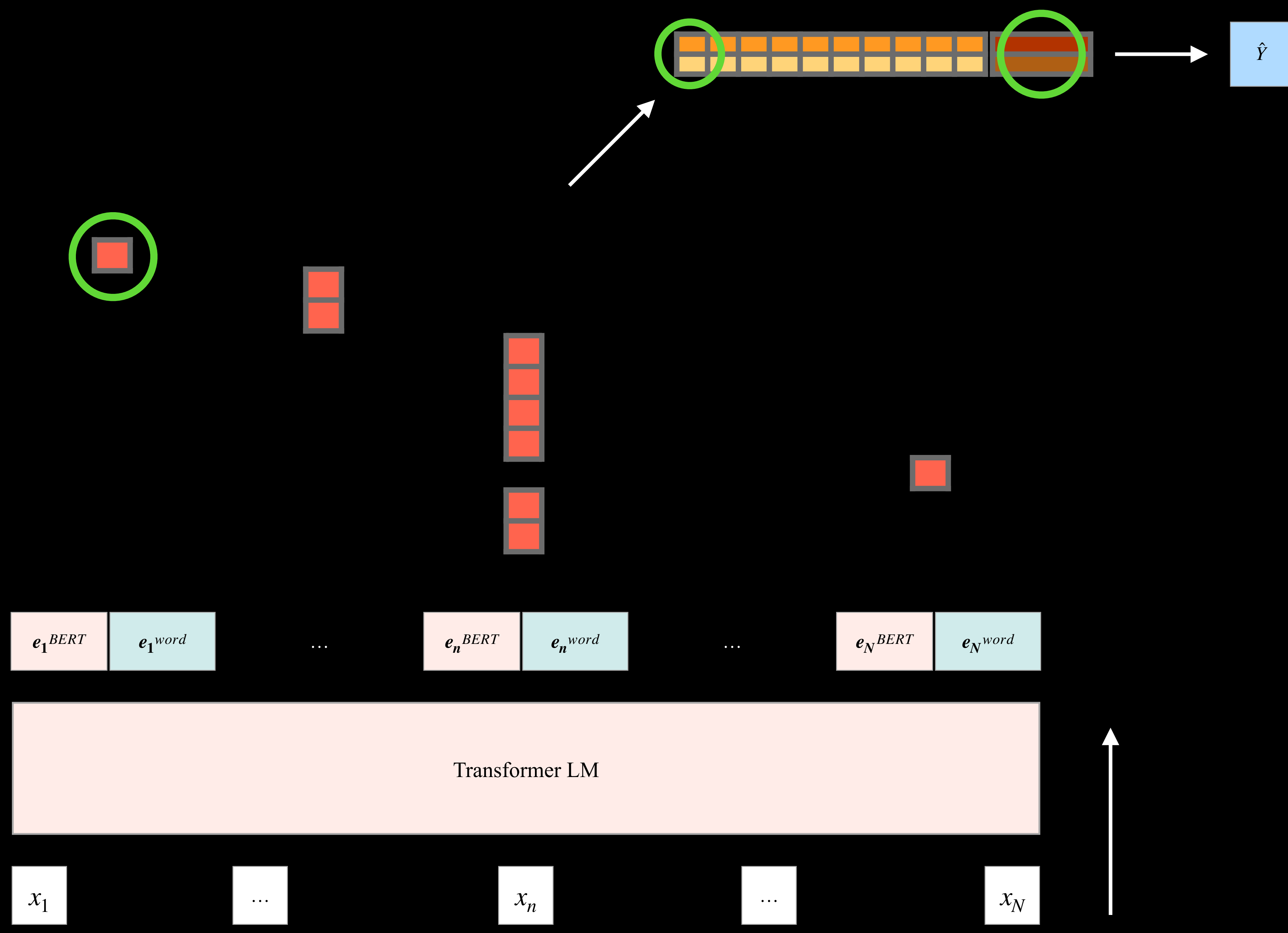
Allen Schmaltz

Horizontal (across the input) Model Decomposition



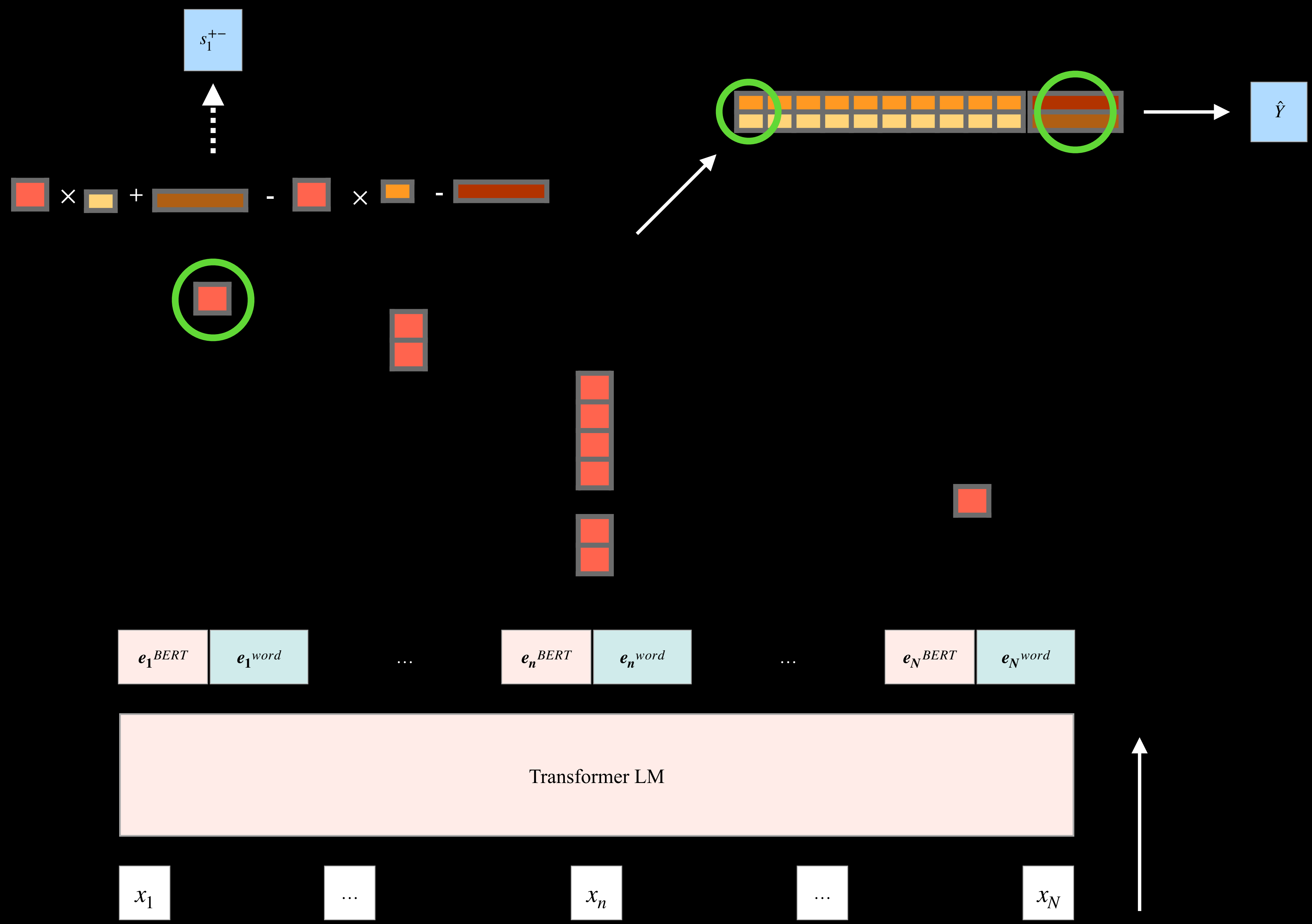
Allen Schmaltz

Horizontal (across the input) Model Decomposition



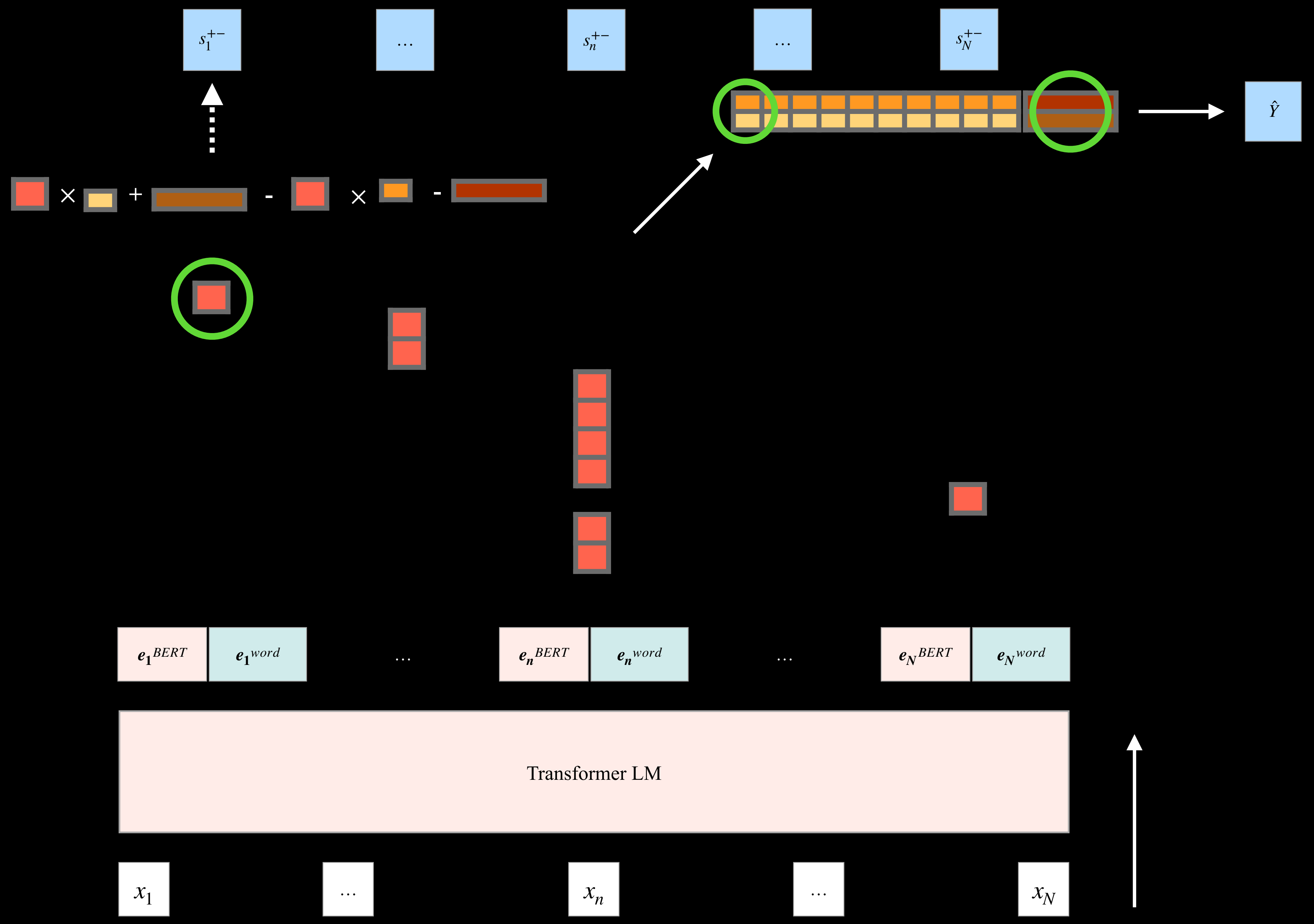
Allen Schmaltz

Horizontal (across the input) Model Decomposition



Allen Schmaltz

Horizontal (across the input) Model Decomposition



Allen Schmaltz

Training

Training

- Cross-entropy against document-level label, $Y' \in \{0,1\}$

Training

- Cross-entropy against document-level label, $Y' \in \{0,1\}$
- Min-max constraint to encourage sparsity
  - $\mathcal{L}_{min} = -\log(1 - \sigma(s^{+-}_{min}))$
    - $s^{+-}_{min} = \min(s^{+-}_1, \ldots, s^{+-}_n, \ldots, s^{+-}_N)$
  - $\mathcal{L}_{max} = -Y' \cdot \log \sigma(s^{+-}_{max}) - (1 - Y') \cdot \log(1 - \sigma(s^{+-}_{max}))$
    - $s^{+-}_{max} = \max(s^{+-}_1, \ldots, s^{+-}_n, \ldots, s^{+-}_N)$

Stronger priors (w.r.t. label distribution)

Training

- Cross-entropy against document-level label, $Y' \in \{0,1\}$
- Min-max constraint to encourage sparsity
  - $\mathscr{L}_{min} = -\log(1-\sigma(s^{+-}_{min}))$
    - $s^{+-}_{min} = \min(s^{+-}_1, \ldots, s^{+-}_n, \ldots, s^{+-}_N)$
  - $\mathscr{L}_{max} = -Y' \cdot \log \sigma(s^{+-}_{max}) - (1-Y') \cdot \log(1-\sigma(s^{+-}_{max}))$
    - $s^{+-}_{max} = \max(s^{+-}_1, \ldots, s^{+-}_n, \ldots, s^{+-}_N)$

Difference practically useful … but also concerning ($\hat{Y}$ similar)

Stronger priors (w.r.t. label distribution)

Allen Schmaltz

Training

- Cross-entropy against document-level label, $Y' \in \{0,1\}$
- Min-max constraint to encourage sparsity
  - $\mathcal{L}_{min} = -\log(1 - \sigma(s^{+-}_{min}))$
    - $s^{+-}_{min} = \min(s^{+-}_1, \ldots, s^{+-}_n, \ldots, s^{+-}_N)$
  - $\mathcal{L}_{max} = -Y' \cdot \log \sigma(s^{+-}_{max}) - (1 - Y') \cdot \log(1 - \sigma(s^{+-}_{max}))$
    - $s^{+-}_{max} = \max(s^{+-}_1, \ldots, s^{+-}_n, \ldots, s^{+-}_N)$

Difference practically useful … but also concerning ($\hat{Y}$ similar)

- Fully-supervised (token-level)
  - $\mathcal{L}_n = -y'_n \cdot \log \sigma(s^{+-}_n) - (1 - y'_n) \cdot \log(1 - \sigma(s^{+-}_n))$

Stronger priors (w.r.t. label distribution) ⟶

Empirical Results

FCE zero-shot sequence labeling test set results (Appendix: Table E.1)
†Results from previous works

Empirical Results

	Sentence-level	Token-level			
Model	F_1	P	R	F_1	$F_{0.5}$
RANDOM	58.30	15.30	50.07	23.44	17.79
MAJORITYCLASS	80.88	15.20	100	26.39	18.31
LIME (ROBERTA$_{BASE}$ TRANSFORMER) †	84.51	19.06	34.70	24.60	20.95

FCE zero-shot sequence labeling test set results (Appendix: Table E.1)
†Results from previous works

Empirical Results

	Sentence-level	Token-level			
Model	F_1	P	R	F_1	$F_{0.5}$
RANDOM	58.30	15.30	50.07	23.44	17.79
MAJORITYCLASS	80.88	15.20	100	26.39	18.31
LIME (ROBERTA$_{BASE}$ TRANSFORMER) †	84.51	19.06	34.70	24.60	20.95
LSTM+SOFTATTENTION †	85.14	28.04	29.91	28.27	28.40
TRANSFORMER (ROBERTA$_{BASE}$) + WEIGHTEDSOFTATTENTION †	85.62	20.76	85.36	33.31	24.46

FCE zero-shot sequence labeling test set results (Appendix: Table E.1)
†Results from previous works

Allen Schmaltz

Empirical Results

Model	Sentence-level	Token-level			
	F_1	P	R	F_1	$F_{0.5}$
RANDOM	58.30	15.30	50.07	23.44	17.79
MAJORITYCLASS	80.88	15.20	100	26.39	18.31
LIME (ROBERTA$_{BASE}$ TRANSFORMER) †	84.51	19.06	34.70	24.60	20.95
LSTM+SOFTATTENTION †	85.14	28.04	29.91	28.27	28.40
TRANSFORMER (ROBERTA$_{BASE}$) + WEIGHTEDSOFTATTENTION †	85.62	20.76	85.36	33.31	24.46
TRANSFORMER (BERT$_{BASE}$) + CNNDECOMPOSITION (M=2)	86.22	57.91	19.33	28.99	41.39

Can only label max 2 tokens

FCE zero-shot sequence labeling test set results (Appendix: Table E.1)
†Results from previous works

Empirical Results

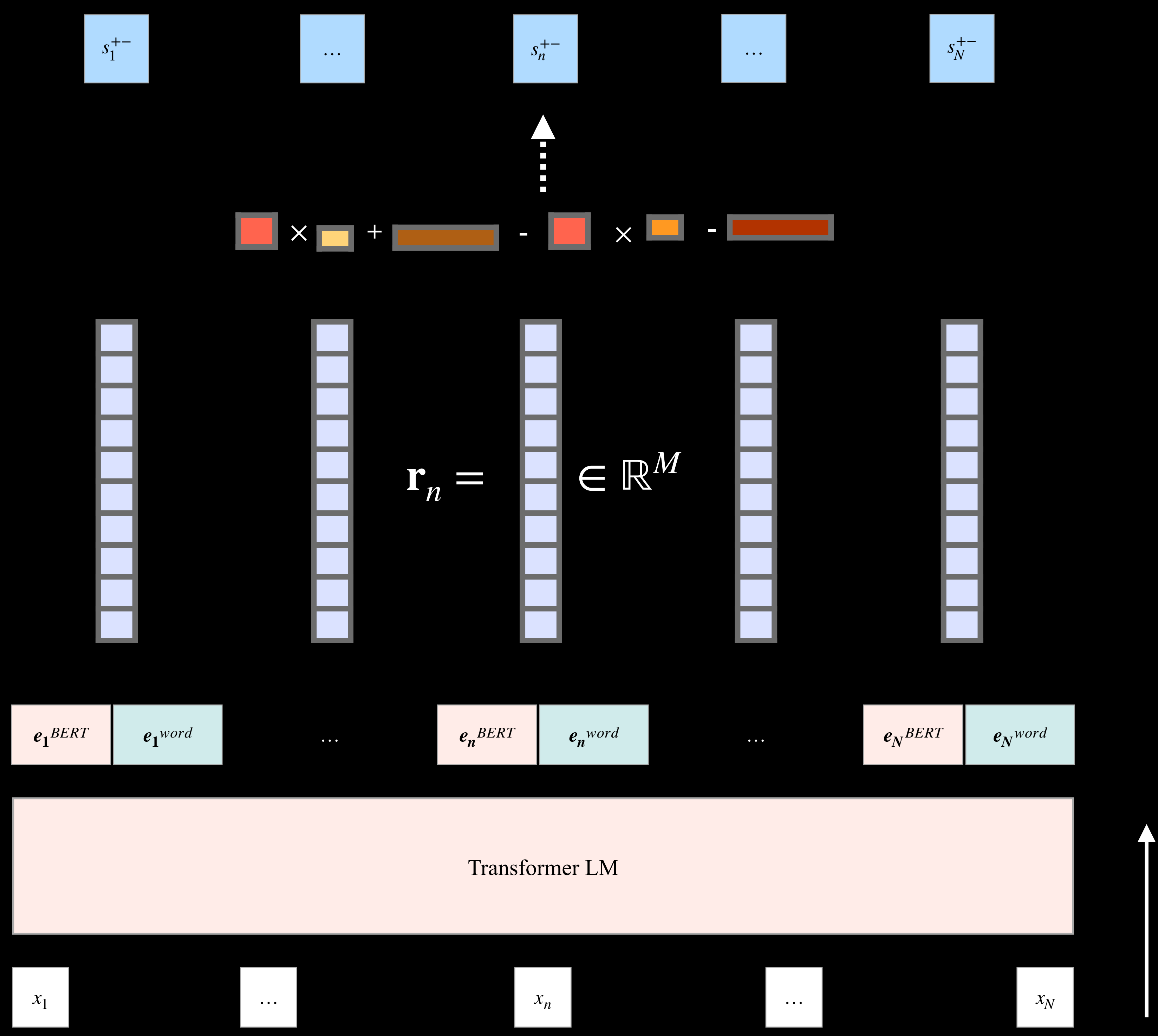
Model	Sentence-level	Token-level			
	F_1	P	R	F_1	$F_{0.5}$
RANDOM	58.30	15.30	50.07	23.44	17.79
MAJORITYCLASS	80.88	15.20	100	26.39	18.31
LIME (ROBERTA$_{BASE}$ TRANSFORMER) †	84.51	19.06	34.70	24.60	20.95
LSTM+SOFTATTENTION †	85.14	28.04	29.91	28.27	28.40
TRANSFORMER (ROBERTA$_{BASE}$) + WEIGHTEDSOFTATTENTION †	85.62	20.76	85.36	33.31	24.46
TRANSFORMER (BERT$_{BASE}$) + CNNDECOMPOSITION (M=2)	86.22	57.91	19.33	28.99	41.39
TRANSFORMER (BERT$_{BASE}$) + CNNDECOMPOSITION	86.29	53.17	35.37	42.48	48.31

Can only label max 2 tokens

FCE zero-shot sequence labeling test set results (Appendix: Table E.1)

†Results from previous works

Vertical (across the support set) Model Decomposition via Dense Matching

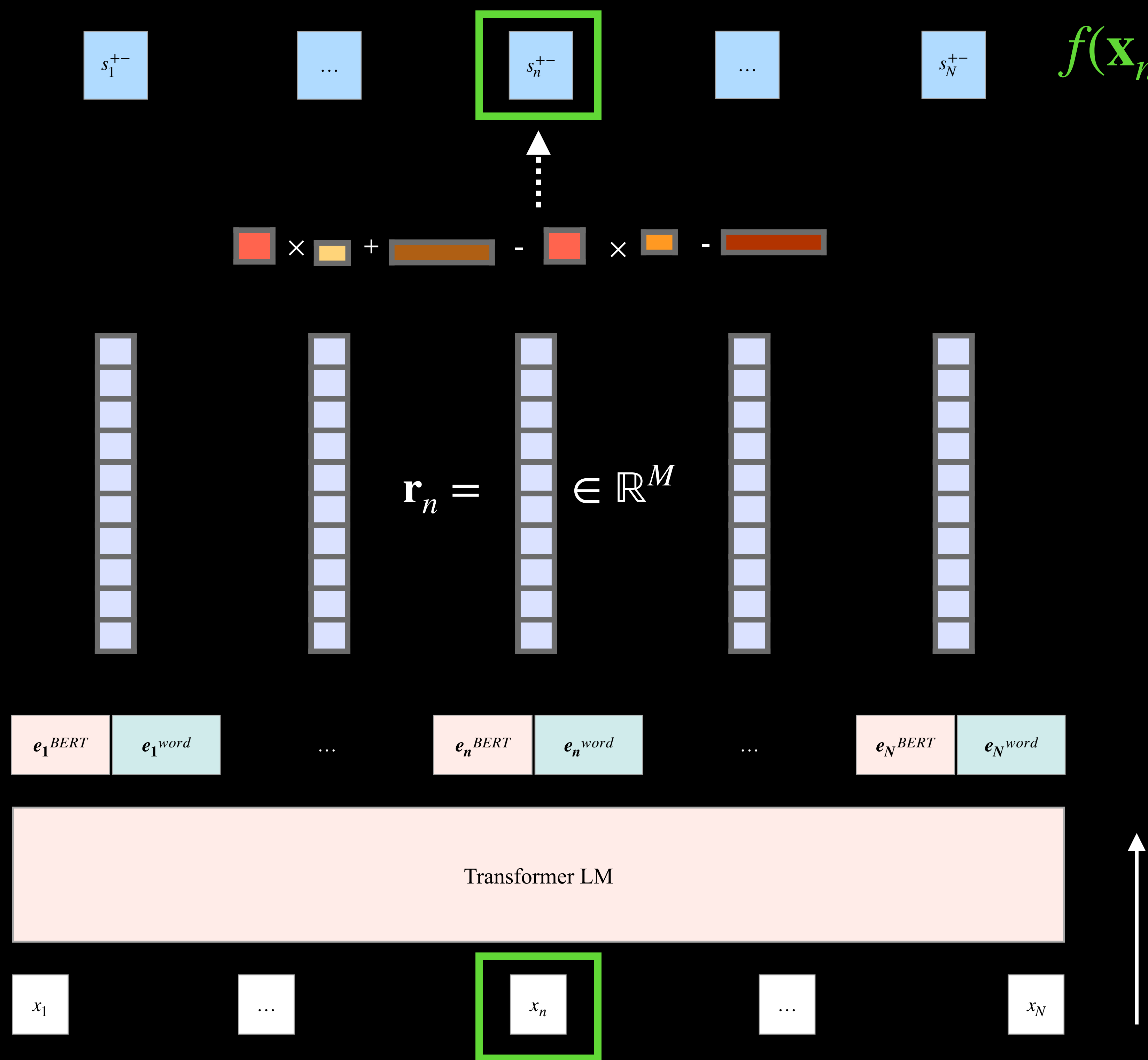


Allen Schmaltz

Vertical (across the support set) Model Decomposition via Dense Matching

$f(\mathbf{x}_n) = s_n^{+-}$

$f(\mathbf{x}_n) = u \circ v$



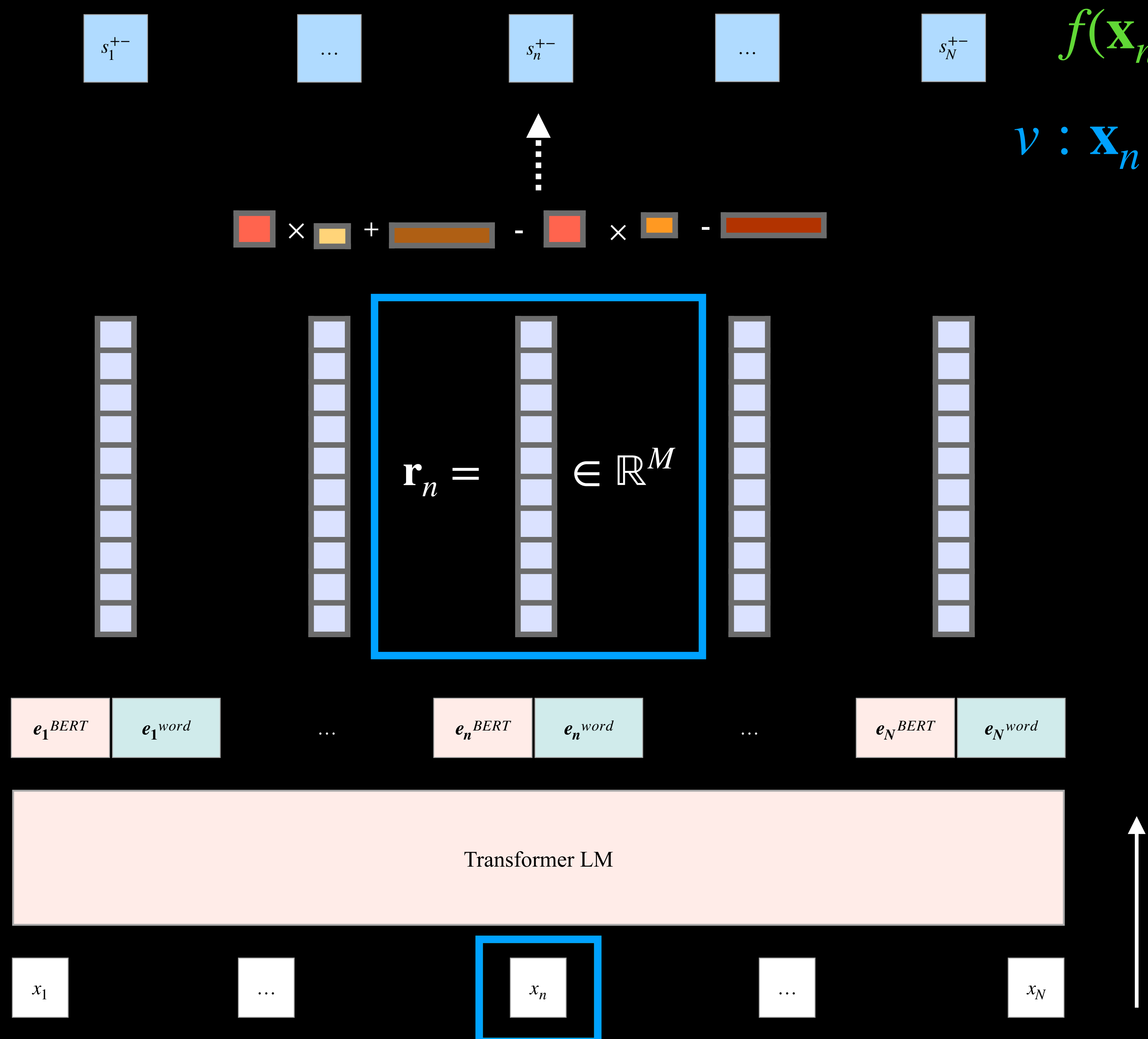
Allen Schmaltz

Vertical (across the support set) Model Decomposition via Dense Matching

$$f(\mathbf{x}_n) = s_n^{+-}$$

$$f(\mathbf{x}_n) = u \circ v$$

$$v : \mathbf{x}_n \mapsto \mathbf{r}_n \in \mathbb{R}^M$$



Allen Schmaltz

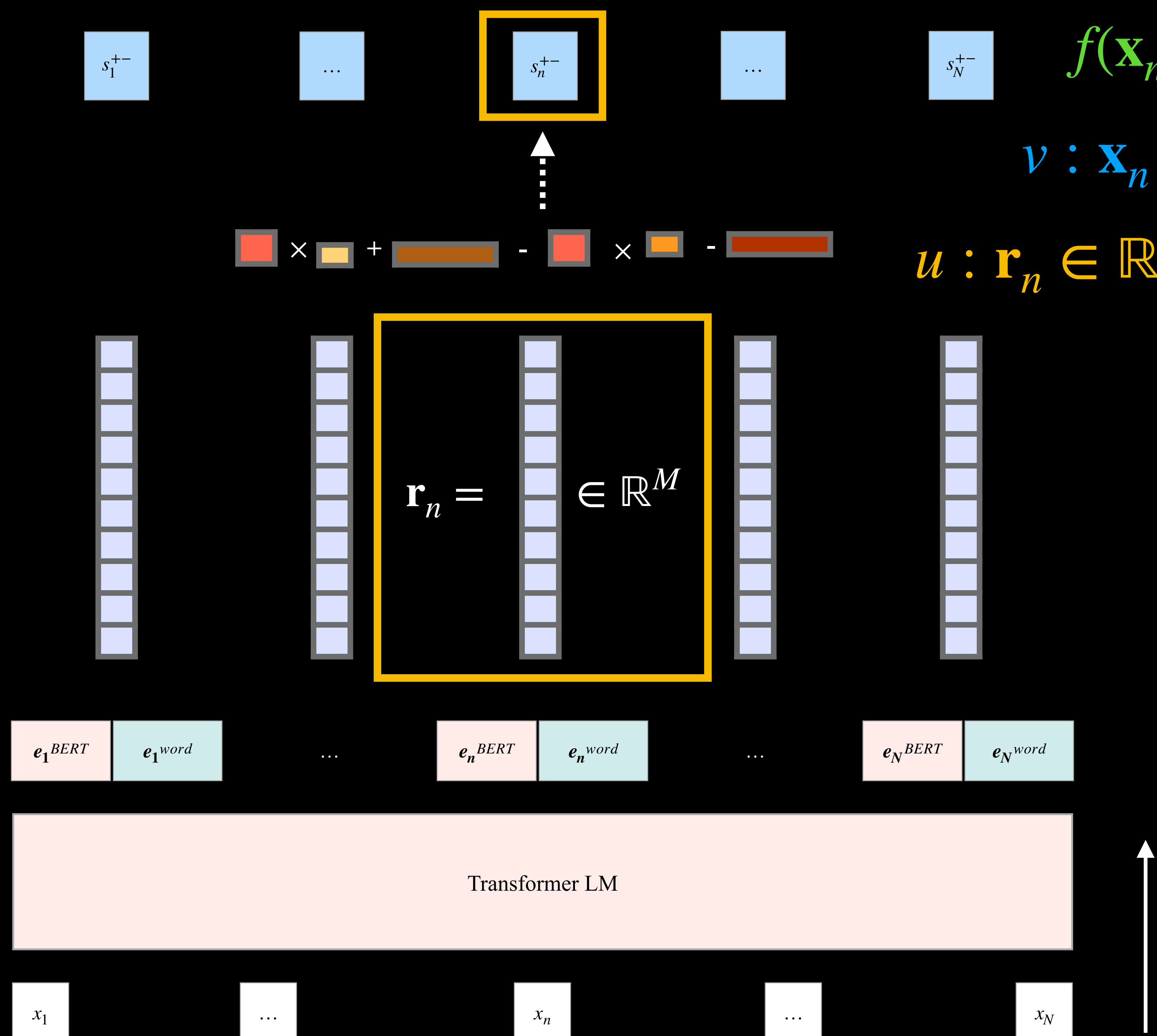
Vertical (across the support set) Model Decomposition via Dense Matching

$$f(\mathbf{x}_n) = s_n^{+-}$$

$$f(\mathbf{x}_n) = u \circ v$$

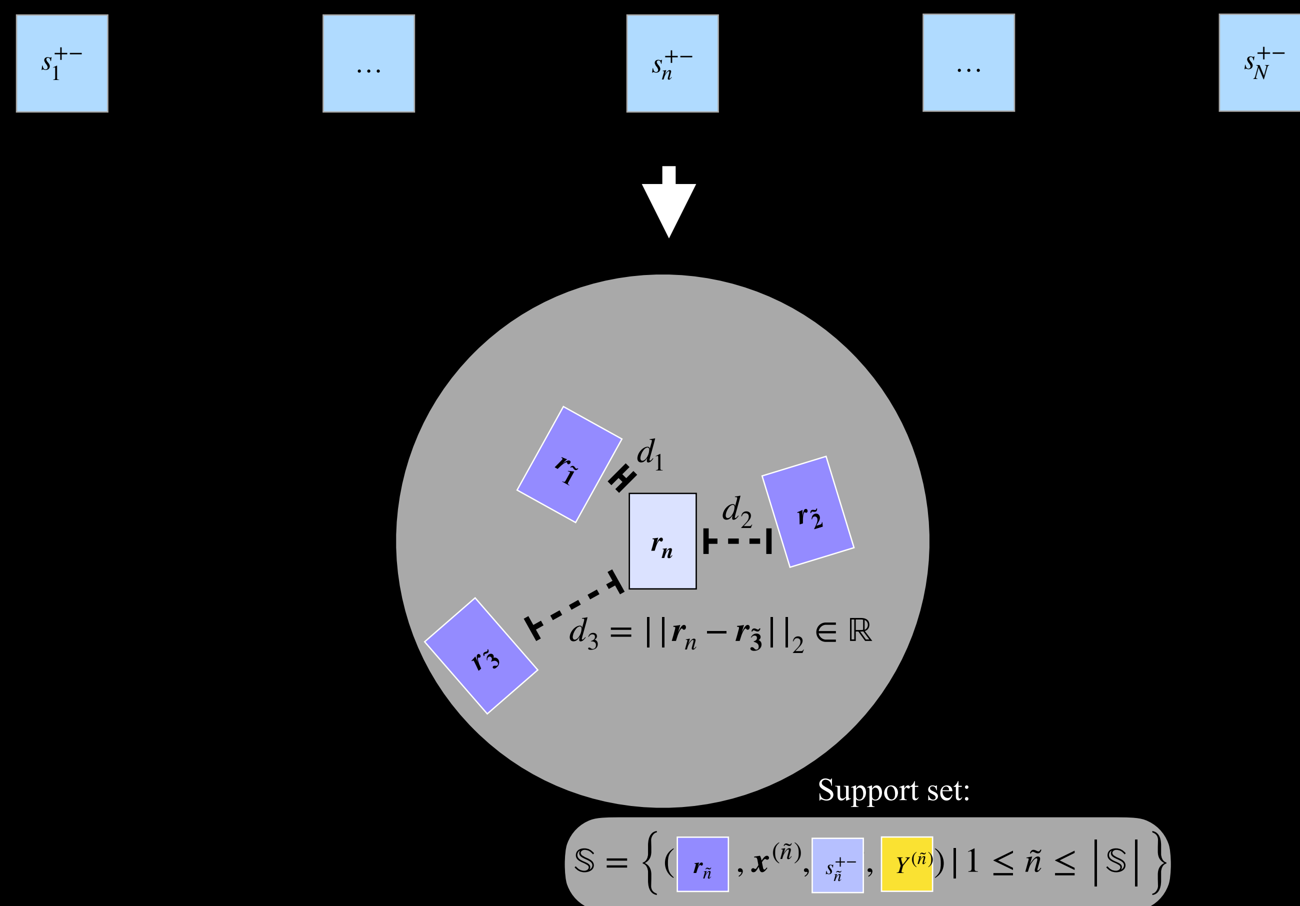
$$v : \mathbf{x}_n \mapsto \mathbf{r}_n \in \mathbb{R}^M$$

$$u : \mathbf{r}_n \in \mathbb{R}^M \mapsto s_n^{+-} \in \mathbb{R}$$



Allen Schmaltz

Vertical (across the support set) Model Decomposition via Dense Matching



$$
\begin{aligned}
s_n^{+-} \approx \beta &+ w_1 \cdot \left(\tanh(s_{\tilde 1}^{+-}) + \gamma \cdot Y^{(\tilde 1)}\right) \\
&+ w_2 \cdot \left(\tanh(s_{\tilde 2}^{+-}) + \gamma \cdot Y^{(\tilde 2)}\right) \\
&+ w_3 \cdot \left(\tanh(s_{\tilde 3}^{+-}) + \gamma \cdot Y^{(\tilde 3)}\right)
\end{aligned}
\qquad
w_k = \frac{\exp(-d_k/\tau)}{\sum_{k'=1}^{3} \exp(-d_{k'}/\tau)}
$$

Model Approximation

Original model output
(from decomposition)

$$\hat{y}_n = \text{sgn}\left(\text{f}(\mathbf{x}_\text{n})\right) = \text{sgn}\left(\text{s}_\text{n}^{+-}\right) \approx$$

$y_k \in \{-1,1\}$ if token-level labels are available; otherwise, document-level $Y^{(k)} \in \{-1,1\}$

$$\hat{y}_n^{KNN} = \text{sgn}\left(\text{f}(\mathbf{x}_\text{n})^\text{KNN}\right) = \text{sgn}\left(\beta + \sum_{\text{k} \in \arg \text{K} \min_{\tilde{\text{n}}} ||\mathbf{r}_\text{n} - \mathbf{r}_{\tilde{\text{n}}}||_2} \text{w}_\text{k} \cdot \left(\tanh(\text{s}_\text{k}^{+-}) + \gamma \cdot \text{Y}^{(\text{k})}\right)\right)$$

K-NN Approximation

$$w_k = \frac{\exp\left(-||\mathbf{r}_n - \mathbf{r}_k||_2/\tau\right)}{\sum_{k' \in \arg K \min_{\tilde{n}} ||\mathbf{r}_n - \mathbf{r}_{\tilde{n}}||_2} \exp\left(-||\mathbf{r}_n - \mathbf{r}_{k'}||_2/\tau\right)}$$

Hyper-parameter: K

Learn β, γ, τ: Loss: $\mathcal{L}_n^{KNN} = -\sigma(s_n^{+-}) \cdot \log \sigma\left(f(\mathbf{x}_n)^{KNN}\right) - (1 - \sigma(s_n^{+-})) \cdot \log\left(1 - \sigma\left(f(\mathbf{x}_n)^{KNN}\right)\right)$

Choose epoch that minimizes: $\delta^{KNN} = \sum_{n \in \textbf{dev}} [\text{sgn}\left(\text{s}_\text{n}^{+-}\right) \neq \text{sgn}\left(\text{f}(\mathbf{x}_\text{n})^\text{KNN}\right)]$

Allen Schmaltz

Empirical Results — Closeness of Approximation

Model Approximation	Model Approximation = Original Model	
	Accuracy	$F_{0.5}$
K-NN Approx. of Transformer ($\text{BERT}_{\text{LARGE}}$) + CNNDecomposition+MinMaxLoss	96.5	76.9
K-NN Approx. of Transformer ($\text{BERT}_{\text{LARGE}}$) + CNNDecomposition (Supervised)	97.0	75.9

Original Model
Transformer ($\text{BERT}_{\text{LARGE}}$) + CNNDecomposition+MinMaxLoss
Transformer ($\text{BERT}_{\text{LARGE}}$) + CNNDecomposition (Supervised)

<u>Token-level</u> FCE K-NN held-out dev set results (Main text: Table 4)

Empirical Results — Closeness of Approximation

Model Approximation	Model Approximation = Ground-truth	Model Approximation = Original Model	
	$F_{0.5}$	Accuracy	$F_{0.5}$
K-NN Approx. of Transformer ($BERT_{LARGE}$) + CNNDecomposition+MinMaxLoss	52.9	96.5	76.9
K-NN Approx. of Transformer ($BERT_{LARGE}$) + CNNDecomposition (Supervised)	59.4	97.0	75.9

Original Model	Original Model = Ground-truth
	$F_{0.5}$
Transformer ($BERT_{LARGE}$) + CNNDecomposition+MinMaxLoss	49.6
Transformer ($BERT_{LARGE}$) + CNNDecomposition (Supervised)	59.5

<u>Token-level</u> FCE K-NN held-out dev set results (Main text: Table 4)

Model Approximation: Error Term

$$\hat{y}_n^{KNN} = \mathrm{sgn}\left(\mathrm{f}(\mathbf{x}_\mathrm{n})^{\mathrm{KNN}}\right) = \mathrm{sgn}\left(\beta + \sum_{\mathrm{k} \in \arg \underset{\tilde{\mathrm{n}}}{\mathrm{K}} \min ||\mathbf{r}_\mathrm{n} - \mathbf{r}_{\tilde{\mathrm{n}}}||_2} \mathrm{w}_\mathrm{k} \cdot \left(\tanh(\mathrm{s}_\mathrm{k}^{+-}) + \gamma \cdot \mathrm{Y}^{(\mathrm{k})}\right)\right) \boxed{+\epsilon}$$

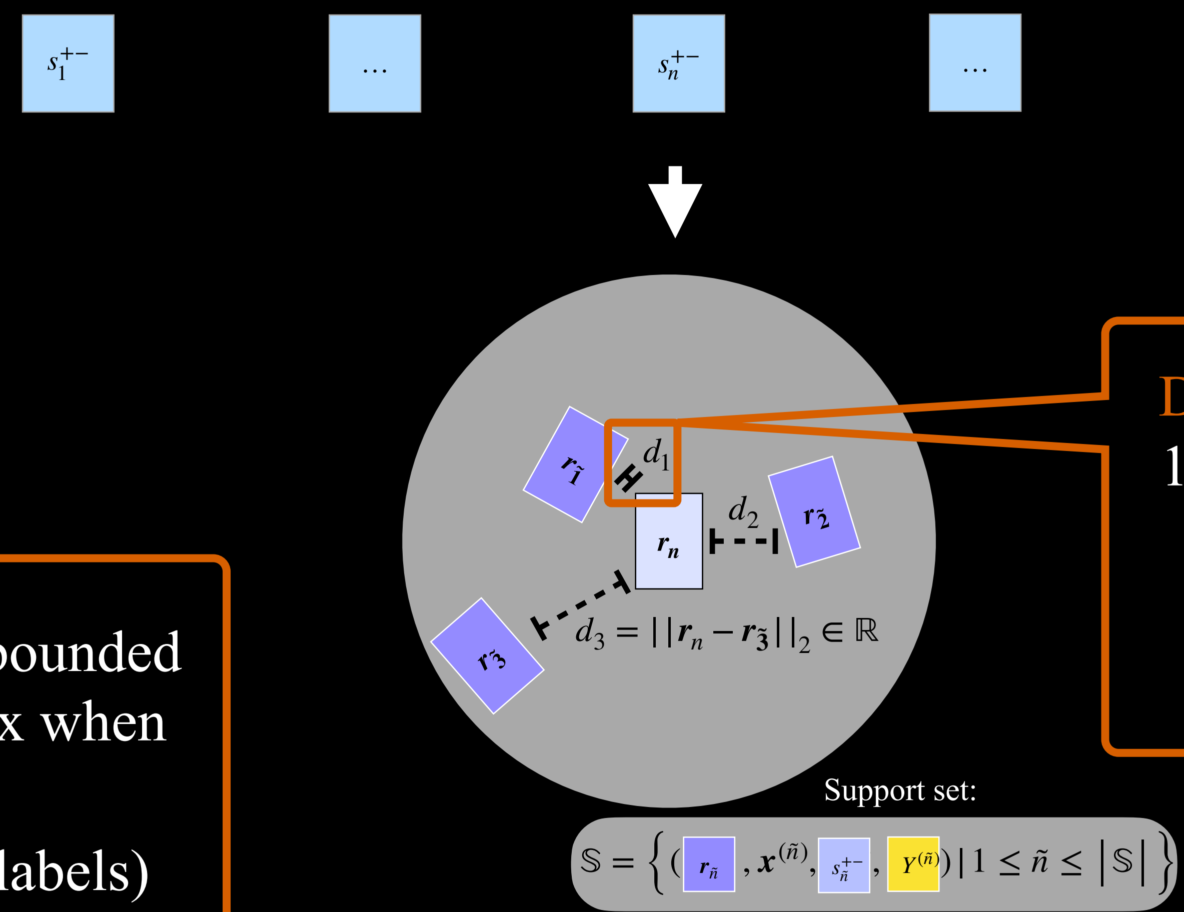
Model Approximation: Error Term

$$\hat{y}_n^{KNN} = \mathrm{sgn}\left(\mathrm{f}(\mathbf{x}_\mathrm{n})^{\mathrm{KNN}}\right) = \mathrm{sgn}\left(\beta + \sum_{\mathrm{k} \in \mathrm{arg}\, \underset{\tilde{\mathrm{n}}}{\mathrm{K}} \min ||\mathbf{r}_\mathrm{n} - \mathbf{r}_{\tilde{\mathrm{n}}}||_2} \mathrm{w}_\mathrm{k} \cdot \left(\tanh(\mathrm{s}_\mathrm{k}^{+-}) + \gamma \cdot \mathrm{Y}^{(\mathrm{k})}\right)\right) \boxed{+\epsilon}$$

Luckily, we can say *a lot* about the errors in practice

Difficult instances to predict also tend to be difficult instances over which to approximate the model.

Leveraging Model Approximations for Prediction Reliability Heuristics & Screening Input Dissimilar to the Support Set

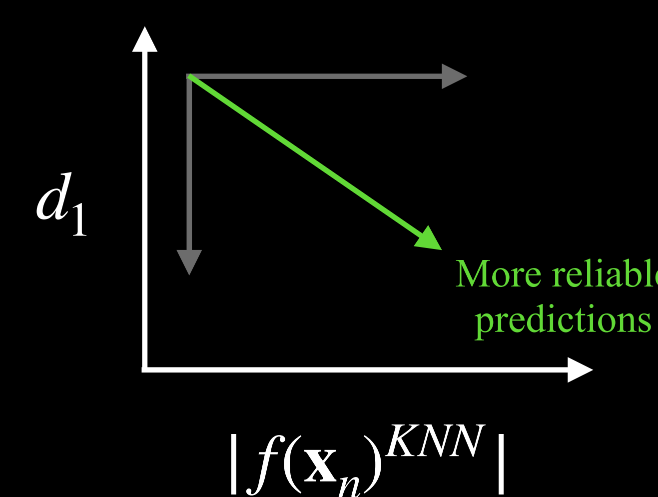


Data uncertainty: Distance to 1st match (d_1), an exogenous factor, captures uncertainty w.r.t. data (training data compared to test data).

Model uncertainty: This bounded value reaches its min/max when $\tanh(s_k^{+-})$ & $Y^{(k)}$ (or y_k, with token-level labels) agree, for all k (assuming $\gamma > 0$).

$$
\begin{aligned}
s_n^{+-} \approx \beta &+ w_1 \cdot \left(\tanh(s_{\tilde 1}^{+-}) + \gamma \cdot Y^{(\tilde 1)}\right) \\
&+ w_2 \cdot \left(\tanh(s_{\tilde 2}^{+-}) + \gamma \cdot Y^{(\tilde 2)}\right) \\
&+ w_3 \cdot \left(\tanh(s_{\tilde 3}^{+-}) + \gamma \cdot Y^{(\tilde 3)}\right)
\end{aligned}
$$

$$
w_k = \frac{\exp(-d_k/\tau)}{\sum_{k'=1}^{3} \exp(-d_{k'}/\tau)}
$$



Magnitude of the K-NN Output

Allen Schmaltz

OOD/Domain-shifted Task Variant

- Add already correct data (NEWS text) to student essay data
    - Distribution of non-errors & language different than training
    - False positive problem
- Analyze ability to detect OOD data & update model (via support set)

Empirical Results—OOD/Domain-Shifted

Model:
K-NN Approx. of Transformer (BERT$_{LARGE}$) +CNNDecomposition+MinMaxLoss

	L^2 distance max constraint	K-NN Output min threshold	Admitted	
$F_{0.5}$	(Class -1, Class 1)	(Class -1, Class 1)	n	n/N
27.0			92597	1.0

Token-level FCE+News2k (domain-shifted) test set results (Main text: Table 6)

Empirical Results—OOD/Domain-Shifted

Model:
K-NN APPROX. OF TRANSFORMER ($\text{BERT}_{\text{LARGE}}$) +CNNDECOMPOSITION+MINMAXLOSS

$F_{0.5}$	L^2 distance max constraint (Class -1, Class 1)	K-NN Output min threshold (Class -1, Class 1)	Admitted n	n/N
27.0			92597	1.0
45.9		(-1.2, 0.8)	38110	0.41
53.5	(34.2, 53.3)		7879	0.09
75.8	(34.2, 53.3)	(-1.2, 0.8)	4180	0.05

Token-level FCE+News2k (domain-shifted) test set results (Main text: Table 6)

Model Approximation: Updatability

$$\hat{y}_n^{KNN} = \mathrm{sgn}\left(\mathrm{f}(\mathbf{x}_\mathrm{n})^{\mathrm{KNN}}\right) = \mathrm{sgn}\left(\beta + \sum_{\mathrm{k} \in \arg \underset{\tilde{\mathrm{n}}}{\mathrm{K}} \min ||\mathbf{r}_\mathrm{n} - \mathbf{r}_{\tilde{\mathrm{n}}}||_2} \mathrm{w}_\mathrm{k} \cdot \left(\tanh(\mathrm{s}_\mathrm{k}^{+-}) + \gamma \cdot \mathrm{Y}^{(\mathrm{k})}\right)\right)$$

Update Support set (representations, labels, meta data)

$$\mathbb{S} = \left\{ \left(\boxed{\boldsymbol{r}_{\tilde{n}}}, \boldsymbol{x}^{(\tilde{n})}, \boxed{s_{\tilde{n}}^{+-}}, \boxed{Y^{(\tilde{n})}} \right) \mid 1 \leq \tilde{n} \leq \left| \mathbb{S} \right| \right\}$$

Allen Schmaltz

Model Approximation: Updatability

$$\hat{y}_n^{KNN} = \mathrm{sgn}\left(\mathrm{f}(\mathbf{x}_\mathrm{n})^{\mathrm{KNN}}\right) = \mathrm{sgn}\left(\beta + \sum_{\mathrm{k} \in \arg \mathrm{K} \min_{\tilde{\mathrm{n}}} ||\mathbf{r}_\mathrm{n} - \mathbf{r}_{\tilde{\mathrm{n}}}||_2} \mathrm{w}_\mathrm{k} \cdot \left(\tanh(\mathrm{s}_\mathrm{k}^{+-}) + \gamma \cdot \mathrm{Y}^{(\mathrm{k})}\right)\right)$$

Update Support set (representations, labels, meta data)

Support set can be viewed as an updatable database

$$\mathbb{S} = \left\{ \left(\boldsymbol{r}_{\tilde{n}}, \boldsymbol{x}^{(\tilde{n})}, s_{\tilde{n}}^{+-}, Y^{(\tilde{n})} \right) \,\middle|\, 1 \le \tilde{n} \le \left|\mathbb{S}\right| \right\}$$

Empirical Results—OOD/Domain-Shift Updatability

Model:
K-NN Approx. of Transformer ($\text{BERT}_{\text{LARGE}}$) +CNNDecomposition+MinMaxLoss

Model	Training set	Support set	$F_{0.5}$
K-NN Approx.	FCE	FCE	27.0
K-NN Approx.	FCE	FCE+OOD	46.3

Original training set

+50k News data

Token-level FCE+News2k (domain-shifted) test set results (Main text: Table 5)

Allen Schmaltz

Empirical Results—OOD/Domain-Shift Updatability

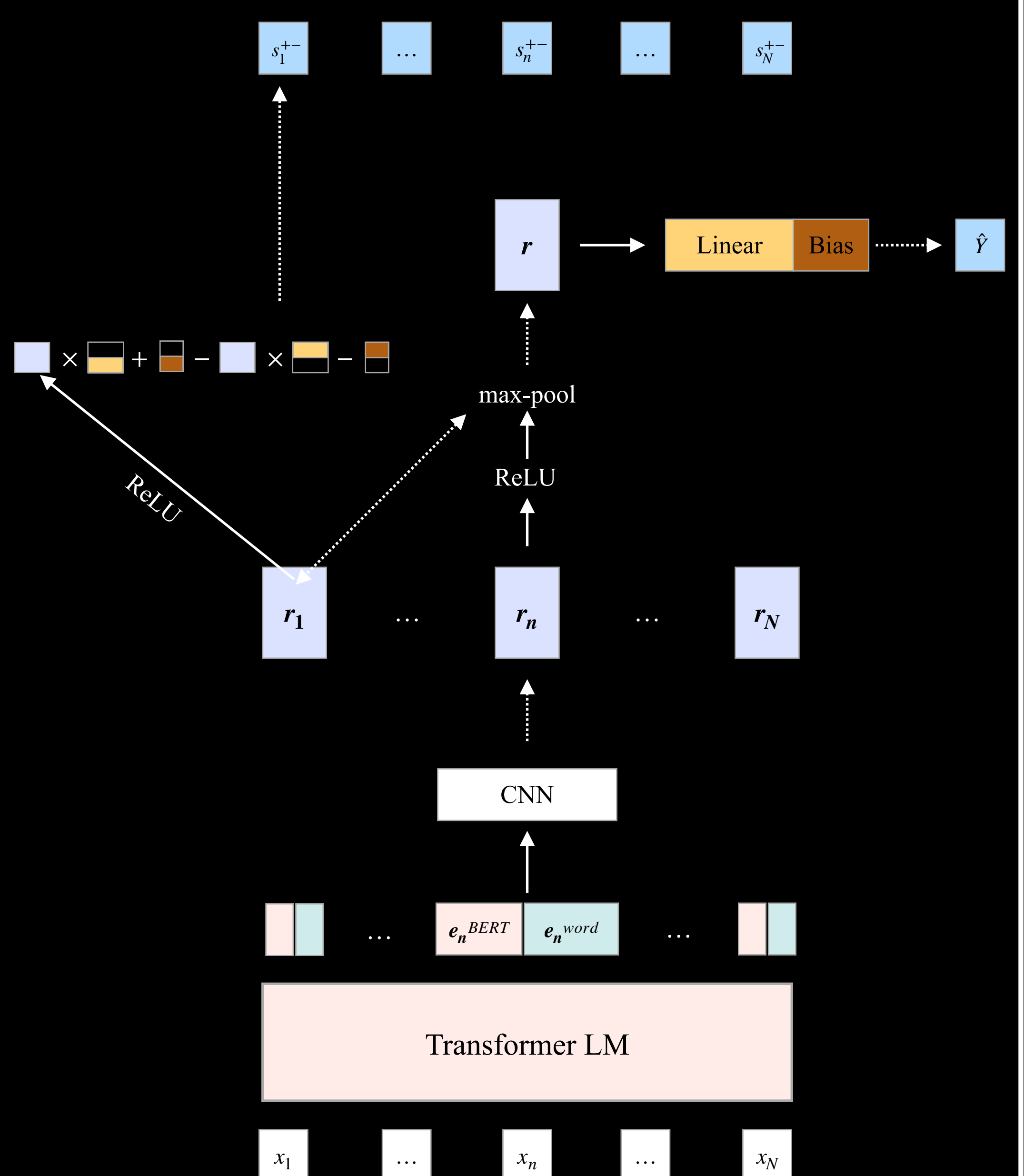
Model:
K-NN Approx. of Transformer ($BERT_{LARGE}$) +CNNDecomposition+MinMaxLoss

Model	Training set	Support set	$F_{0.5}$
K-NN Approx.	FCE	FCE	27.0
K-NN Approx.	FCE	FCE+OOD	46.3
Original Model	FCE	-	25.8
Original Model	FCE+OOD	-	33.3

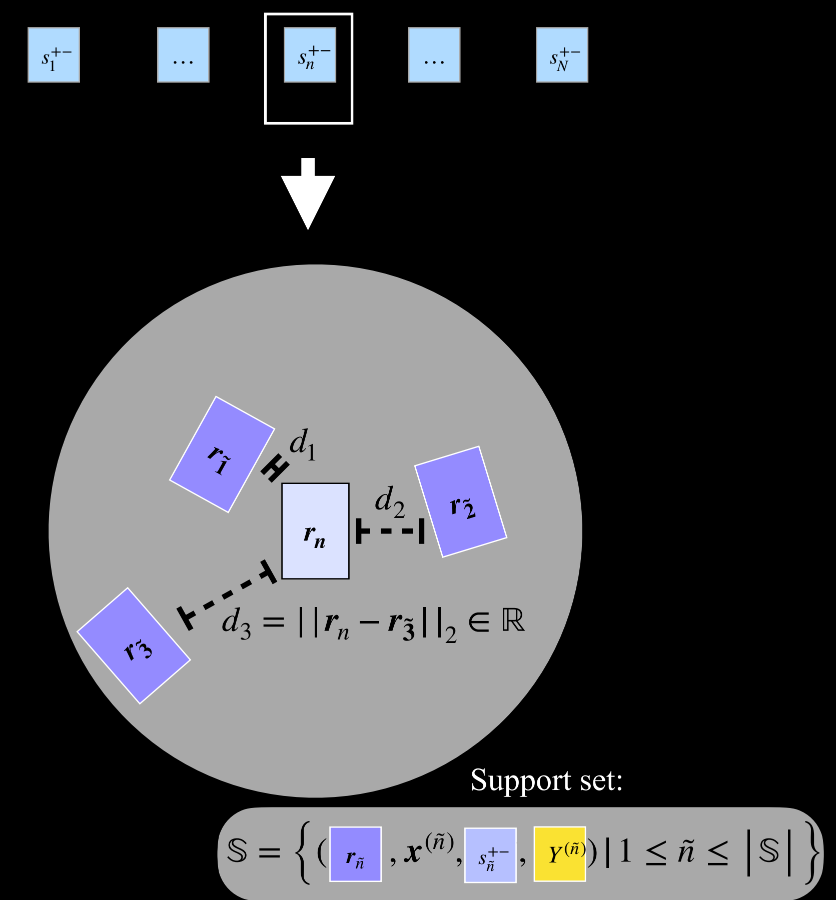
Token-level FCE+News2k (domain-shifted) test set results (Main text: Table 5)

Summary

Sequence Labeling via a Convolutional Decomposition



K-NN Approximation



$$s_n^{+-} \approx \beta + w_1 \cdot \left(\tanh(s_{\tilde{1}}^{+-}) + \gamma \cdot Y^{(\tilde{1})}\right) + w_2 \cdot \left(\tanh(s_{\tilde{2}}^{+-}) + \gamma \cdot Y^{(\tilde{2})}\right) + w_3 \cdot \left(\tanh(s_{\tilde{3}}^{+-}) + \gamma \cdot Y^{(\tilde{3})}\right)$$

$$w_k = \frac{\exp(-d_k/\tau)}{\sum_{k'=1}^{3} \exp(-d_{k'}/\tau)}$$

Allen Schmaltz

Appendix

Presentation Appendix: *Parting Thoughts*

- Predictions from deep networks become more reliable as the following increase (potentially at expense of lower admitted N):
  - Closer distances to the support set ↓
  - Greater agreement between predictions and labels (i.e., stronger models, greater K-NN output magnitude) ↓
  - More labeled data at the desired resolution of analysis ↓

The decompositions described today provide a new means of analyzing and constraining the predictions against the data, yielding new levers for deploying and interpreting networks

More reliable predictions ⟵

Presentation Appendix: *Not Covered Today*

- Aggregate, comparative feature extraction/importance

$$\text{E.g., } \mathrm{ngram}^{-}_{\mathrm{n:n+(z-1)}} = \sum_{\mathrm{i=n}}^{\mathrm{n+(z-1)}} (\mathrm{s}_{\mathrm{i}}^{-} - \mathrm{b}_1)$$

- Decision rules

  E.g., only admit true positive matches:

$$\hat{y}_{N+1} = f(x_{N+1}) \cdot \left[f(x_{N+1}) = f(x_n) \wedge f(x_n) = y_n\right] + NULL \cdot \left[f(x_{N+1}) \neq f(x_n) \vee f(x_n) \neq y_n\right], \text{where } n = \underset{n \in \{1,\ldots,N\}}{\arg\min} \; ||r_n - r_{N+1}||_2$$

- Additional tasks and datasets, further illustrating:
  - Implications/juxtaposition of OOD robustness vs. detection and updatability
  - Ability to detect features for text analysis of large document sets

Presentation Appendix: Additional Considerations

- Alignment ("diagonally within sequence")

E.g., NLI & fact verification

  - Use bi-encoder, or masked cross-encoder, instead
- "Non-sparse" fully-supervised labeling for long sequences
  - Larger *M* makes dense search more expensive
  - If sparse feature detection not needed, can dispense with max-pool (& thus, the *horizontal* decomposition)

Can then proceed to use the K-NN model approximation as described today