Alignment Methods for Attention-based Neural Machine Translation

Arne Nix

arne.nix@rwth-aachen.de

September 26, 2016, Aachen

Human Language Technology and Pattern Recognition
Computer Science Department, RWTH Aachen University

Outline

Motivation

- **Statistical Machine Translation:**
  - **Goal: For source sentence $f_1^J := f_1 \ldots f_j \ldots f_J$ find a translation hypothesis $\hat{e}_1^{\hat{I}} := \hat{e}_1 \ldots \hat{e}_i \ldots \hat{e}_{\hat{I}}$ such that:**

$$\hat{e}_1^{\hat{I}} = \operatorname*{argmax}_{I, e_1^I} \left\{ Pr(e_1^I | f_1^J) \right\}$$

- **Neural Machine Translation:**
  - **Use recurrent neural network to model $Pr(e_1^I | f_1^J)$**
  - **Attention-based neural machine translation state-of-the-art on many tasks**

Related Work

I. Sutskever, O. Vinyals, Q. Le [Sutskever & Vinyals^{+} 14]:

Sequence to sequence learning with neural networks.
NIPS, December 2014.

- Introducing the encoder-decoder model
- Application to machine translation

D. Bahdanau, K. Cho, Y. Bengio [Bahdanau & Cho^{+} 15]:

Neural machine translation by jointly learning to align and translate.
ICLR, May 2015.

- Introducing an attention mechanism to neural machine translation
- State of the art for neural machine translation

Related Work

J. Chorowski, D. Bahdanau et al. [Chorowski & Bahdanau^{+} 15]:

Attention-Based Models for Speech Recognition.
***NIPS, December 2015*.**

- **Applies the attention mechanism to ASR**
- **Introduces convolutional alignment feedback**

W. Chen, E. Matusov et al. [Chen & Matusov^{+} 16]:

Guided Alignment Training for Topic-Aware Neural Machine Translation.

- **Extends standard network error by additional alignment error**

Related Work

Z. Tu, Z. Lu et al. [Tu & Lu$^+$ 16]:

Modeling coverage for neural machine translation.
***ACL, August 2016*.**

- **First empirical alignment analysis of attention-based alignments**
- **Introduces SAER measure to evaluate soft alignments**
- **Extends attention models by coverage vector**

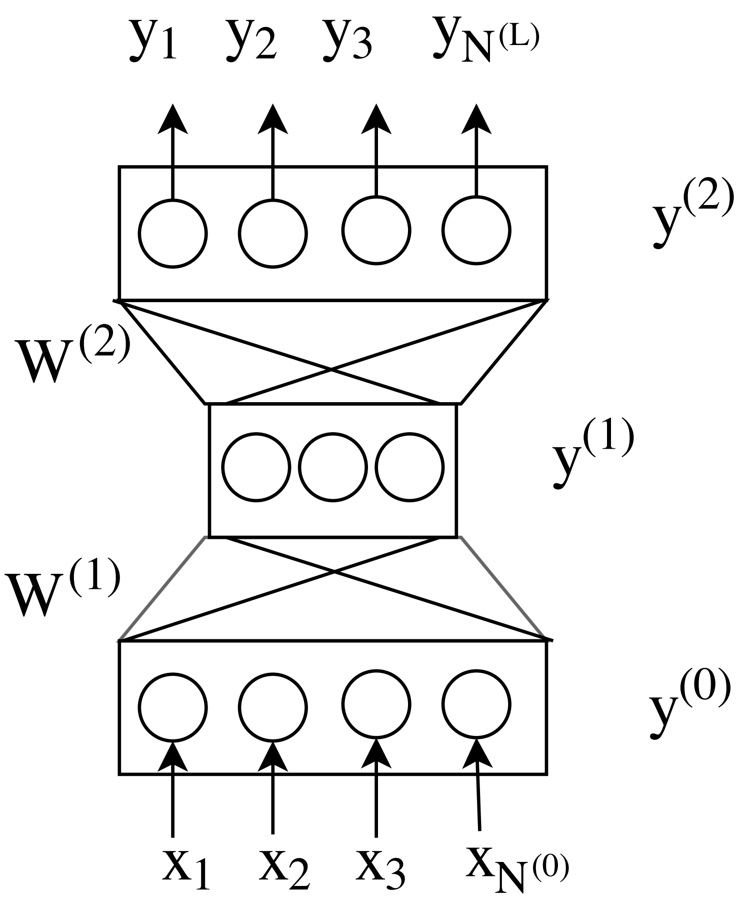
B. Zhang, D. Xiong, J. Su [Zhang & Xiong$^+$ 16]:

Recurrent Neural Machine Translation

- **Replaces attention-mechanism by a RNN that computes the context vector**
- **Recurrent over the source representation**
- **Slower by a factor of 3**

Outline

Neural Networks



Feed forward neural network

- **Activation of layer l:**

$$\boldsymbol{y}^{(l)} = \boldsymbol{\sigma}^{(l)}\left(\underbrace{\boldsymbol{W}^{(l)} \cdot \boldsymbol{y}^{(l-1)} + \boldsymbol{b}^{(l)}}_{=:\boldsymbol{z}^{(l)}}\right)$$

$$\boldsymbol{y}^{(0)} = \boldsymbol{x}$$

- **Common activation functions:**

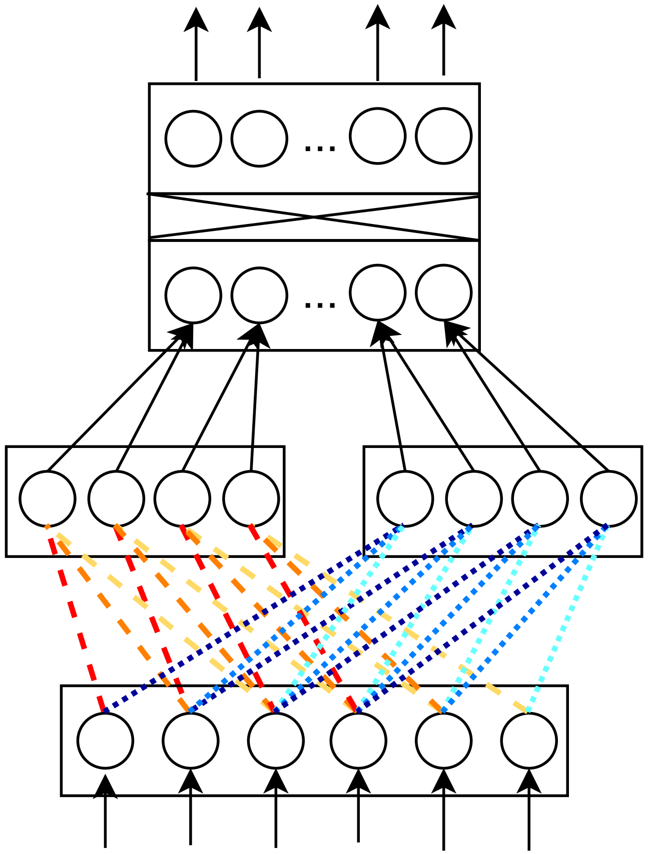
$$\sigma_{\text{sigmoid}}(z) = \frac{1}{1 + \exp(-z)}$$

$$\sigma_{\tanh}(z) = \tanh(z) = \frac{\exp(2z) - 1}{\exp(2z) + 1}$$

- **Output normalization:**

$$p_\theta(c|x) = \frac{\exp(y_c^{(L)})}{\sum_k \exp(y_k^{(L)})} \; \forall c = 1, \ldots, N^{(L)}$$

Convolutional Neural Networks



CNN with $M^{(l)} = 2$ and $D^{(l)} = 3$

- **Apply $M^{(l)}$ filters of width $D^{(l)}$:**

$$\boldsymbol{y}^{(l)} = \boldsymbol{\sigma}(\boldsymbol{W}^{(l)} * \boldsymbol{y}^{(l-1)})$$

where $\boldsymbol{W}^{(l)} \in \mathbb{R}^{M^{(l)} \times D^{(l)}}$ and $\boldsymbol{D}^{(l)} = 2 \cdot \boldsymbol{k}^{(l)} + 1$

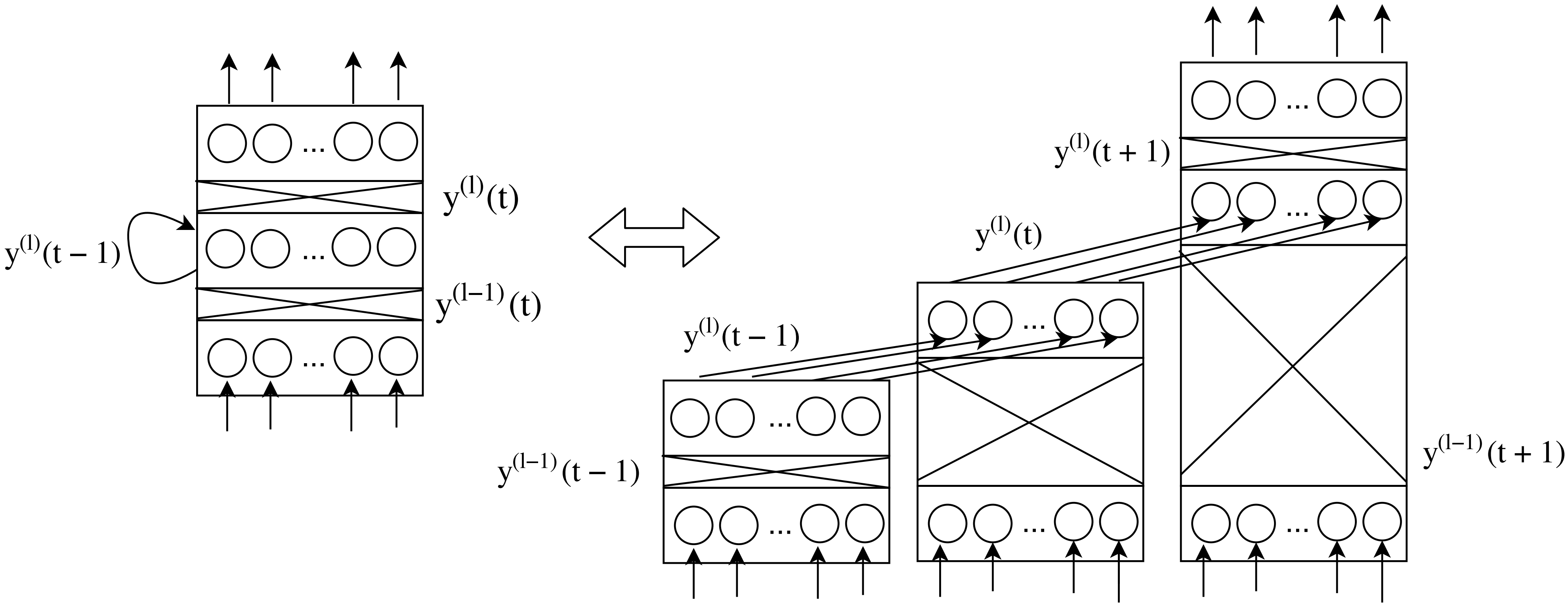
- **Activation of neuron j in layer l:**

$$\boldsymbol{y}_j^{(l)} = \boldsymbol{\sigma}\Big(\sum_{i=j-k^{(l)}}^{j+k^{(l)}} \boldsymbol{w}_{j-i}^{(l)} \cdot \boldsymbol{y}_i^{(l-1)}\Big)$$

Recurrent Neural Networks

- **Activation of layer l for timestep t:**

$$\boldsymbol{y}^{(l)}(t) = \boldsymbol{\sigma}^{(l)}\left(\boldsymbol{W}^{(l)}\boldsymbol{y}^{(l-1)}(t) + \boldsymbol{U}^{(l)}\boldsymbol{y}^{(l)}(t-1)\right)$$

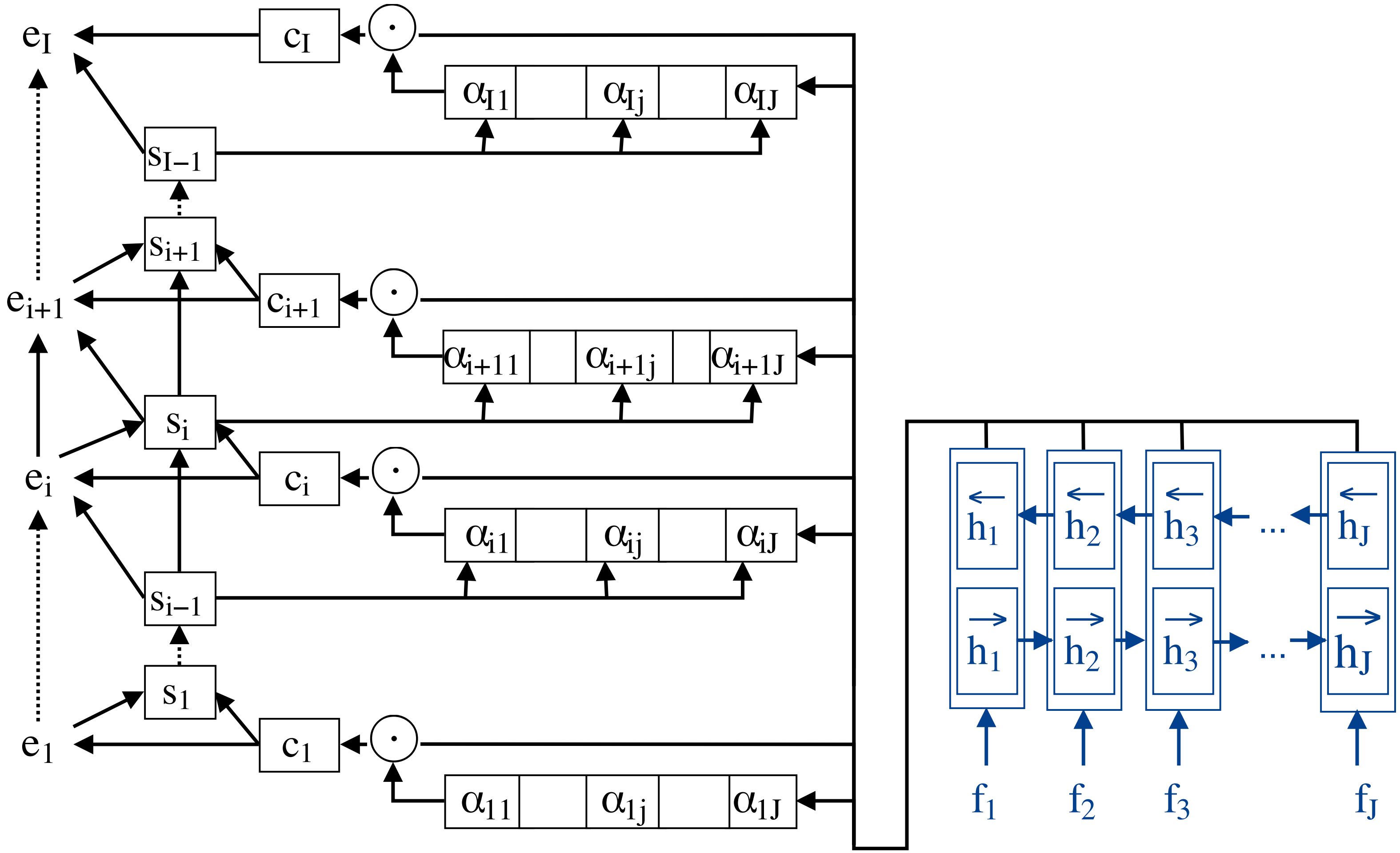


RNN with its equivalent unfolded in time for three time steps.

Outline

Attention Based NMT [Bahdanau & Cho+ 15]

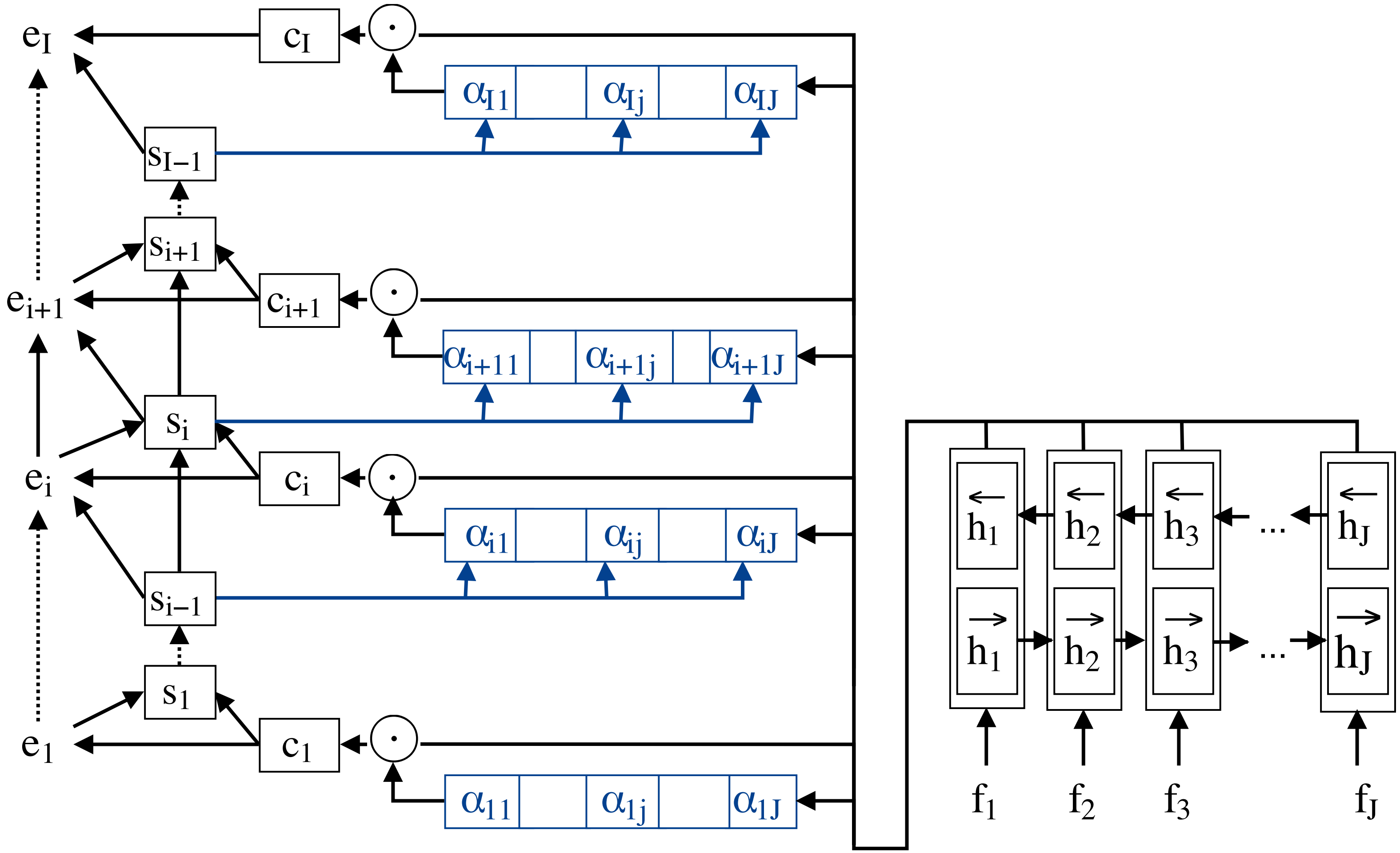
- Bidirectional RNN encodes source sentence f_1^J into $\overrightarrow{h}_1^J$ and $\overleftarrow{h}_1^J$
- $h_j := [\overrightarrow{h}_j^T; \overleftarrow{h}_j^T]^T$



Attention Based NMT [Bahdanau & Cho+ 15]

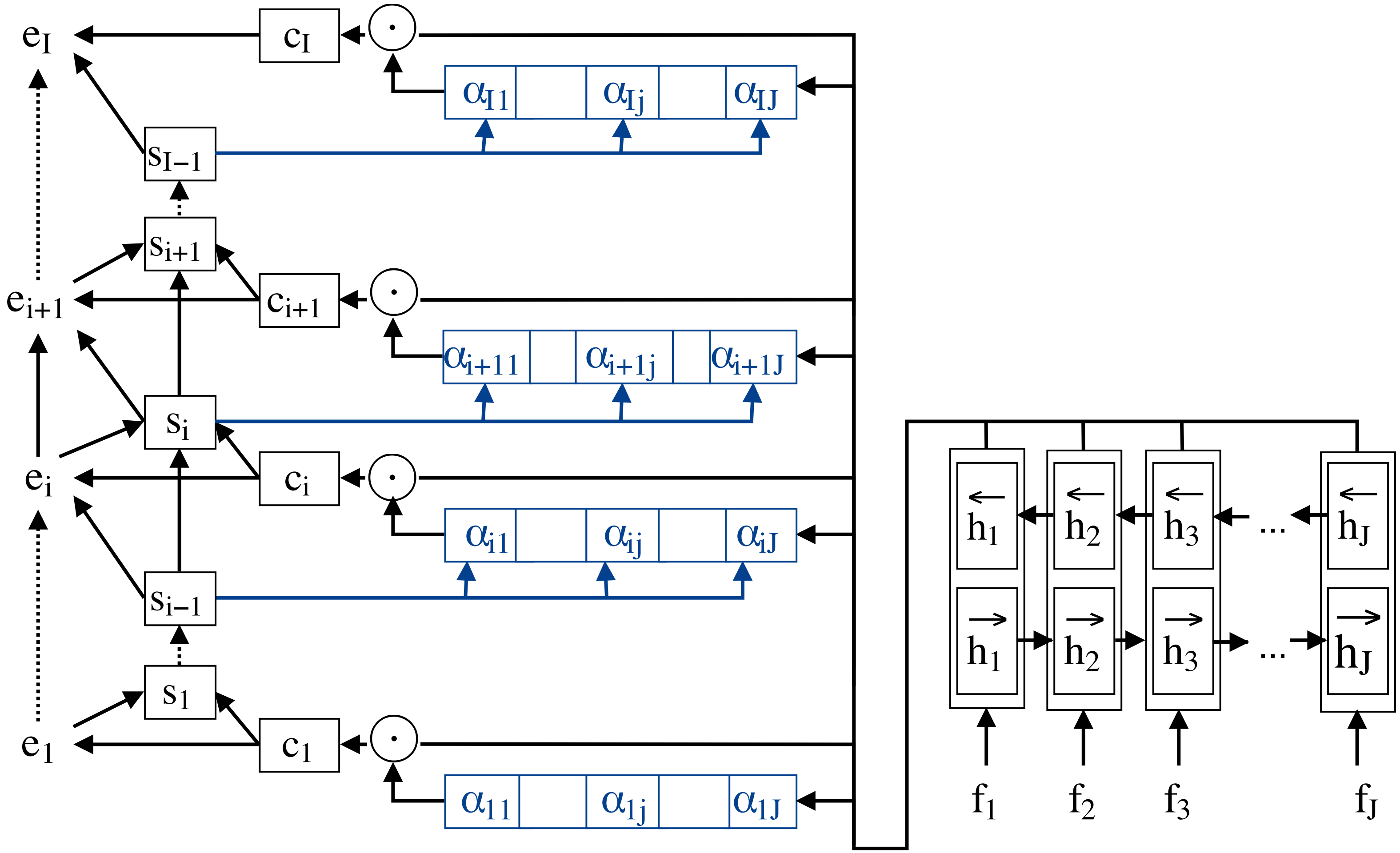
- **Energies computed through MLP:** $\tilde{\alpha}_{ij} = v_a^T \tanh(W_a s_{i-1} + U_a h_j)$

$W_a \in \mathbb{R}^{n \times n}, U_a \in \mathbb{R}^{n \times 2n}, v_a \in \mathbb{R}^n$**: weight parameters**



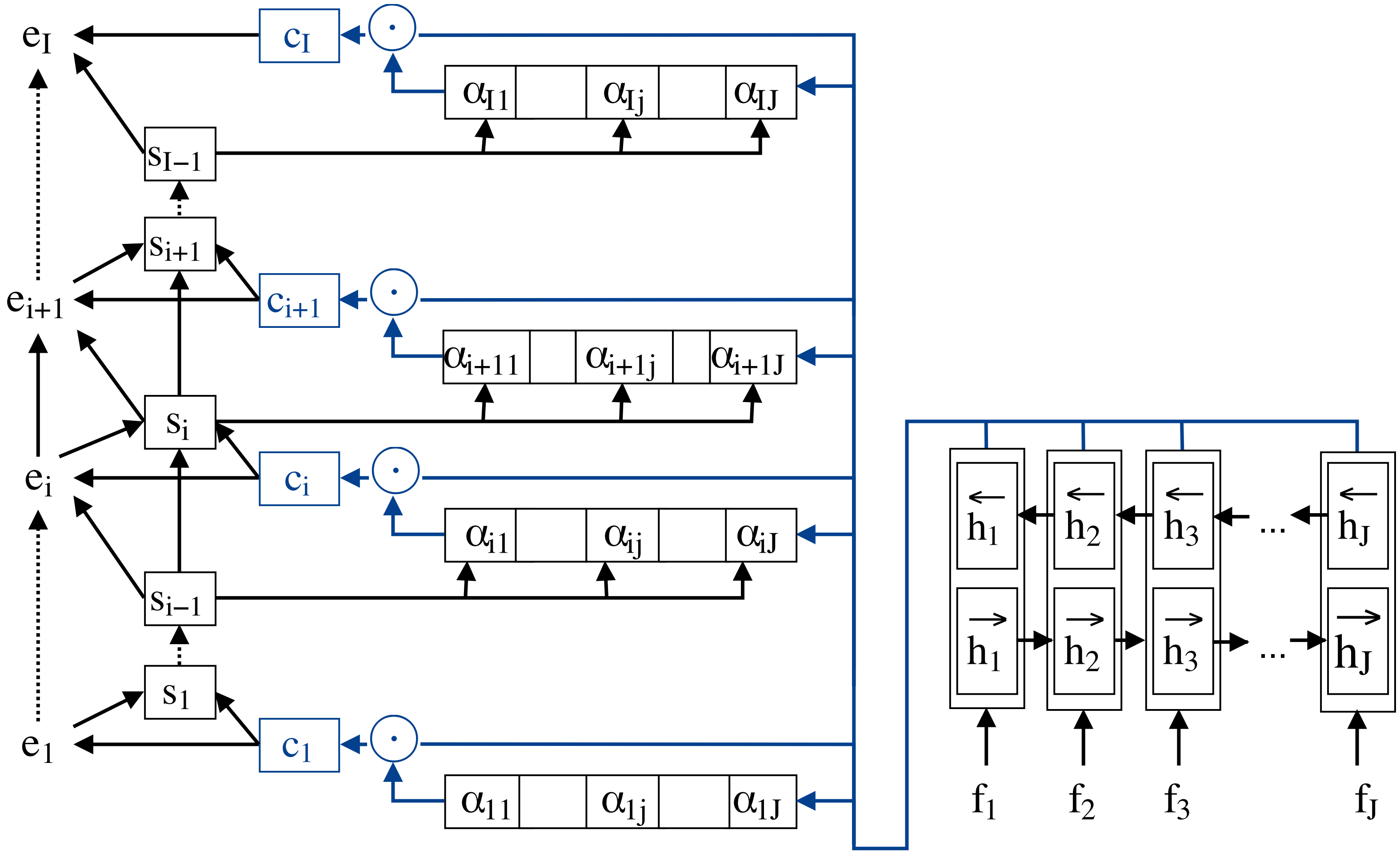
Attention Based NMT [Bahdanau & Cho+ 15]

- **Attention weights normalized with softmax:** $\alpha_{ij} = \frac{\exp(\tilde{\alpha}_{ij})}{\sum_{k=1}^{J} \exp(\tilde{\alpha}_{ik})}$



Attention Based NMT [Bahdanau & Cho+ 15]

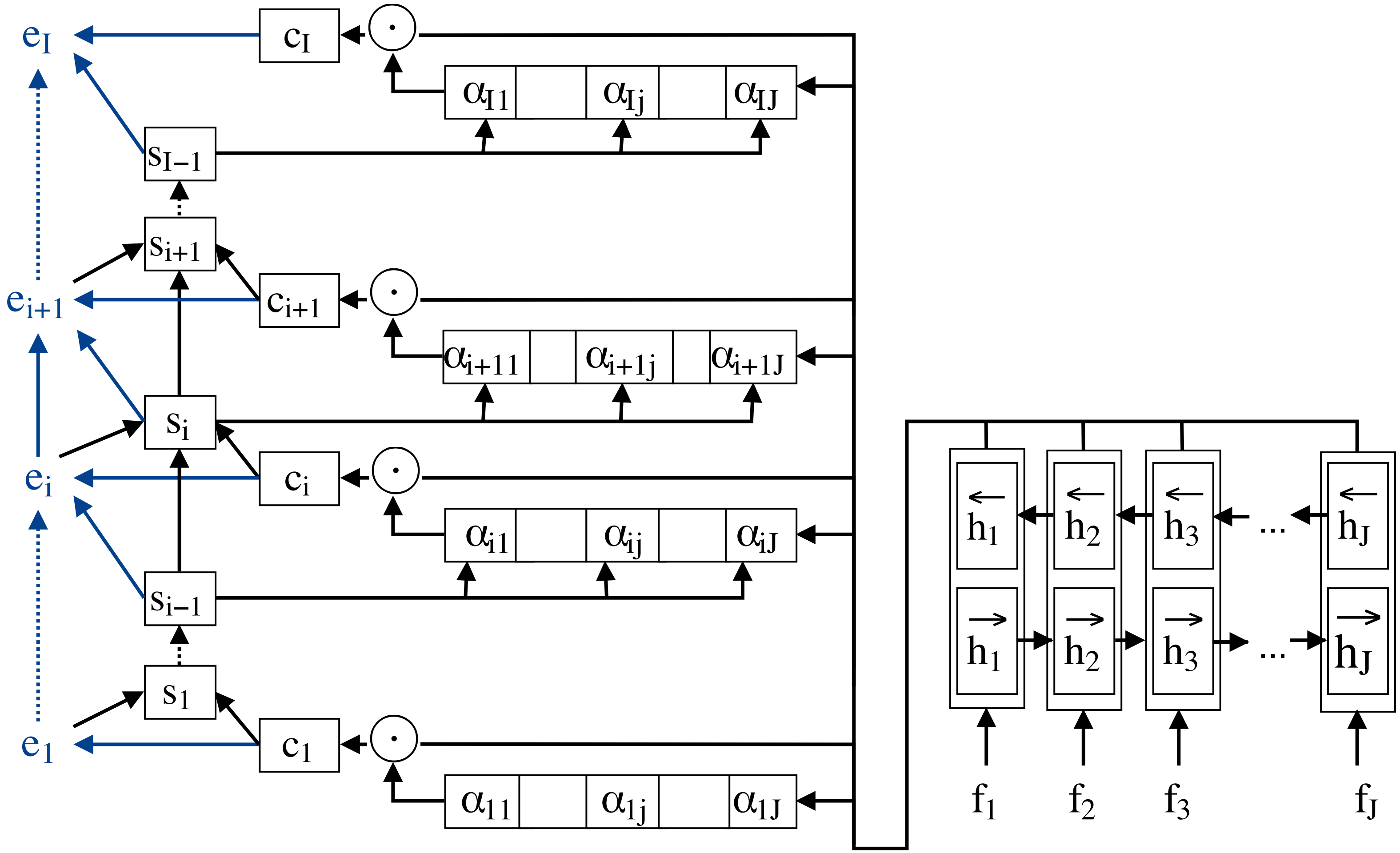
- **Context vector as weighted sum:** $c_i = \sum_{j=1}^{J} \alpha_{ij} h_j$



Attention Based NMT [Bahdanau & Cho+ 15]

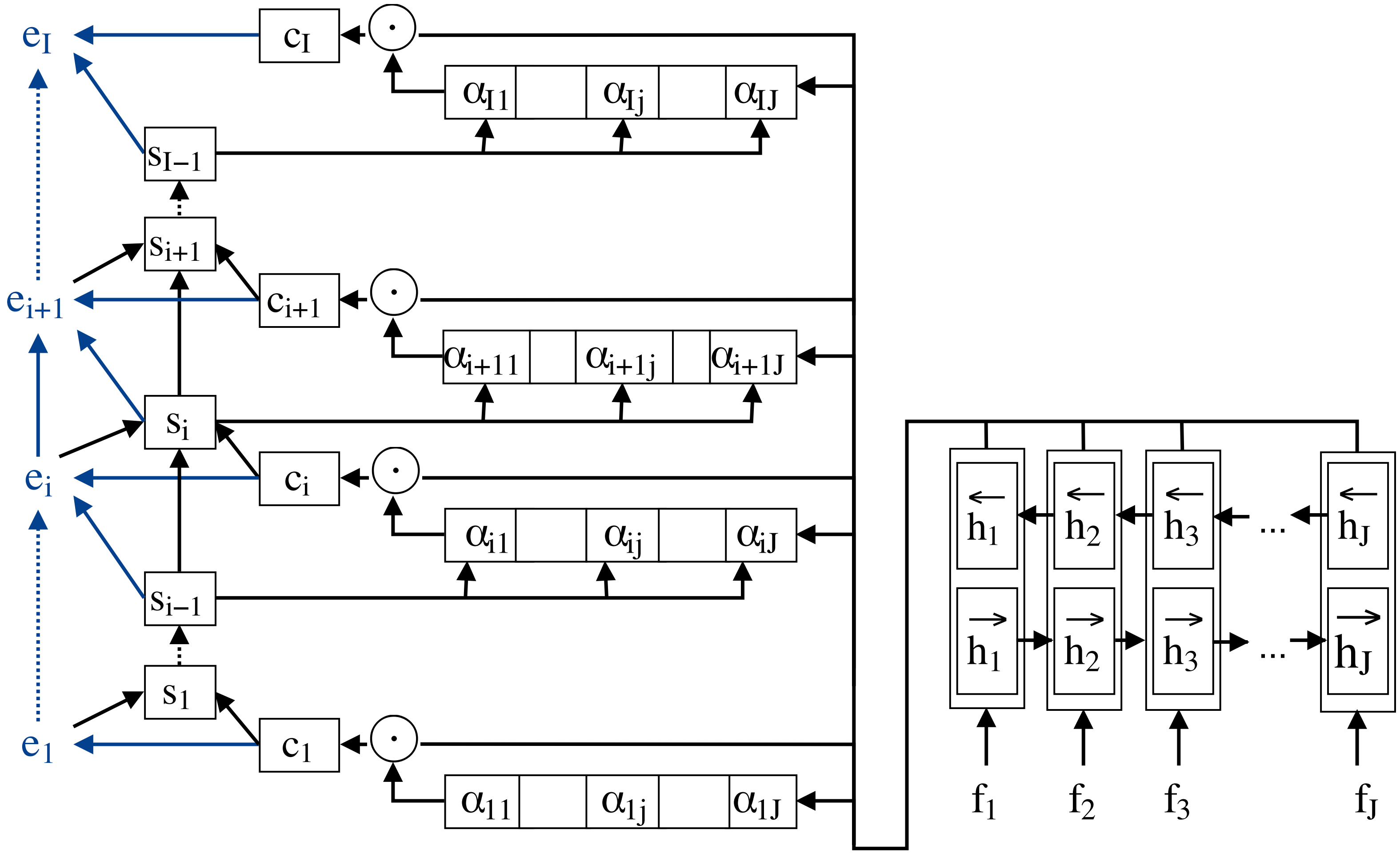
- **Neural network output:** $p(e_i|e_1^{i-1}, f_1^J) = g_{\text{out}}(e_{i-1}, s_{i-1}, c_i)$

g_{out}: **output function**



Attention Based NMT [Bahdanau & Cho$^+$ 15]

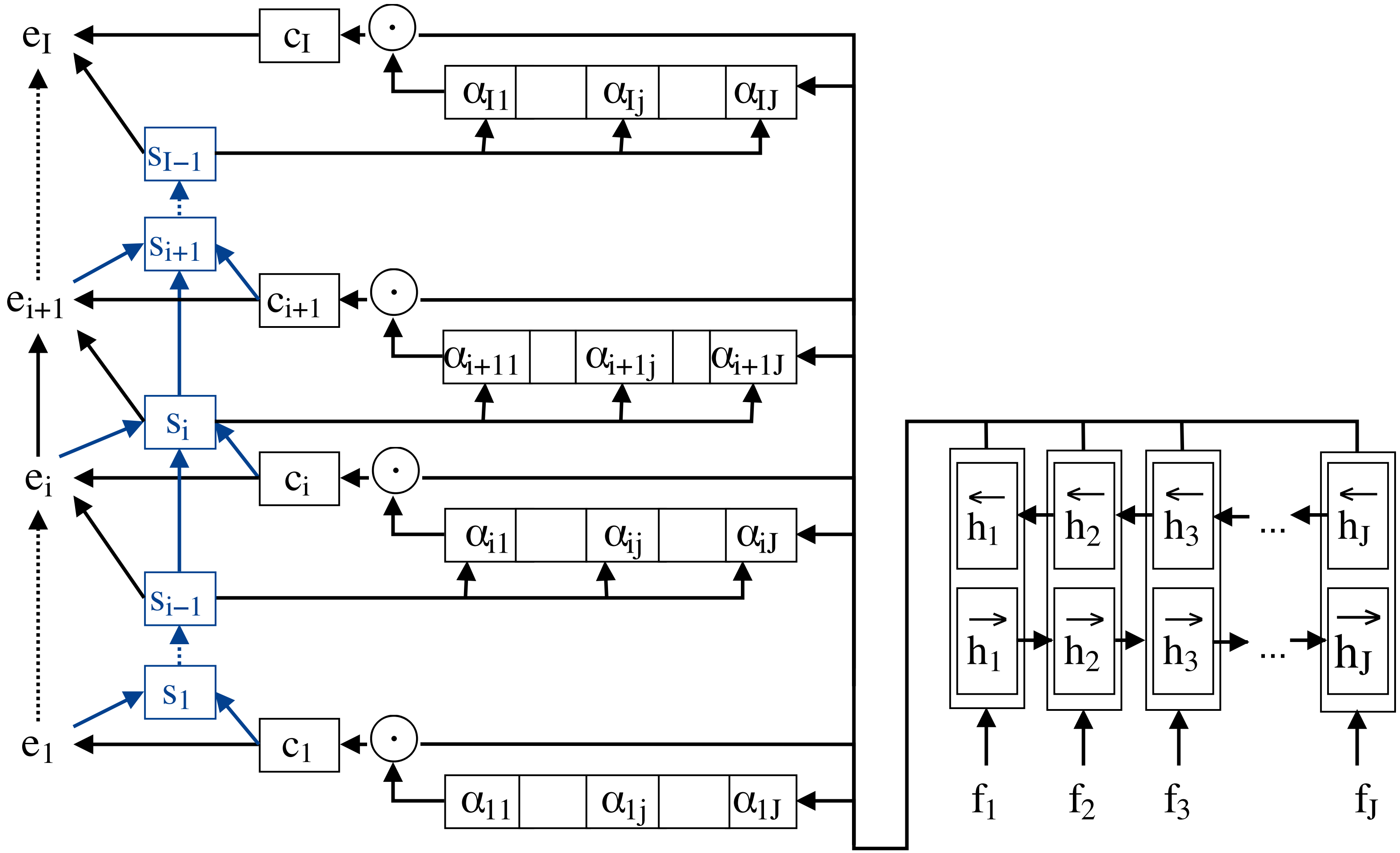
$$Pr(e_1^I|f_1^J) = \prod_{i=1}^{I} p(e_i|e_1^{i-1}, f_1^J)$$



Attention Based NMT [Bahdanau & Cho+ 15]

- **Hidden decoder state:** $s_i = g_{\textsf{dec}}(e_i, c_i; s_{i-1})$

$g_{\textsf{dec}}$**: gated recurrent unit**



IWSLT 2013 De-En

		German	English
`train (full data)`	Sentences	4.3M	
	Running Words	108M	108M
	Vocabulary	836K	792K
`train (in-domain)`	Sentences	138K	
	Running Words	2.6M	2.7M
	Vocabulary	75K	50K
`dev`	Sentences	887	
	Running Words	20K	20.1K
	Vocabulary	4.1K	3.3K
	OOVs with full vocabulary (Rate)	468 (2.3%)	197 (0.9%)
	OOVs with 30K shortlist (Rate)	1346 (6.7%)	656 (3.3%)
`eval`	Sentences	1436	
	Running Words	27.2K	27.6K
	Vocabulary	4.6K	3.7K
	OOVs with full vocabulary (Rate)	449(1.6%)	1110(4.1%)
	OOVs with 30K shortlist (Rate)	1526 (5.6%)	1716 (6.5%)
`test`	Sentences	1565	
	Running Words	31.6K	32.6K
	Vocabulary	5.0K	3.9K
	OOVs with full vocabulary (Rate)	677 (2.1%)	1377 (4.4%)
	OOVs with 30K shortlist (Rate)	1811 (5.7%)	2000 (6.4%)

WMT 2016 En-Ro

		English	Romanian
`train`	Sentences	605K	
	Running Words	15.5M	15.8M
	Vocabulary	92K	128K
	OOV Rate with 30k short list	0.7%	1.8%
`newsdev2016_1`	Sentences	1000	
	Running Words	24.7K	26.7K
	Vocabulary	5K	6.4K
	OOVs (Rate)	938 (3.8%)	1504 (5.6%)
	OOVs with 30k short list (Rate)	1602 (6.5%)	2987 (11.2%)
`newsdev2016_2`	Sentences	999	
	Running Words	25.2K	25.6K
	Vocabulary	4.7K	6.4K
	OOVs (Rate)	733 (2.9%)	1296 (5.0%)
	OOVs with 30k short list (Rate)	1289 (5.1%)	2992 (11.7%)
`newstest2016`	Sentences	1999	
	Running Words	48K	49.7K
	Vocabulary	7.1K	10.3K
	OOVs (Rate)	1309 (2.7%)	2538 (5.1%)
	OOVs with 30k short list (Rate)	2368 (4.9%)	5847 (11.7%)

Europarl De-En

		German	English
`train (full data)`	Sentences	1.2M	
	Running Words	32M	34M
	Vocabulary	305K	100K
`align-test`	Sentences	504	
	Running Words	9.9K	10.3K
	Vocabulary	2.8K	2.4K
	OOVs with full vocabulary	6 (0.1%)	1 (0.0%)
	OOVs with 30K shortlist (Rate)	276 (2.8%)	50 (0.5%)

Experiment Setup

System configuration:

- 30000 **most frequent words as source and target vocabulary**
- **Out-of-vocabulary words are mapped to unknown tokens**
- **Bi-directional encoder with** 1000 **GRU nodes each**
- **GRU based decoder with** 1000 **nodes**
- **Alignment computation also has an internal dimension of** 1000

Training:

- **IWLST2013:** 500000 **iterations and in-domain data included twice**
- **WMT2016:** 300000 **iterations**
- **Europarl:** 250000 **iterations**
- **Evaluation after each** 10000 **iterations on corresponding dev set**

Analysing Attention-based Alignments

- How good is the alignment quality of attention-based NMT?
- How can we evaluate attention-based alignments?
- How important are attention-based alignments for translation?

Baseline Results: Europarl

Europarl De-En	alignment-test			
Model	BLEU%	TER%	AER%	SAER %
GIZA++	-	-	22.7	28.2
Attention-Based	28.2	57.7	38.1	63.6

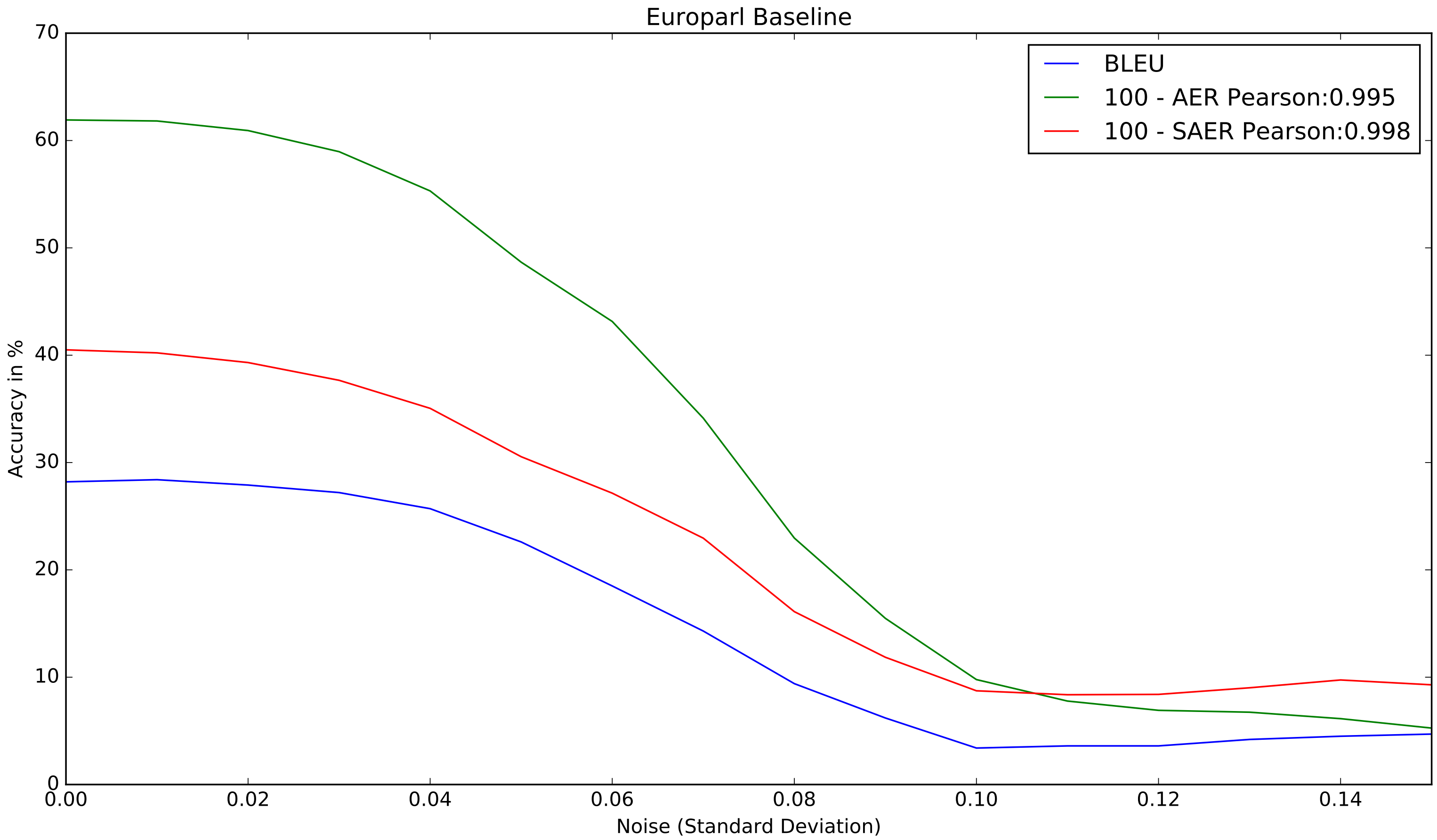
- **Alignment Evaluation:**

$$\text{AER}(S, P; A) = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}$$ **[Och & Ney 03]**

$$\text{SAER}(M_S, M_P; M_A) = 1 - \frac{|M_A \odot M_S| + |M_A \odot M_P|}{|M_A| + |M_S|}$$ **[Tu & Lu+ 16]**

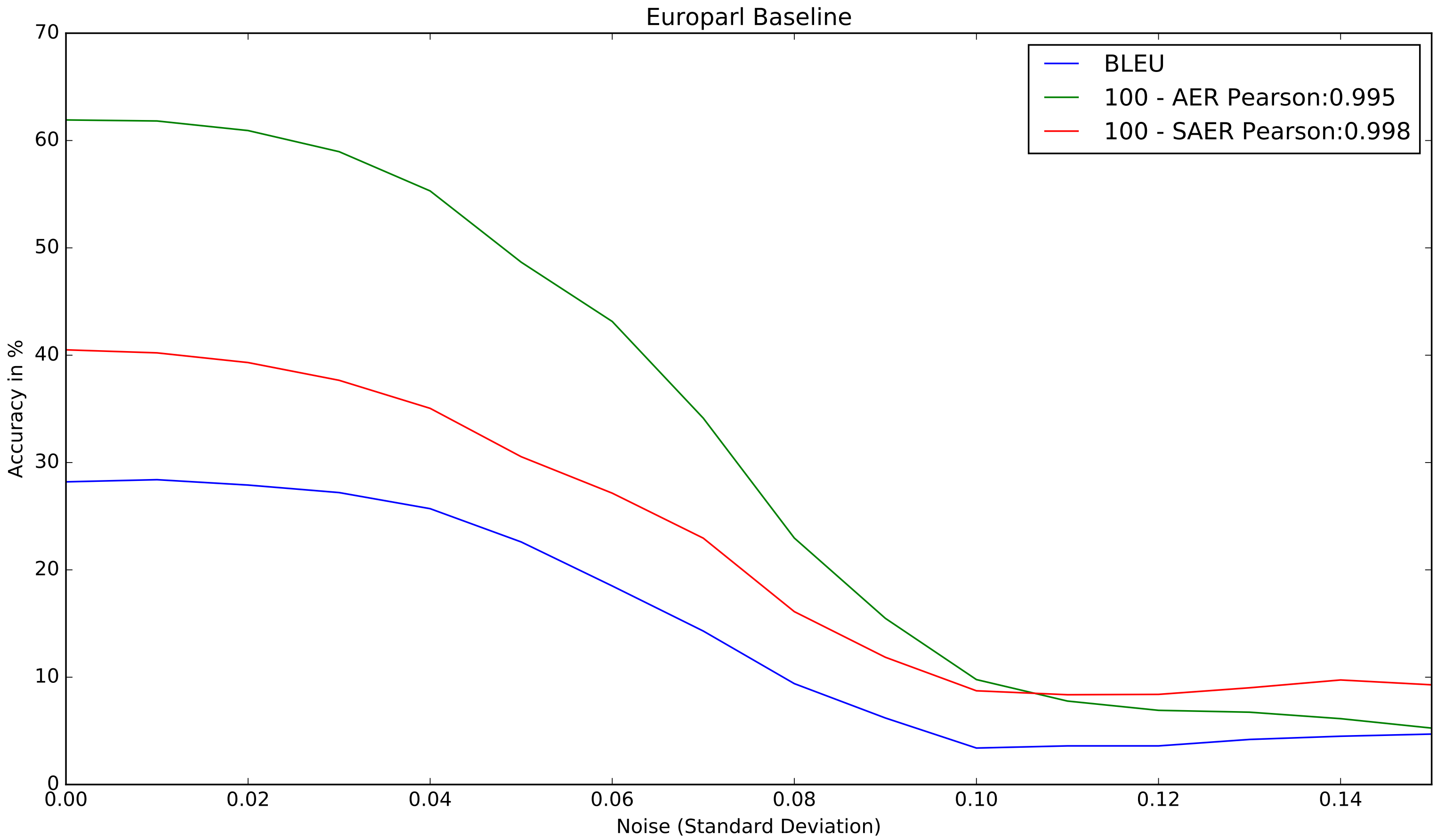
Analysing Attention-based Alignments (Europarl)

- **Compare BLEU to AER, SAER on model with increasing noise on alignment parameters**



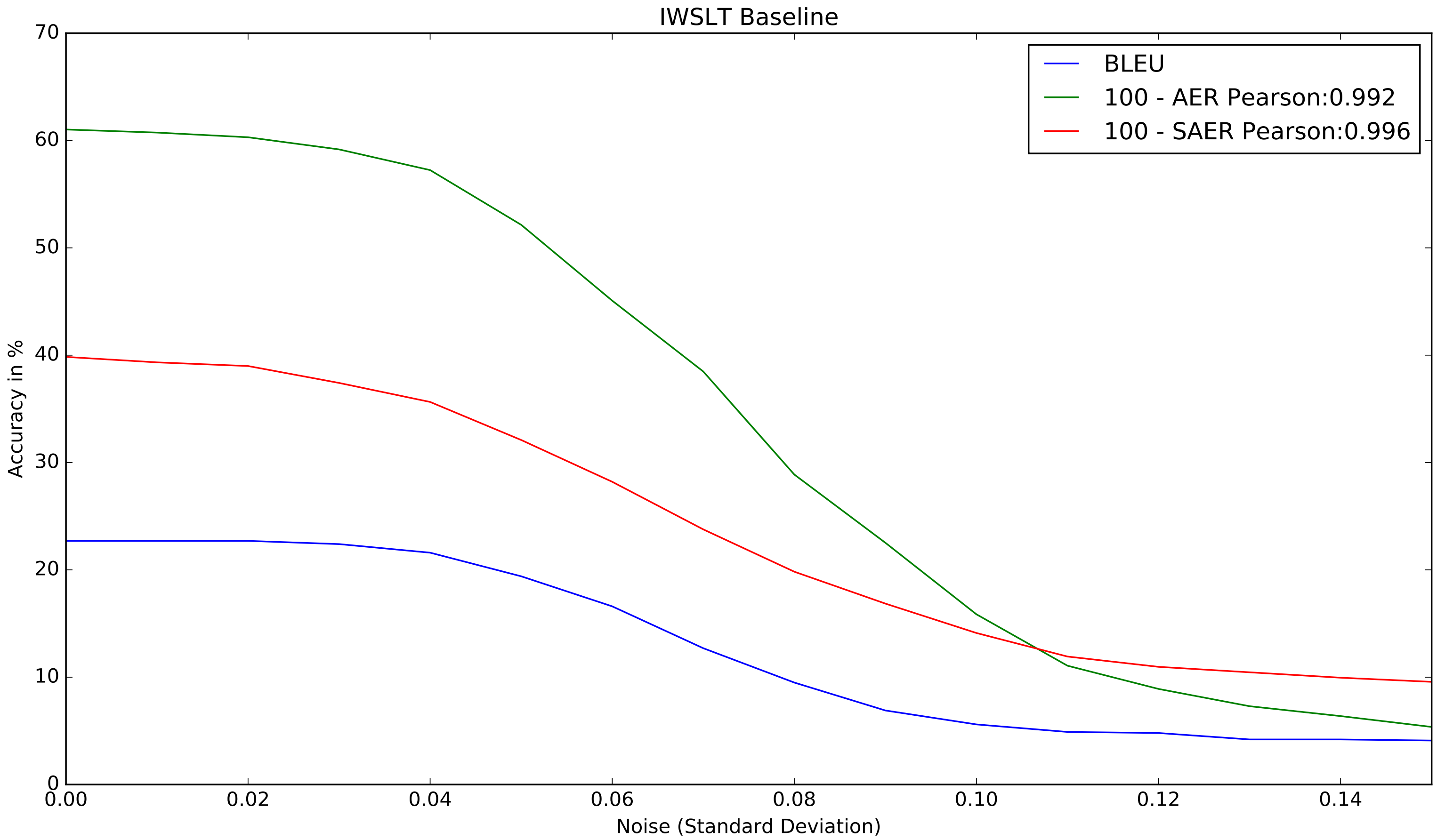
Analysing Attention-based Alignments (Europarl)

- Attention parameters are robust to noise up to a certain degree
- Alignment quality correlates with translation quality for all evaluation methods



Analysing Attention-based Alignments (IWSLT2013)

- Attention parameters are robust to noise up to a certain degree
- Alignment quality correlates with translation quality for all evaluation methods



Outline

Alignment Feedback

- Standard attention-mechanism: past alignment information disregarded
- Linguistic coverage [Tu & Lu$^+$ 16]:
  - sum of past alignments
- Neural network based coverage [Tu & Lu$^+$ 16, Mi & Wang$^+$ 16]:
  - separate RNN to compute coverage vector of past alignments
- Include feedback vector γ_{ij} (similar to coverage vector):

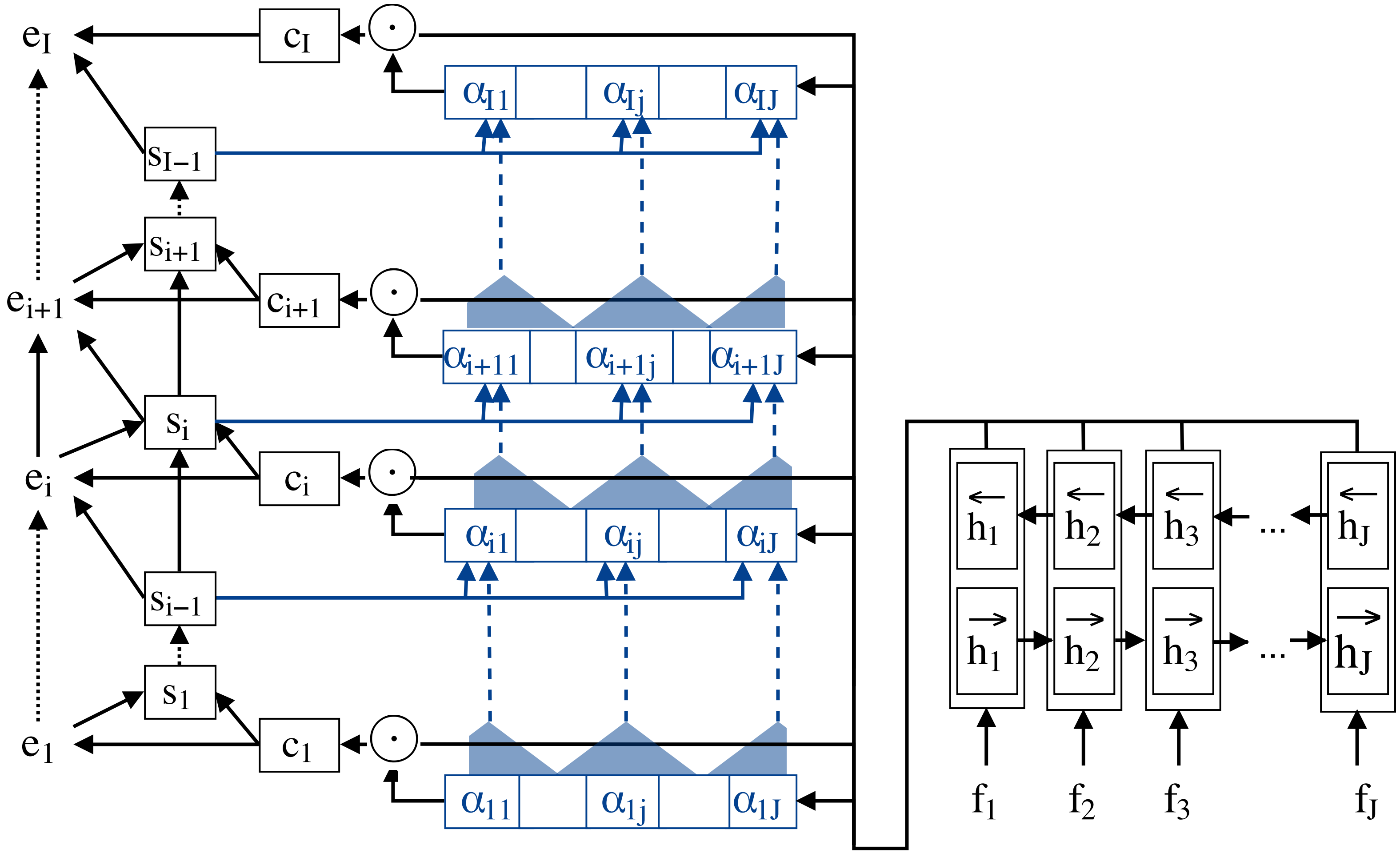
$$\tilde{\alpha}_{i+1j} = v_a^T \tanh(W_a s_i + U_a h_j + V_a \gamma_{ij})$$

- Compute γ_{ij} as weighted combination of prior alignments α_{i1}^J
- Problem: source sentence length J varies
- Solution: use shared weights $\rightarrow$ RNN and CNN

Convolutional Feedback

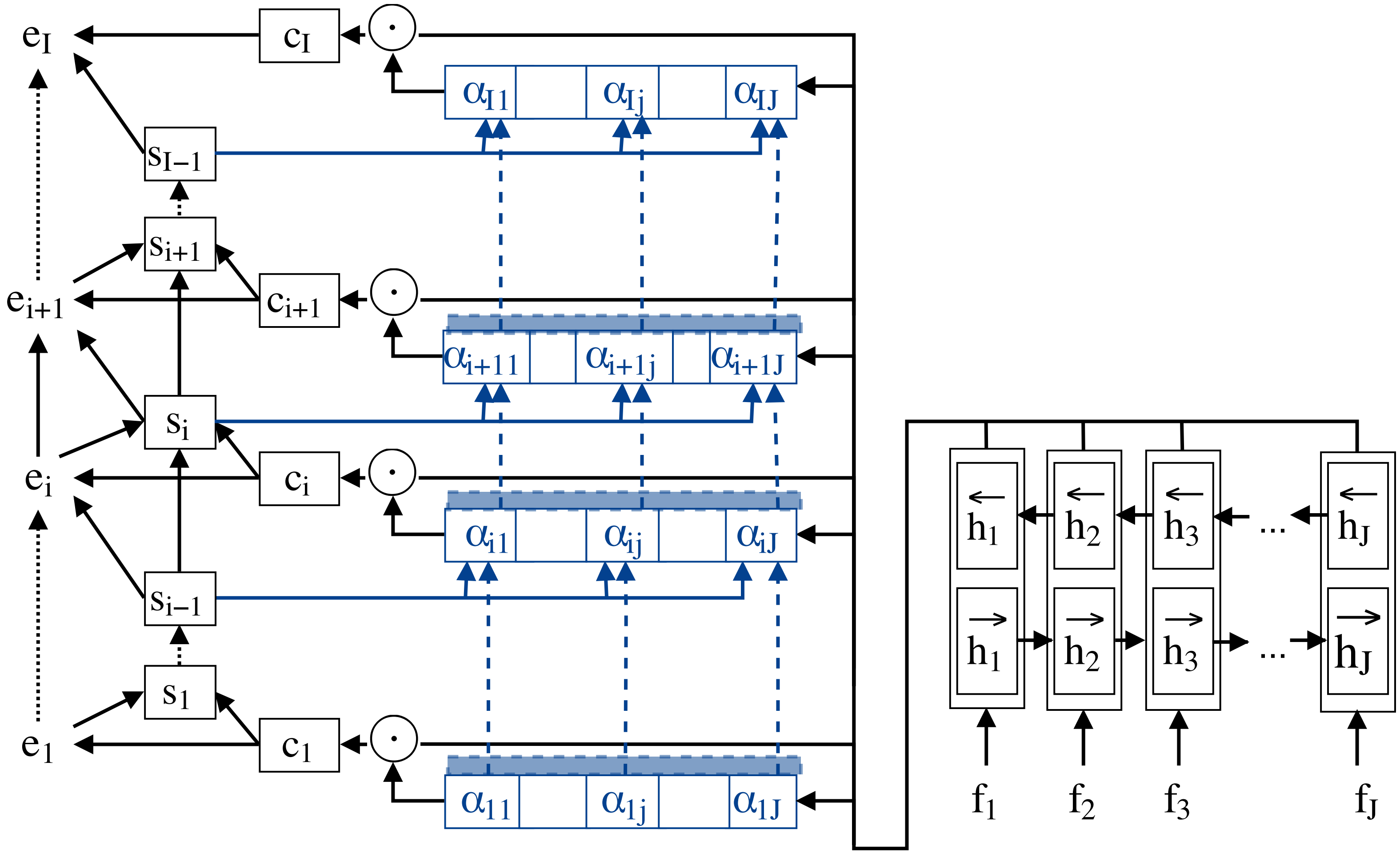
- **Introduced by [Chorowski & Bahdanau+ 15] for ASR to encourage monotonicity by convolving M filters of size D**

$$\gamma_i = G * \alpha_i \qquad G \in \mathbb{R}^{M \times D}$$



Bidirectional RNN Feedback

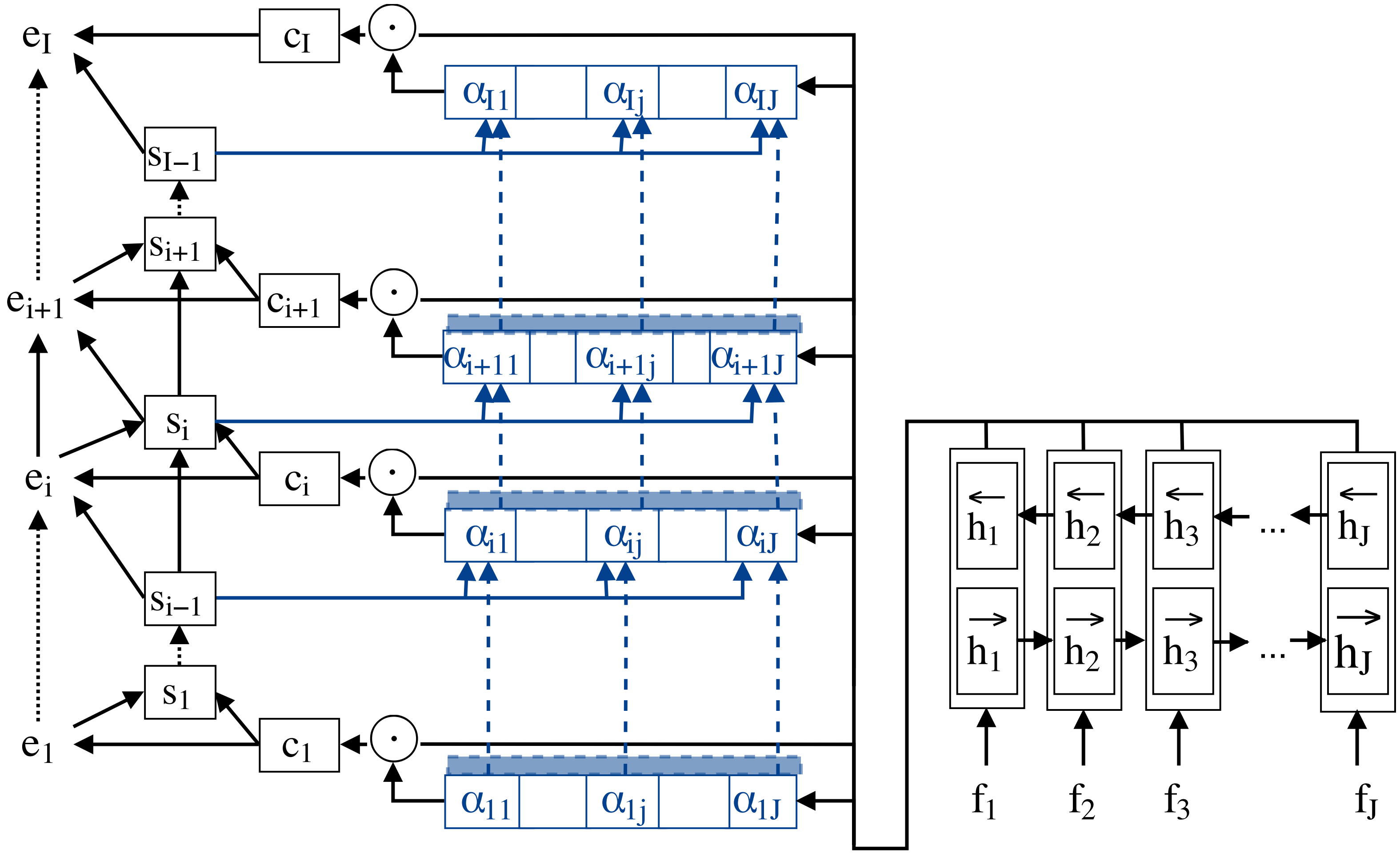
$$\overrightarrow{\gamma}_{ij} = \overrightarrow{g_{\text{rec}}}(\alpha_{ij}; \overrightarrow{\gamma}_{ij-1})$$
$$\overleftarrow{\gamma}_{ij} = \overleftarrow{g_{\text{rec}}}(\alpha_{ij}; \overleftarrow{\gamma}_{ij+1})$$



Bidirectional RNN Feedback

- **Use bidirectional RNN over past attention weights to compute γ_{ij}**

$$\gamma_{ij} = [\overrightarrow{\gamma}_{ij}^T; \overleftarrow{\gamma}_{ij}^T]^T$$



Results: Alignment Feedback (IWSLT2013)

IWSLT De-En	dev		test		eval11		alignment-test	
Model	BLEU%	TER%	BLEU%	TER%	BLEU%	TER%	AER%	SAER%
Attention-Based	30.5	48.7	29.3	50.6	33.9	46.6	41.8	66.3
+ conv ($D = 5, M = 5$)	30.8	49.0	29.5	50.2	34.3	45.8	41.3	66.6
+ conv ($D = 20, M = 1$)	31.4	49.6	29.5	50.7	33.2	47.1	41.8	67.6
+ bid-feedback	29.6	48.5	28.2	49.8	33.1	44.5	41.9	65.6

- No significant improvement in alignment quality
- Bid-feedback not successfull
- Small improvements for convolutional feedback ($D = 5, M = 5$)

Results: Alignment Feedback (WMT2016)

WMT En-Ro	newsdev2016/1		newsdev2016/2		newstest2016	
Model	BLEU%	TER%	BLEU%	TER%	BLEU%	TER%
Attention-Based	19.8	62.0	21.3	58.1	20.3	60.4
+ conv ($D = 5, M = 5$)	21.4	61.5	23.8	56.6	22.0	60.2
+ conv ($D = 20, M = 1$)	21.0	61.4	23.5	56.9	21.4	60.1
+ bid-feedback	19.2	64.4	21.7	60	19.9	63.5

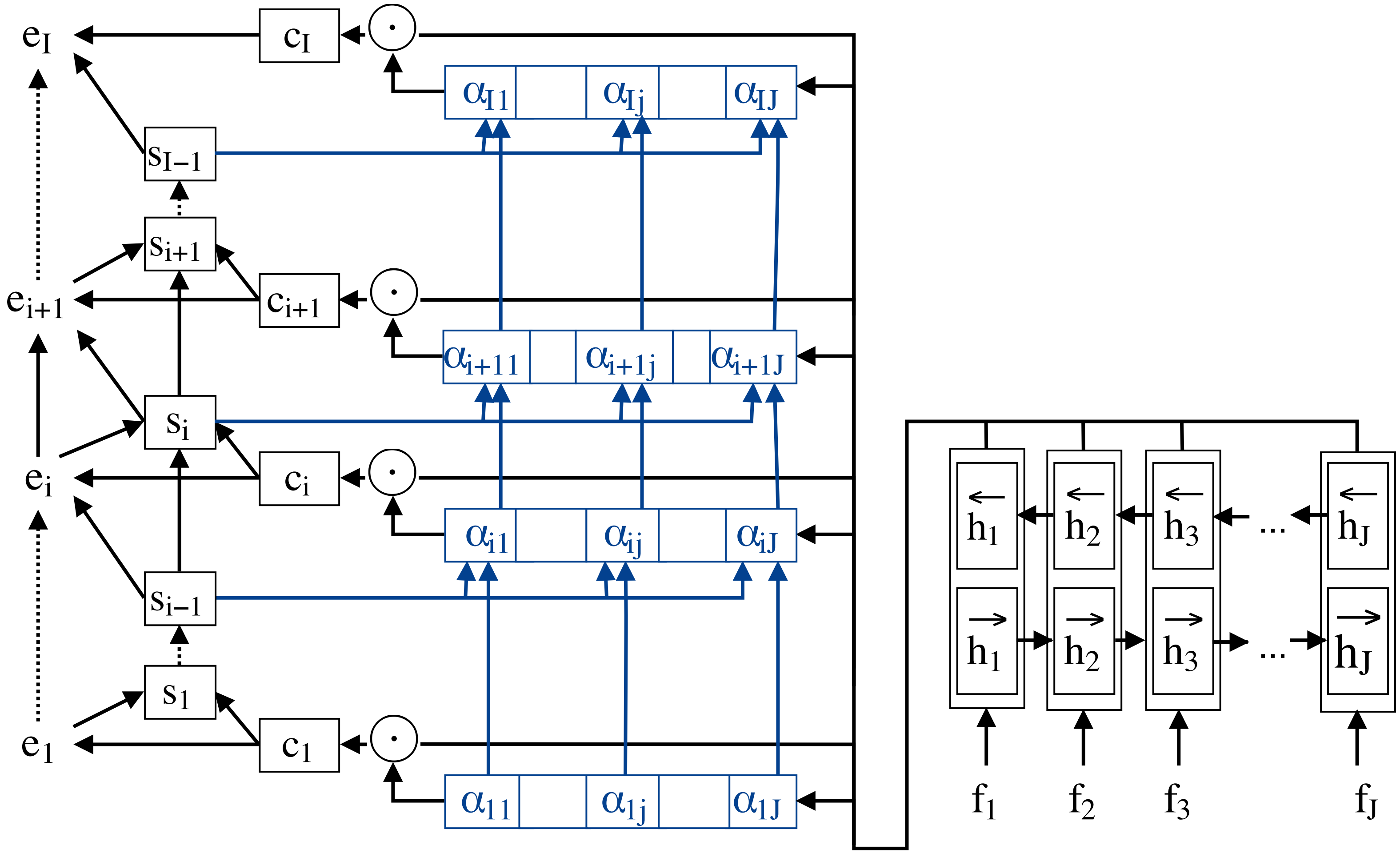
- Bid-feedback not successfull
- Large improvements of up to 2.5 BLEU for convolutional feedback ($D = 5, M = 5$ and $D = 20, M = 1$) on WMT2016

Outline

Recurrent Attention

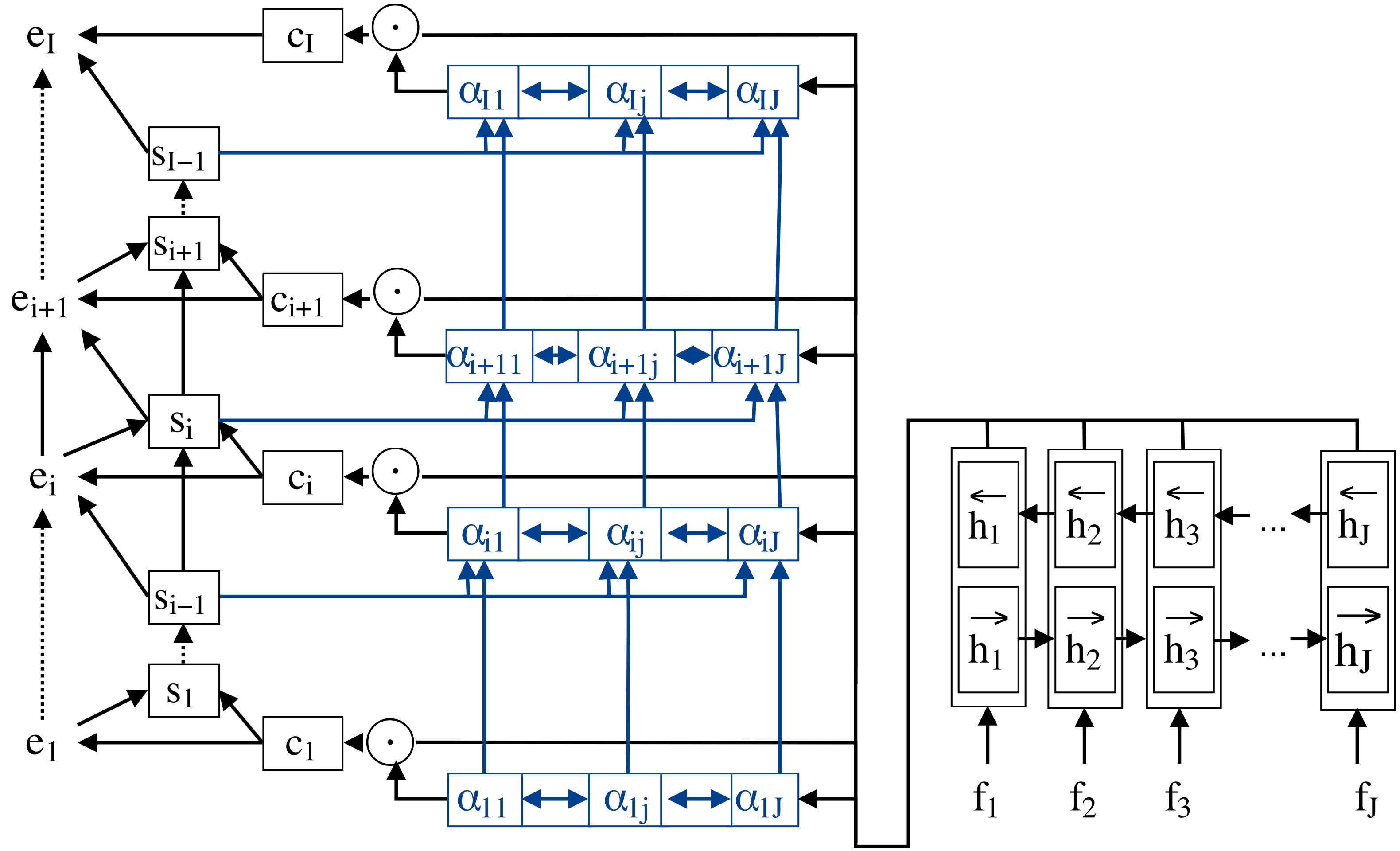
- Include context $\tilde{\alpha}_{1j}, \ldots, \tilde{\alpha}_{ij}$ by computing alignments through an RNN

$$\tilde{\alpha}_{i+1j} = v_a^T \cdot g_{\text{rec}}(s_i, h_j; \tilde{\alpha}_{ij})$$



Multidimensional Attention

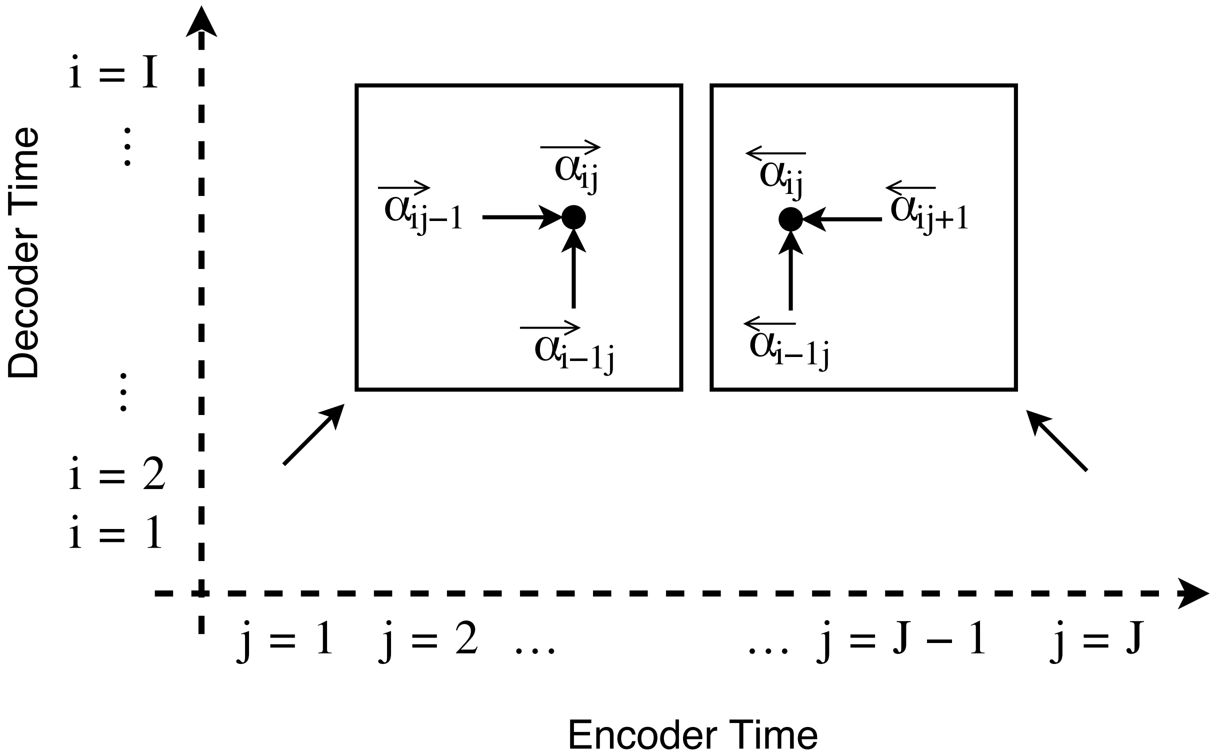
- Use context over encoder and decoder time
- Including context $\tilde{\alpha}_{i1}, \ldots, \tilde{\alpha}_{ij-1}, \tilde{\alpha}_{ij+1}, \ldots \tilde{\alpha}_{iJ}$ introduces interdependence between alignments



Multidimensional Attention

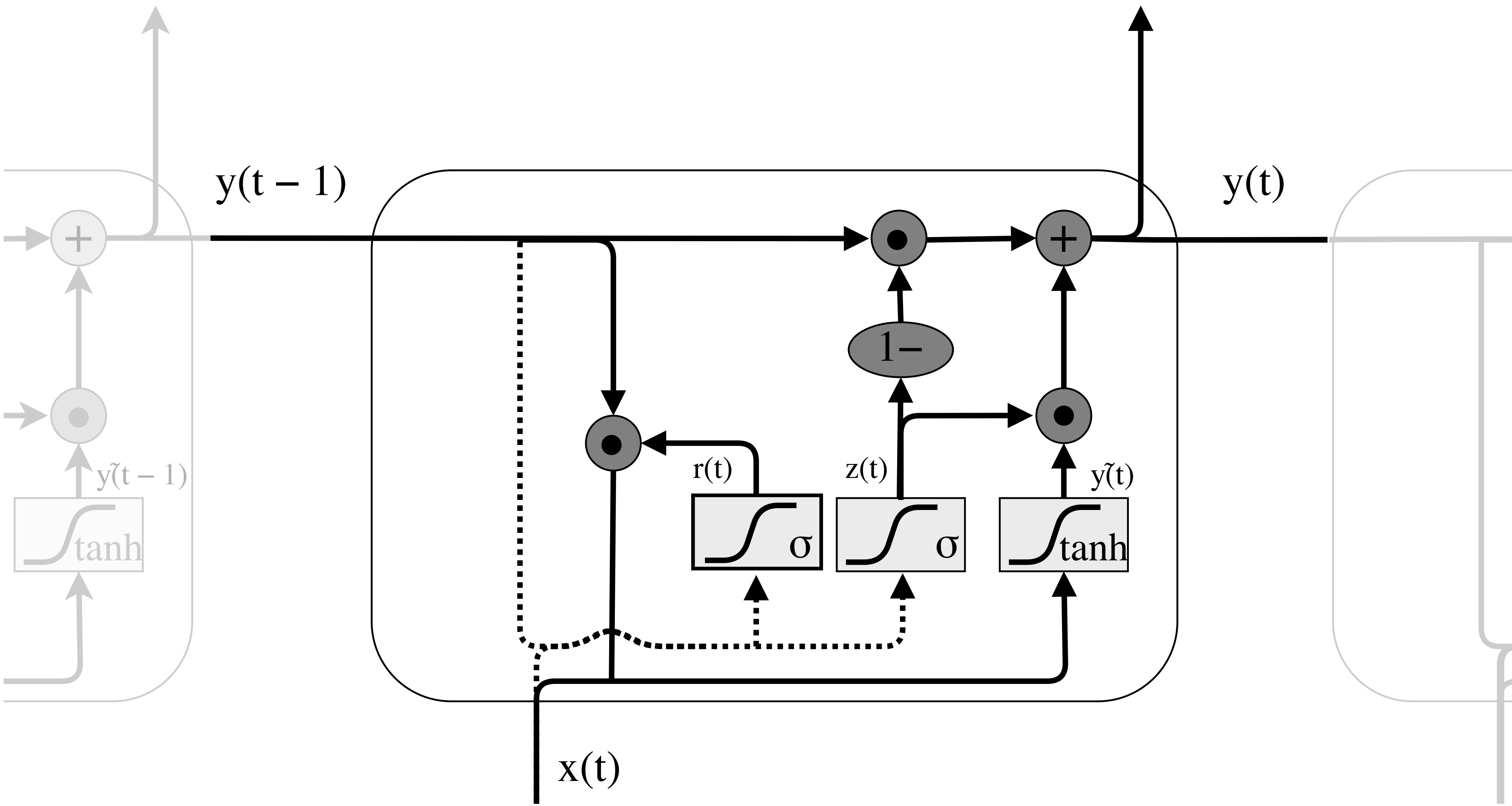
- **Use Multidimensional RNN [Graves & Fernández^{+} 07]**

$$\tilde{\alpha}_{ij} = v_a^T \cdot [\overrightarrow{\alpha_{ij}}^T; \overleftarrow{\alpha_{ij}}^T]^T$$

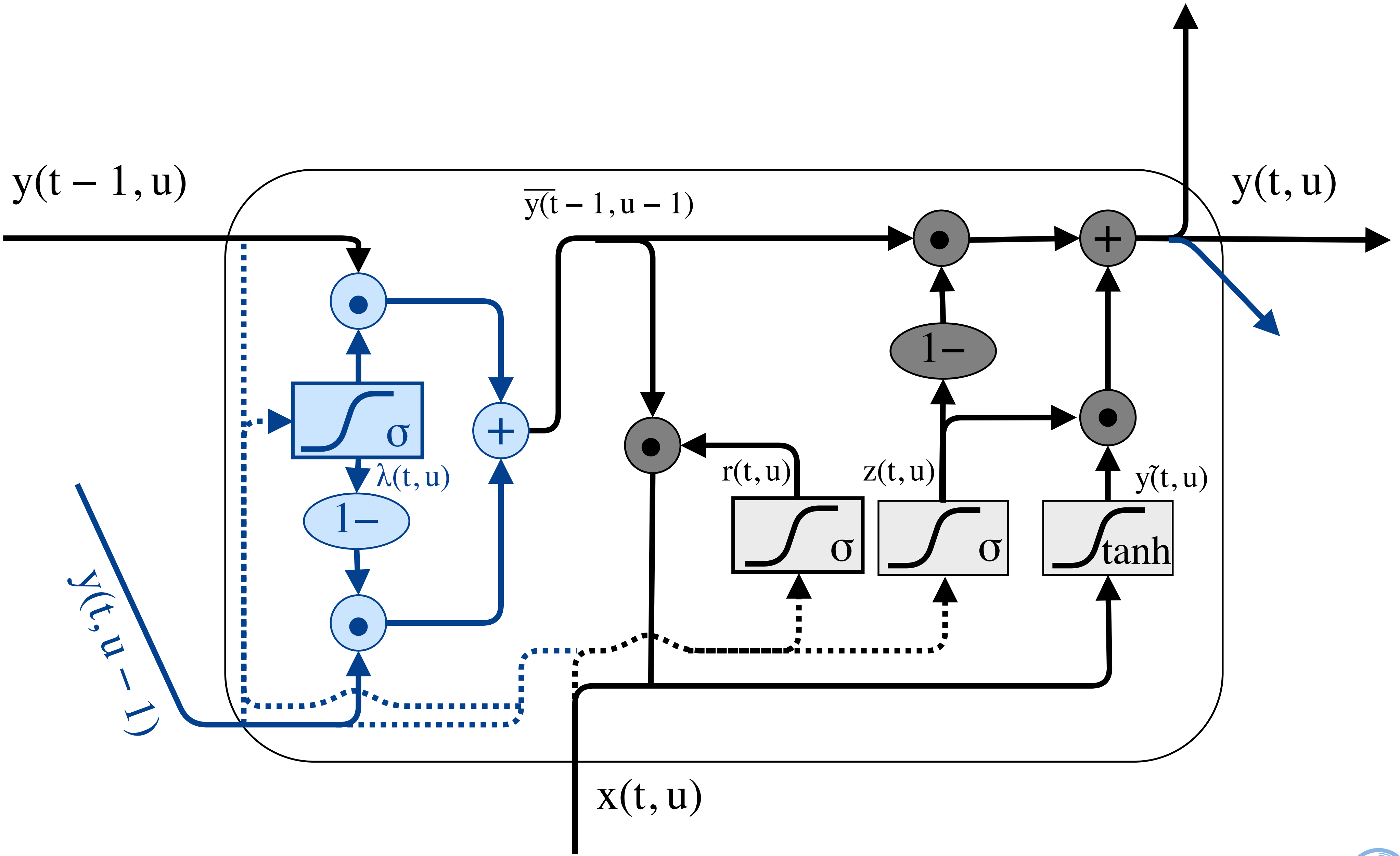


Multidimensional GRU

Standard one-dimensional GRU



Multidimensional GRU



Results: Recurrent Attention (IWSLT2013)

IWSLT De-En	dev		test		eval11		alignment-test	
Model	BLEU%	TER%	BLEU%	TER%	BLEU%	TER%	AER%	SAER%
Attention-Based	30.5	48.7	29.3	50.6	33.9	46.6	41.8	66.3
+ LSTM attention	30.8	48.0	29.2	49.9	33.3	45.6	33.5	68.1
+ bid-feedback	30.6	49.2	29.7	49.9	33.1	47.3	33.4	67.2
+ MDGRU attention	*27.5*	*51.9*	*26.0*	*52.3*	*29.6*	*48.2*	*36.7*	*70.3*

- MD-Attention takes 8 times as long as baseline to train one epoch
- MD-Attention results reported after 500000 iterations (< 2 epochs)
- No improvement on IWSLT

Results: Recurrent Attention (WMT2016)

WMT En-Ro	newsdev2016/1		newsdev2016/2		newstest2016	
Model	BLEU%	TER%	BLEU%	TER%	BLEU%	TER%
Attention-Based	19.8	62.0	21.3	58.1	20.3	60.4
+ LSTM attention	20.7	60.8	22.0	57.2	21.0	60.2
+ conv ($D = 5$,$M = 5$)	21.1	61.4	23.7	56.6	21.7	60.0
+ conv ($D = 20$,$M = 1$)	21.2	61.1	23.3	56.6	21.7	60.3
+ bid-feedback	20.5	61.5	22.8	57.1	21.3	60.1
+ MDGRU attention	20.1	61.0	22.7	56.8	20.4	60.1

- LSTM-Attention improves only on WMT by an average of 0.8 BLEU
  - adding bidirectional alignment feedback: additional 0.3 BLEU
  - combining with convolutional feedback did not improve
- MDGRU-Attention improves only on WMT by an average of 0.6 BLEU

Outline

Guided Alignment Training [Chen & Matusov^{+} 16]

- **Problem: Attention-based alignments are much worse compared to statistical alignments like GIZA-alignments [Och & Ney 03]**
- **Idea: Introducing target alignment A as a second objective**
- **Cross-Entropy cost $\mathcal{L}_{\text{align}}$ between the attention weights α and target alignment A**

$$\mathcal{L}_{\text{align}}(A, \alpha) := -\frac{1}{N} \sum_{n} \sum_{i=1}^{I_n} \sum_{j=1}^{J_n} A_{n,ij} \log \alpha_{n,ij}$$

- **Optimize w.r.t. $\mathcal{L}(A, \alpha, e_1^I, f_1^J) := \lambda_{\text{CE}} \cdot \mathcal{L}_{\text{CE}} + \lambda_{\text{align}} \cdot \mathcal{L}_{\text{align}}$**
  - **$\mathcal{L}_{\text{CE}}$: standard decoder cost function (cross-entropy)**
  - **$\lambda_{\text{align}}, \lambda_{\text{CE}}$: weights determined through experiments**

Results: Guided Alignment Training (IWSLT2013)

IWSLT De-En	dev		test		eval11		alignment-test	
Model	BLEU%	TER%	BLEU%	TER%	BLEU%	TER%	AER%	SAER%
Attention-Based	30.5	48.7	29.3	50.6	33.9	46.6	41.8	66.3
+ GA	31.5	47.2	30.3	49.0	34.3	44.3	35.4	44.2

- Improves translation by an average of 0.8 BLEU on IWSLT2013
- Great improvement in AER and SAER

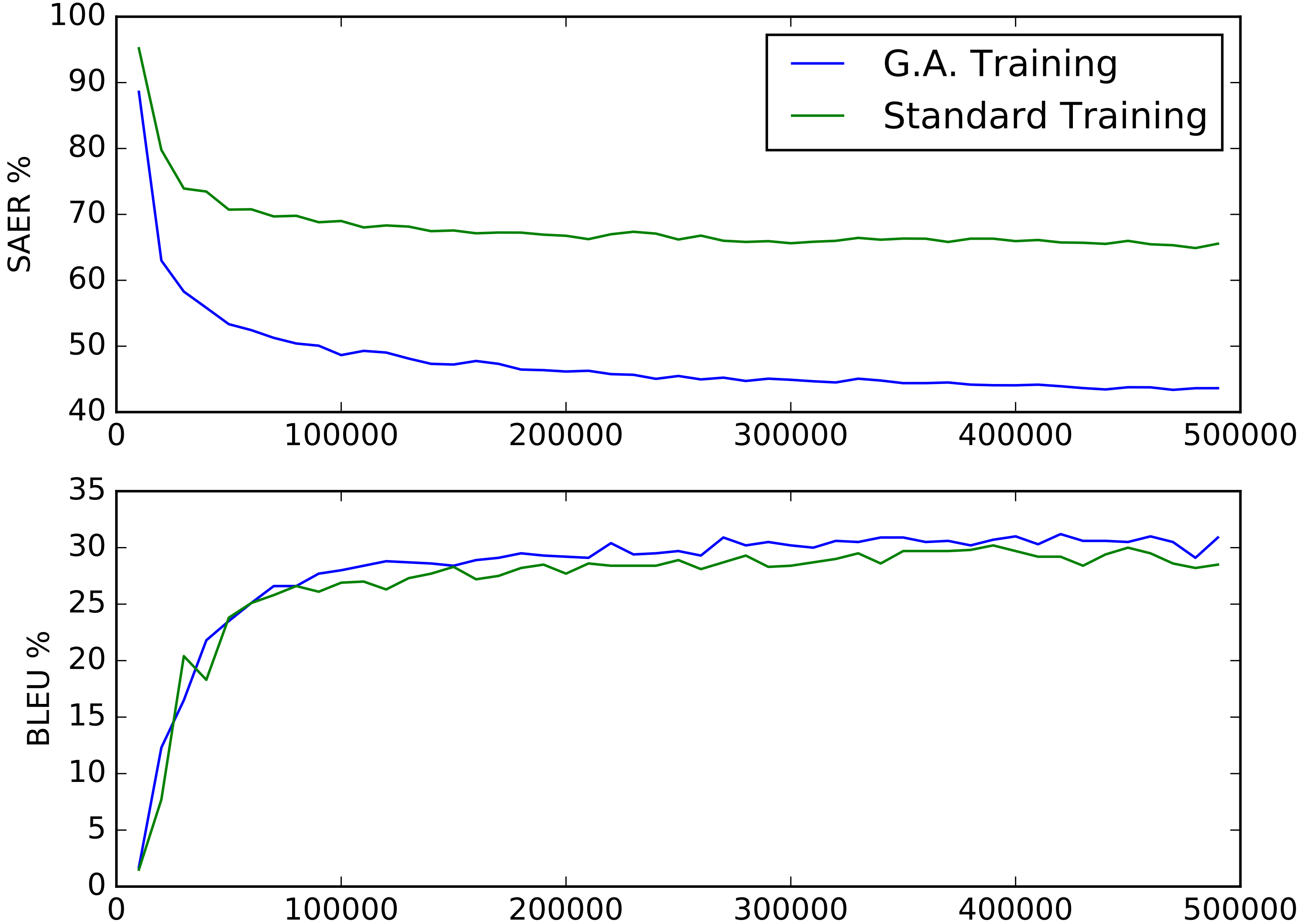
Results: Guided Alignment Training (WMT2016)

WMT En-Ro	newsdev2016/1		newsdev2016/2		newstest2016	
Model	BLEU%	TER%	BLEU%	TER%	BLEU%	TER%
Attention-Based	19.8	62.0	21.3	58.1	20.3	60.4
+GA	21.0	61.1	23.6	56.4	21.8	59.4
+GA + conv ($D = 10$,$M = 1$)	21.4	60.1	24.7	55.4	22.3	58.7

- Improves translation by an average of 0.8 BLEU on IWSLT2013
- Great improvement in AER and SAER
- Improves translation by an average of 1.7 BLEU on WMT2016
  - Adding convolutional feedback gives an additional 0.6 BLEU on average

Guided Alignment Training vs. Standard Training (IWSLT2013)

- Guided alignment training results in better and more stable in training
- Problem: Still relying on GIZA++ to generate alignments



Outline

Introduction

Introduction to Neural Networks

Neural Machine Translation

Alignment Feedback

Recurrent Attention

Guided Alignment Training

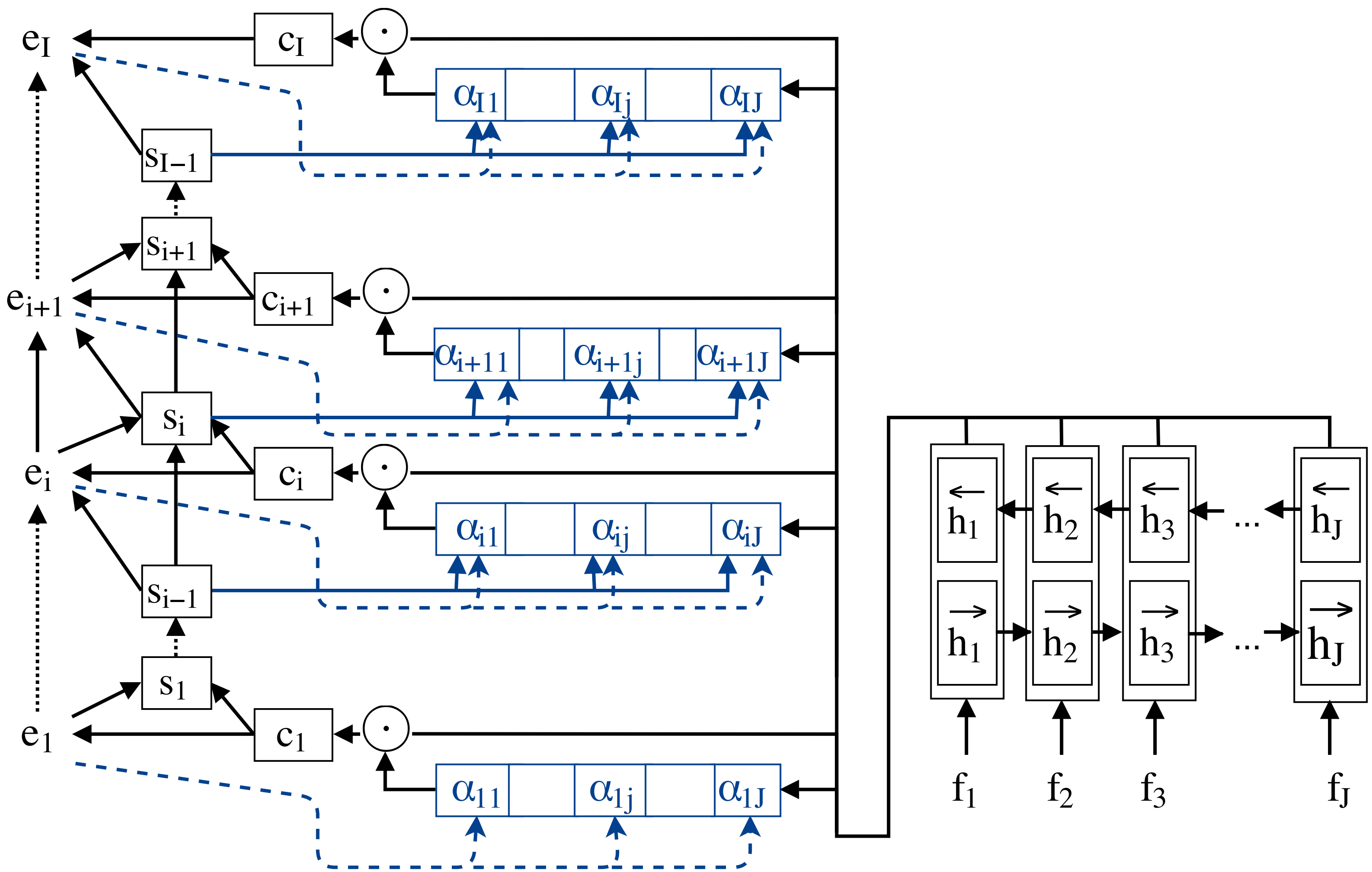
Alignment Foresight

Conclusion and Outlook

Alignment Foresight

- **Idea: Use knowledge of the target sentence e_1^I to improve the attention**

$$\tilde{\alpha}_{ij} = v_a^T \tanh(W_a s_{i-1} + U_a h_j + V_a \tilde{e}_i) \qquad V_a \in \mathbb{R}^{n \times p}$$



Alignment Foresight

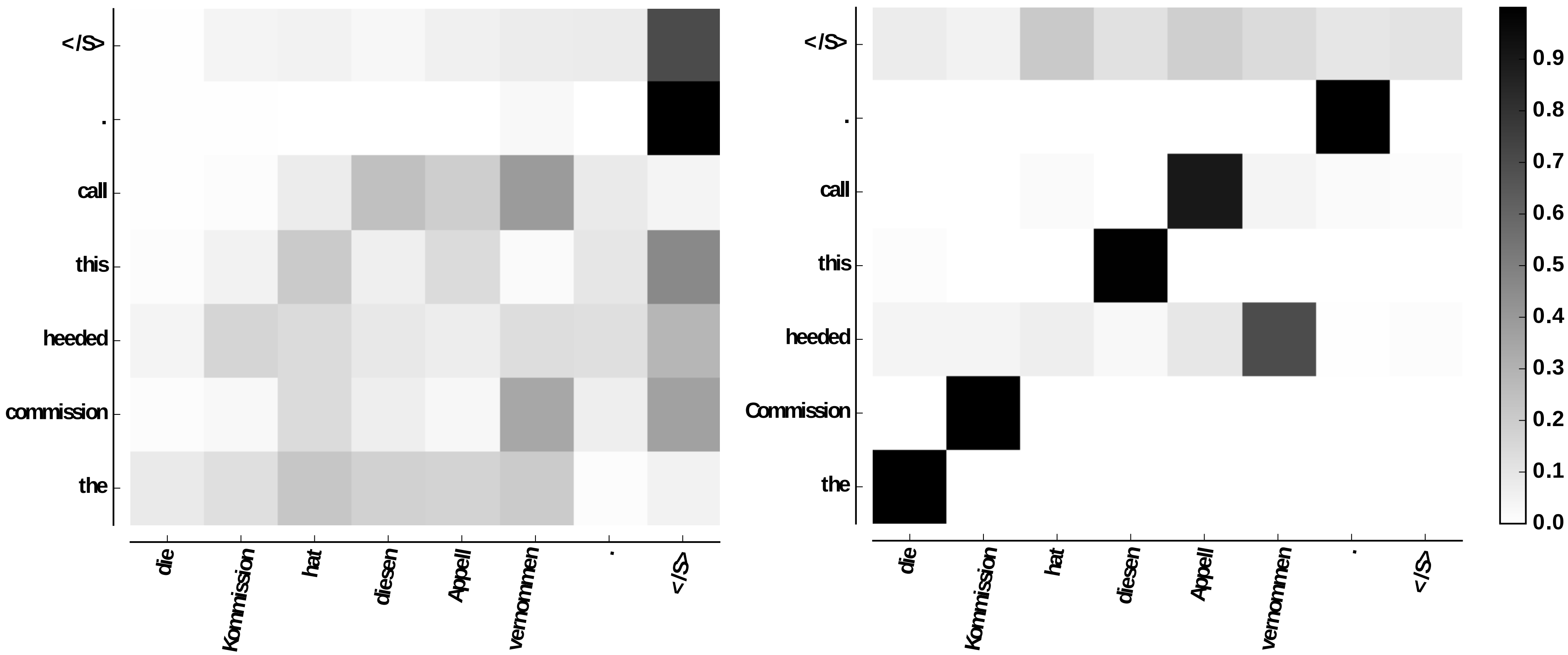
Problem in practice:

- **The network learns to encode the target word in the attention weights**

Solution:

- **Start by training the attention using *guided alignment training***

Alignment Foresight + Noise | **Alignment Foresight + GA**



Results: Alignmentment Foresight (Europarl)

Europarl De-En	alignment-test			
Model	BLEU%	TER%	AER%	SAER %
GIZA++	-	-	22.7	28.2
Attention-Based	28.2	57.7	38.1	63.6
+ AF + GA $\lambda_{\text{align}} = 5$, $\lambda_{\text{CE}} = 1$	*82.3*	*8.6*	20.0	32.6
+ AF + GA $\lambda_{\text{align}} = 1$, $\lambda_{\text{CE}} = 0.001, \dots, 1.0$	*87.2*	*5.9*	19.0	34.9
+ hard $j \rightarrow i$	-	-	20.6	25.9
+ hard $j \leftarrow i$	-	-	23.6	29.0
+ hard merged $j \rightarrow i$, $j \leftarrow i$	-	-	19.0	24.6
+ GA (GIZA++)	28.7	57.3	29.8	38.0
+ GA (AF-alignment)	28.3	57.5	28.5	36.7

Note: Aligment foresight models use knowledge of target word in translation! BLEU and TER are not valid for comparison to standard models!

Outline

Conclusions

Alignment Analysis:

- Attention-based alignment is important for NMT
- AER and SAER are meaningful for attention-based alignments
- NMT models can outperform GIZA-alignments in AER and SAER

Advanced Attention Methods:

- MD-Attention is too complex to learn on large data sets
- Convolutinal alignment feedback improves translation
- Guided alignment training stabilizes training, improves translation and alignment quality

Outlook

- Find a way to use alignment foresight without GIZA++
- Extend convolutional alignment feedback to two-dimensional convolution over decoder time i
- Dependencies of MD-attention should help:
  - Try to make learning easier for MD-attention (GA-Training,...)
  - If this is successful: efficient implementation in CUDA

Thank you for your attention!

Arne Nix

arne.nix@rwth-aachen.de

Multidimensional GRU - Formulas

- **Reset Gate:**

$$r(t,u) = \sigma_{\text{sigmoid}}(W_{xr}x(t,u) + W_{yr}y(t-1,u) + U_{yr}y(t,u-1) + b_r)$$

- **Update Gate:**

$$z(t,u) = \sigma_{\text{sigmoid}}(W_{xz}x(t,u) + W_{yz}y(t-1,u) + U_{yz}y(t,u-1) + b_z)$$

- **λ Gate:**

$$\lambda(t,u) = \sigma_{\text{sigmoid}}(W_{x\lambda}x(t,u) + W_{y\lambda}y(t-1,u) + U_{y\lambda}y(t,u-1) + b_\lambda)$$

Multidimensional GRU - Formulas

- **Recurrent Information:**

$$\overline{y}(t-1,u-1) = \lambda(t,u) \odot y(t-1,u) + [1-\lambda(t,u)] \odot y(t,u-1)$$

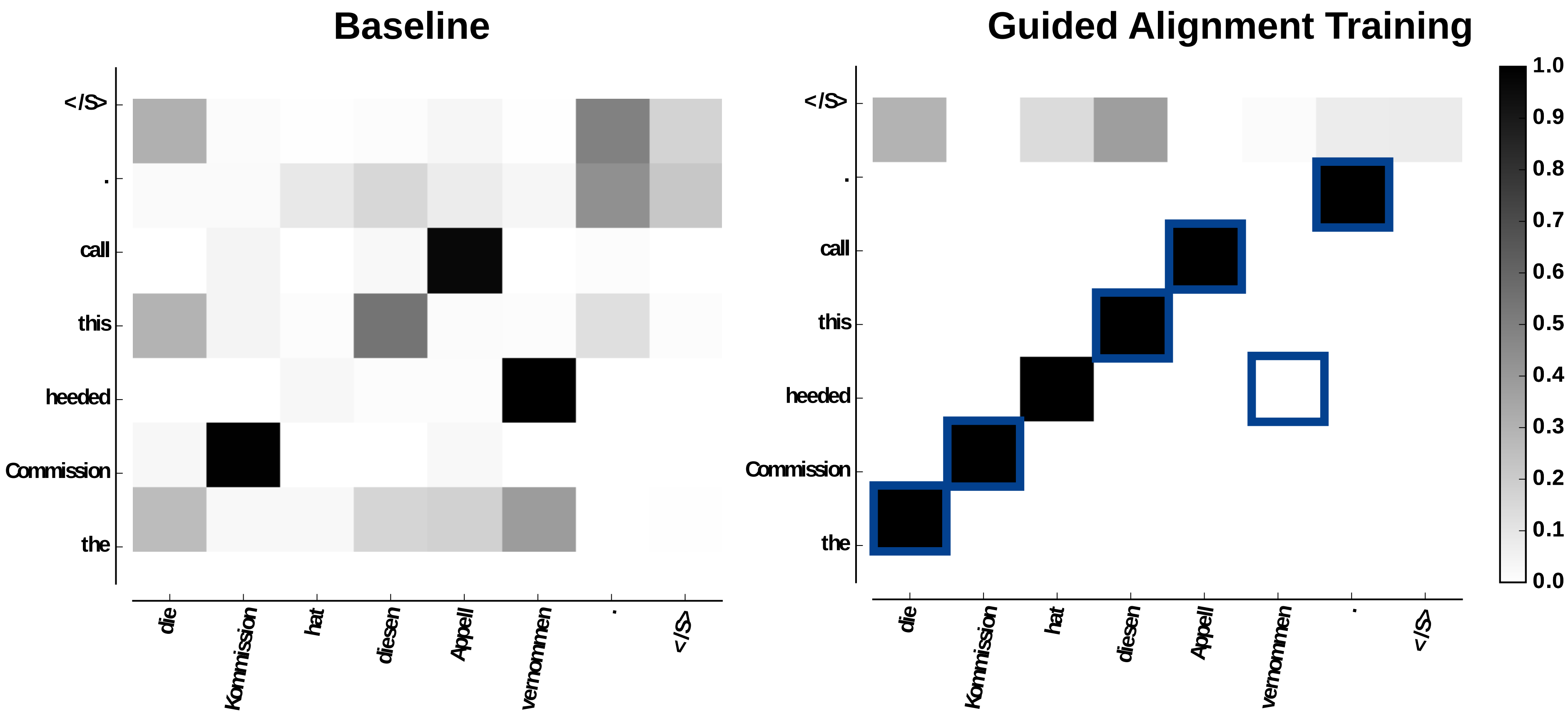
- **Update Candidate:**

$$\tilde{y}(t,u) = \sigma_{\text{tanh}}(W_{xy}x(t,u) + W_{yy}[r(t,u) \odot \overline{y}(t-1,u-1)] + b_y)$$

- **Output:**

$$y(t,u) = [1-z(t,u)] \odot \overline{y}(t-1,u-1) + z(t,u) \odot \tilde{y}(t,u)$$

Heatmaps: Baseline vs Guided Alignment (Europarl)



- **D. Bahdanau, K. Cho, Y. Bengio:**
  Neural machine translation by jointly learning to align and translate.
  Proc. *ICLR*, May 2015.
- **W. Chen, E. Matusov, S. Khadivi, J.T. Peter:**
  Guided Alignment Training for Topic-Aware Neural Machine Translation.
  Austion, Texas, October 2016. Association for Machine Translation in the Americas.
- **J.K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, Y. Bengio:**
  Attention-Based Models for Speech Recognition.
  In C. Cortes, N.D. Lawrence, D.D. Lee, M. Sugiyama, R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pp. 577–585. Curran Associates, Inc., 2015.
- **A. Graves, S. Fernández, J. Schmidhuber:**
  ***Multi-dimensional Recurrent Neural Networks*, pp. 549–558.**
  Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

- **H. Mi, Z. Wang, A. Ittycheriah:**
  Supervised Attentions for Neural Machine Translation.
  arXiv preprint arXiv:1608.00112, Vol., 2016.

- **F.J. Och, H. Ney:**
  A Systematic Comparison of Various Statistical Alignment Models.
  Computational Linguistics, Vol. 29, No. 1, pp. 19–51, 2003.

- **I. Sutskever, O. Vinyals, Q.V. Le:**
  Sequence to Sequence Learning with Neural Networks.
  Proc. *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pp. 3104–3112, 2014.

- **Z. Tu, Z. Lu, Y. Liu, X. Liu, H. Li:**
  Modeling Coverage for Neural Machine Translation.
  Proc. *54th Annual Meeting of the Association for Computational Linguistics*, August 2016.

- **B. Zhang, D. Xiong, J. Su:**
  Recurrent Neural Machine Translation.
  arXiv preprint arXiv:1607.08725, Vol., 2016.