国际人工智能会议
AAAI 2021 论文北京预讲会

Future-Guided Incremental Transformer for Simultaneous Translation

Shaolei Zhang [1,2], **Yang Feng** [1,2*], **Liangyou Li**[3]

[1] Key Laboratory of Intelligent Information Processing
Institute of Computing Technology, Chinese Academy of Sciences (ICT/CAS)
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Huawei Noah's Ark Lab
{zhangshaolei20z, fengyang}@ ict.ac.cn
liliangyou@huawei.com

张绍磊
2020.12.19

Motivation

- Simultaneous Translation (ST): starts translations synchronously while reading source sentences.
- The source sentence is incomplete and incremental at every decoding step during translating.
  - Incremental: re-calculation of the all previous hidden states at each decoding step.
  - Incomplete: trade-off between translation quality and latency.

Motivation

- Wait-k policy:
  - First waits for *k* source tokens, and then translates concurrently with the rest of source sentence.

$$g(t) = \min\{k + t - 1, |\mathbf{x}|\}, t = 1, 2, \cdots$$

  - Trained by a “prefix-to-prefix” architecture, and integrates some implicit anticipation.

$$e_{ij}^{(t)} = \begin{cases} \frac{Q(x_i)K(x_j)^T}{\sqrt{d_k}} & \text{if } i, j \leq g(t) \\ -\infty & \text{otherwise} \end{cases}$$

$$\alpha_{ij}^{(t)} = \begin{cases} \frac{\exp e_{ij}^{(t)}}{\sum_{l=1}^{n} \exp e_{il}^{(t)}} & \text{if } i, j \leq g(t) \\ 0 & \text{otherwise} \end{cases}$$

$$z_i^{(t)} = \sum_{j=1}^{n} \alpha_{ij}^{(t)} V(x_j)$$

Motivation

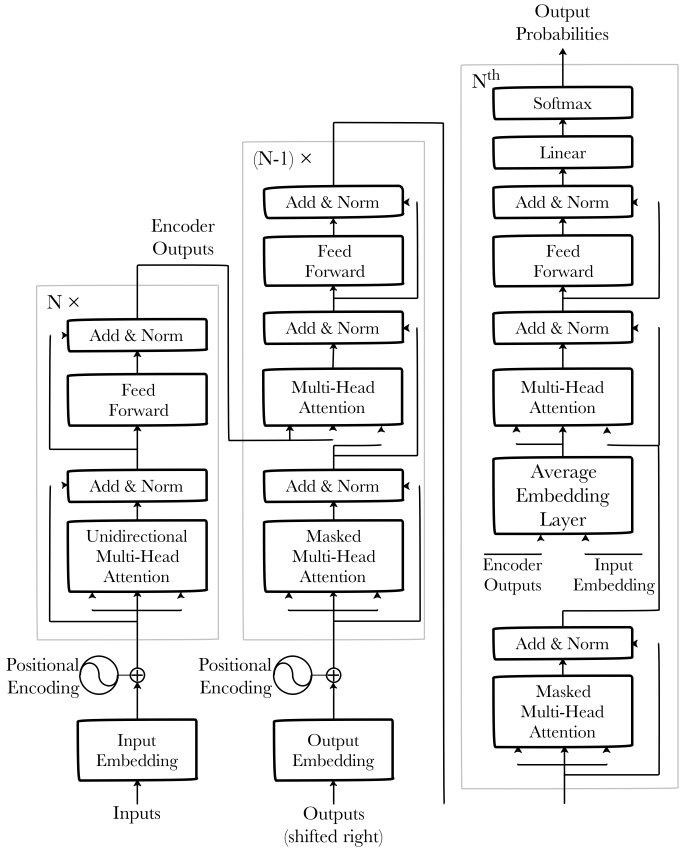
- Weakness of wait-k policy:
  - High complexity :
    - re-calculation of the all previous hidden states at each decoding step, making the computational cost increase quadratically.
    - per-layer complexity of self-attention in wait-k policy is up to $O(n^3 \cdot d)$
  - Lack of future:
    - acquisition of implicit anticipation through “prefix-to-prefix” training is data-driven, since the training data contains many prefix-pairs in the similar form.
    - inefficient and uncontrollable.

Motivation

- Avoid high complexity : incremental Transformer
  - a unidirectional encoder.
  - a decoder with an average embedding layer (AEL).
- Enhance the predictive ability: future-guided training
  - encourage the model to embed some future information.
  - simultaneously trained a conventional Transformer for full-sentence NMT as the teacher of incremental Transformer.

Method

- Incremental Transformer:
  - Unidirectional encoder (left-to-right):
    - The newly-appearing source word will not change the hidden states of the previous position.
  - Decoder with Average Embedding Layer (AEL)
    - Make up for the lack of attention to the later tokens.
    - AEL summarize the information of all consumed sources, and add it to the unidirectional hidden states.
    - Do not increase too much complexity.

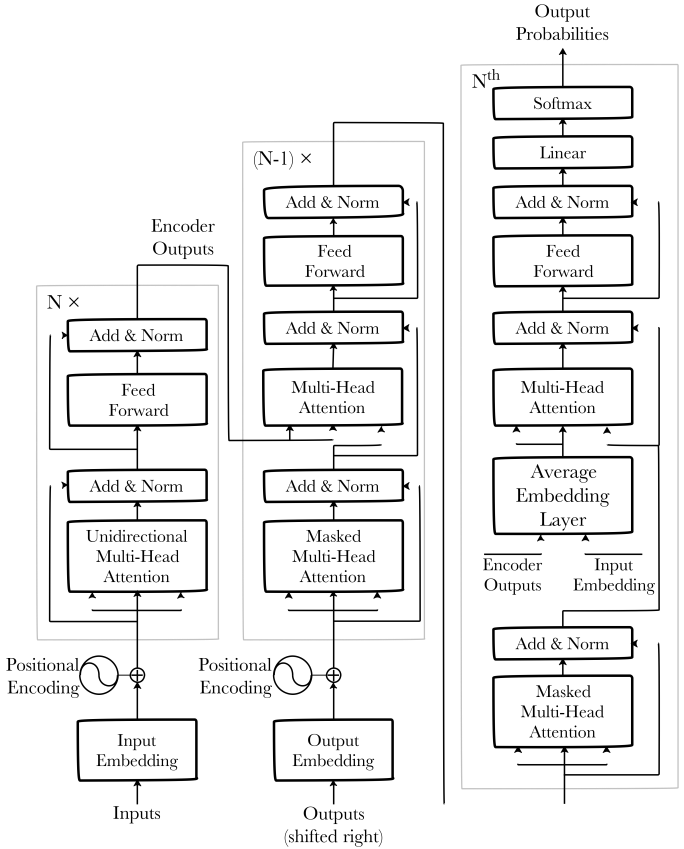


Method

- Incremental Transformer:
  - Unidirectional encoder (left-to-right)

$$e_{ij} = \begin{cases} \frac{Q(x_i)K(x_j)^T}{\sqrt{d_k}} & \text{if } j \le i \le g(t) \\ -\infty & \text{otherwise} \end{cases}$$

$$\alpha_{ij} = \begin{cases} \frac{\exp e_{ij}}{\sum_{l=1}^{n} \exp e_{il}} & \text{if } j \le i \le g(t) \\ 0 & \text{otherwise} \end{cases}$$



Method

- Incremental Transformer:
  - Decoder with Average Embedding Layer (AEL)
    - AEL performs an average operation on the input embedding:

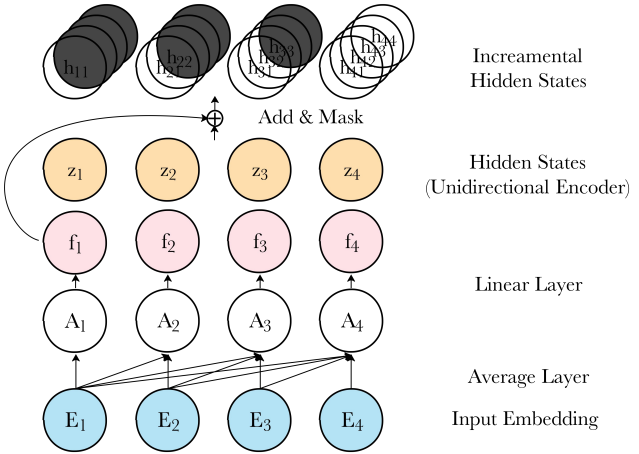
$$A_i = \frac{1}{i}\sum_{j=1}^{i} E_j$$

    - map ***A*** from the embedding space to the hidden states space:

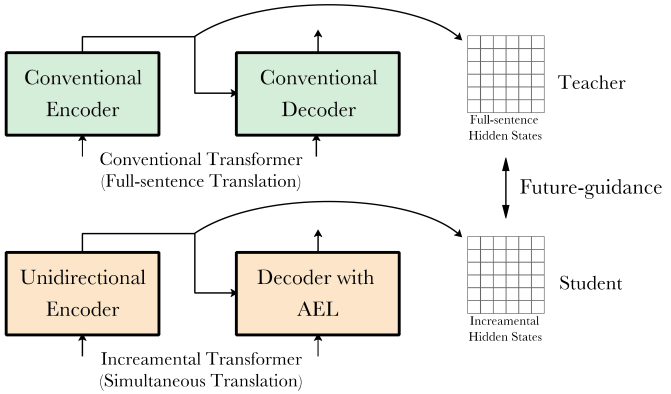
$$f_i = \mathbf{W}A_i$$

$$h_{ij} = \begin{cases} f_i + z_j & j \le i \\ \mathbf{0} & \text{otherwise} \end{cases}$$

    - ***f*** is added to the hidden states of the tokens have been read in:



Method



- Future-guided training:
  - Knowledge Distillation
    - Introduced a conventional Transformer as the teacher of the incremental Transformer, and apply L_2 regularization term between the hidden states of them:

$$\mathcal{L}\left(\mathbf{z}^{incr}, \mathbf{z}^{full}\right) = \frac{1}{n}\sum_{i=1}^{n}\left\| z_i^{incr} - z_i^{full} \right\|^2$$

    - Both incremental Transformer and conventional Transformer are trained with cross-entropy loss:

$$\mathcal{L}\left(\theta_{incr}\right) = -\sum_{(\mathbf{x},\mathbf{y}^\star)\in D} \log p_{incr}\left(\mathbf{y}^\star \mid (\mathbf{x}, \theta_{incr})\right)$$

$$\mathcal{L}\left(\theta_{full}\right) = -\sum_{(\mathbf{x},\mathbf{y}^\star)\in D} \log p_{full}\left(\mathbf{y}^\star \mid (\mathbf{x}, \theta_{full})\right)$$

    - The total loss is calcul$\mathcal{L} = \mathcal{L}\left(\theta_{incr}\right) + \mathcal{L}\left(\theta_{full}\right) + \lambda\mathcal{L}\left(\mathbf{z}^{incr}, \mathbf{z}^{full}\right)$

Experiments

- Datasets:
  - Nist Chinese → English
  - WMT15 German → English
- Systems:
  - offline model: bi-Transformer, uni-Transformer
  - wait-k policy: baseline(bi), baseline(uni)
  - +Teacher: only add a conventional Transformer as the teacher model
  - +AEL: only add average embedding layer we proposed
  - +AEL+Teacher: add both AEL and the conventional Transformer as the teacher model

Experiments

- Comparison between Joint Training and Pre-training

		Teacher	Student	
		BLEU	AL	BLEU
$k=9$	Pre-training	45.13	9.81	40.57
	Joint training	44.91	9.63	**41.86**
$k=7$	Pre-training	45.13	7.81	39.71
	Joint training	44.88	8.11	**40.73**
$k=5$	Pre-training	45.13	6.50	38.39
	Joint training	44.84	6.26	**40.00**
$k=3$	Pre-training	45.13	4.62	37.00
	Joint training	44.62	4.43	**38.28**
$k=1$	Pre-training	45.13	2.34	32.11
	Joint training	44.58	2.32	**34.20**

1. Jointly training makes the student model get better performance than pre-training.
2. The teacher model is for full-sentence MT, while the student model is for ST, and the two have inherent differences in the hidden states distribution.
3. Should not let the incremental Transformer learn from the conventional Transformer without any difference, but narrow the distance between them.

Experiments

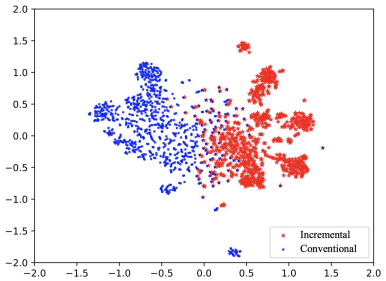
- Comparison with baseline
  - The training speed of '+AEL' is about 27.86 times.
  - the training speed of '+AEL+Teacher' is increased by about 13.67 times, and translation quality improves about 1.88 BLEU on Zh-En and 0.91 BLEU on DeEn (average on different k).

		AL	BLEU	Δ
offline	bi-Transformer	28.60	31.42	
	uni-Transformer	28.70	30.12	
$k=9$	baseline(bi)	9.36	28.48	
	baseline(uni)	9.24	28.10	
	+AEL+Teacher	9.25	**29.42**	+1.32
$k=7$	baseline(bi)	7.44	28.09	
	baseline(uni)	7.83	27.84	
	+AEL+Teacher	7.90	**28.38**	+0.54
$k=5$	baseline(bi)	5.58	26.38	
	baseline(uni)	5.78	25.73	
	+AEL+Teacher	5.74	**26.97**	+1.24
$k=3$	baseline(bi)	3.48	24.18	
	baseline(uni)	3.91	24.04	
	+AEL+Teacher	3.95	**24.39**	+0.35
$k=1$	baseline(bi)	1.60	18.48	
	baseline(uni)	1.32	18.29	
	+AEL+Teacher	1.31	**19.36**	+1.07

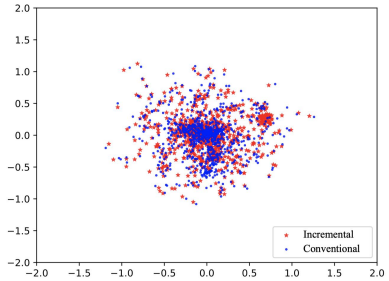
		MT03	MT04	MT05	MT06	MT08	AVERAGE		Δ	Training Time (secs/b)
				BLEU			AL	BLEU		
offline	bi-transformer	44.56	45.69	45.28	44.63	34.51	28.83	42.93		0.31
	uni-transformer	43.22	44.40	43.12	42.31	32.51	28.82	41.11		0.31
$k=9$	baseline(bi)	40.35	42.21	40.21	40.78	32.45	9.99	39.20		9.92
	baseline(uni)	39.42	42.08	40.33	40.12	31.59	9.99	38.71		0.31
	+AEL	40.77	42.27	40.11	40.77	32.17	10.09	39.22	+0.51	0.41
	+Teacher	41.52	**43.05**	**41.75**	41.59	**33.12**	9.74	40.21	+0.99	0.78
	+AEL+Teacher	**41.75**	43.03	41.63	**41.76**	33.06	9.73	**40.25**	**+1.54**	0.80
$k=7$	baseline(bi)	40.27	41.94	39.90	40.35	31.84	8.05	38.86		10.26
	baseline(uni)	38.79	41.12	38.77	39.13	30.61	8.01	37.68		0.31
	+AEL	39.81	41.66	38.81	40.14	31.16	8.17	38.32	+0.63	0.41
	+Teacher	**40.51**	41.81	**40.35**	**40.90**	32.16	8.31	39.15	+1.46	0.79
	+AEL+Teacher	40.41	**42.08**	40.29	40.44	**32.94**	8.10	**39.23**	**+1.55**	0.81
$k=5$	baseline(bi)	40.12	41.46	39.58	40.21	**31.57**	6.34	38.59		10.70
	baseline(uni)	37.09	39.62	37.78	37.66	29.82	6.27	36.39		0.31
	+AEL	38.74	40.11	38.36	39.04	30.30	6.06	37.31	+0.92	0.41
	+Teacher	39.47	40.42	38.82	39.78	30.05	6.24	37.71	+1.31	0.82
	+AEL+Teacher	**40.15**	**41.53**	**39.58**	**40.59**	31.29	5.98	**38.63**	**+2.23**	0.83
$k=3$	baseline(bi)	37.08	**39.11**	36.69	37.20	28.28	4.15	35.67		11.11
	baseline(uni)	35.94	36.98	34.64	34.80	26.48	4.42	33.77		0.31
	+AEL	37.40	38.72	36.64	36.59	28.06	4.11	35.48	+1.71	0.41
	+Teacher	37.42	38.94	37.13	37.37	**29.58**	4.53	36.09	+2.32	0.84
	+AEL+Teacher	**38.15**	38.88	**37.14**	**37.46**	28.98	4.41	**36.12**	**+2.35**	0.86
$k=1$	baseline(bi)	32.67	34.51	32.55	32.04	24.79	2.45	31.31		15.11
	baseline(uni)	31.99	33.75	31.47	31.56	23.86	2.71	30.53		0.31
	+AEL	32.97	34.41	32.37	32.04	24.16	2.29	31.19	+0.66	0.41
	+Teacher	33.95	34.51	33.07	33.17	25.14	2.35	31.97	+1.44	0.84
	+AEL+Teacher	**34.21**	**35.10**	**33.11**	**33.72**	**25.19**	2.37	**32.27**	**+1.74**	0.86

Experiments

- Impact of the Knowledge Distillation
  - L_2 regularization term successfully makes incremental Transformer learn some future information from conventional Transformer.
  - Most of the improvement brought by '+Teacher' comes from the knowledge distillation between the full-sentence / incremental encoder.



(a) without L_2 regularization



(b) with L_2 regularization

		AVG BLEU	Δ
$k = 9$	baseline(uni)	38.71	
	+uni-Teacher	39.72	+1.01
	+bi-Teacher	**40.21**	+1.50
$k = 7$	baseline(uni)	37.68	
	+uni-Teacher	38.95	+1.27
	+bi-Teacher	**39.15**	+1.46
$k = 5$	baseline(uni)	36.39	
	+uni-Teacher	37.50	+1.11
	+bi-Teacher	**37.71**	+1.32
$k = 3$	baseline(uni)	33.77	
	+uni-Teacher	36.02	+2.25
	+bi-Teacher	**36.09**	+2.32
$k = 1$	baseline(uni)	30.53	
	+uni-Teacher	**32.08**	+1.55
	+bi-Teacher	31.97	+1.44

Experiments

- Prediction Accuracy
  - Use GIZA++ to align the tokens between the generated translation and the source sentence.
  - 'Absent' represents the aligned source token has not been read in when generating the target token. The generated target token is implicitly predicted by the model.
  - 'Present' represents the aligned source token has been read in when generating the

	$k=1$		$k=3$		$k=5$		$k=7$		$k=9$	
	baseline	+Teacher	baseline	+Teacher	baseline	+Teacher	baseline	+Teacher	baseline	+Teacher
Absent	54.88	**59.82**	61.34	**63.26**	63.54	**65.38**	70.72	**71.80**	70.48	**71.57**
Present	82.47	**83.32**	84.76	**85.22**	85.33	**86.04**	85.94	**86.51**	86.25	**86.92**

Conclusion

- Future-guided incremental Transformer
  - Incremental Transformer with AEL:

    Accelerate the training speed of the wait-k policy about 28 times, meanwhile attends to all consumed source tokens.
  - future-guided training:

    Incremental Transformer successfully embeds some implicit future information and has a stronger predictive ability, without adding any latency or parameters in the inference time.

THANKS

2020.12.19