Mask-Predict: Parallel Decoding of Conditional Masked Language Models

Ghazvininejad et al.

Presenter: Yoalli Rezepka García
Ruprecht Karls Universität Heidelberg
Recent Advances in Seq2Seq
with Tsz Kin Lam

February 7, 2020

Schedule

Motivation

Motivation

OLD

- Most machine translation systems use sequential decoding strategies
- Words are predicted one by one
- Entire sequence is predicted repeatedly

NEW

- Model which generates translations in a constant number of decoding iterations
- Conditional masked language models (CMLMs): encoder-decoder architectures trained with a masked language model objective
- Decoding algorithm: *mask predict*
- Only words that were predicted with low confidence are repredicted
- CMLMs offer a trade-off between speed and performance (2 BLEU for 3x speed-up)

Related Work

Related Work: Training Masked Language Models with Translation Data

- Lample and Conneau, Cross-lingual language model pretraining (2019)[5]
  - Training a masked model on translation data (pretraining step) can improve performance on cross-lingual tasks
  - Different goal: use CMLMs for pre-training vs. to generate text
  - Ghazvininejad et al. use separate model parameters for source and target texts (encoder + decoder)
  - Input tokens are replaced with noise vs. masking tokens
- Song et al., MASS: Masked Sequence to Sequence Pre-training for Language Generation (2019)[8]
  - Separate encoder decoder parameters (same)
  - Monolingual data
  - Autoregressive masked language modeling
  - No text generation

Related Work: Parallel Decoding for Machine Translation

- Non-autoregressive neural machine translation, Gu et al. (2018)[3]
  - Transformer-based approach
  - Non-autoregressive
  - Identify multi-modality problem
- End-to-End Non-Autoregressive Neural Machine Translation with Connectionist Temporal Classification, Libovický and Helcl (2018)[7]
  - collapse repititions with Connectionist Temporal Classification training objective
- Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement, Lee et al. (2018)[6]
  - very similar to Ghazvininejad et al.
  - Iterative refinement
  - Non-autoregressive prediction is corrected via denoising autoencoder
  - Main difference: application of stochastic corruption heuristics on training data

Conditional Masked Language Models (CMLMs)

Conditional Masked Language Models (CMLMs)

Definition CMLM

Predicts a set of **target tokens $\mathbf{Y_{mask}}$** given a **source text X** and part of the **target text $\mathbf{Y_{obs}}$**.

- Assumption: tokens $\mathrm{Y_{mask}}$ are conditionally independent
- Predicts individual probabilities $P(y|X, Y_{obs})$ for each $y \epsilon Y_{mask}$
- Model is implicitly conditioned on length of target sequence $N = |Y_{mask}| + |Y_{obs}|$

Architecture

- Standard encoder-decoder transformer for machine translation (Vaswani et al., 2017[9])
- Deviation: no self-attention mask that prevents left-to-right decoders from attending on future tokens
- Decoder is bi-directional (uses both left and right contexts for prediction)

Training Objective

1. Sample number of masked tokens
2. Replace inputs of tokens Y_{mask} with a special *MASK* token
3. Optimize CMLM for cross-entropy loss over every token in Y_{mask}
4. Only compute loss for tokens in Y_{mask}

Predicting Target Sequence Length

- Traditionally: predict *EOS* (end of sentence) token
- CMLMs: must know length in advance
- Prior work: length is predicted with a fertility model (Gu et al., 2018[4])
- Here: special *LENGTH* token is added to encoder (Devlin et al, 2018[1]
- Model is trained to predict length of target sequence *N* as the *LENGTH* token's output
- Loss of *LENGTH* token is added to the cross-entropy loss

Decoding with Mask-Predict

Decoding with Mask-Predict

- Target sequence's length N
- Target sequence $(y_1, ..., y_N)$
- Probability of each token $(p_1, ..., p_N)$
- Number of iterations T
  1. Mask
  2. Predict

Decoding with Mask-Predict: Mask

1. $t = 0$: mask all tokens
2. $t > 0$: mask n tokens with lowest probability scores:

$$Y_{mask}^{(t)} = \underset{i}{argmin}(p_i, n) \tag{1}$$

$$Y_{obs}^{(t)} = Y \setminus Y_{mask}^{(t)} \tag{2}$$

3. Number of masked tokens n depends on t

$$n = N \cdot \frac{T - t}{T} \tag{3}$$

Decoding with Mask-Predict: Predict

1. Predict masked tokens $Y^{(t)}_{mask}$
2. Select prediction with highest probability for each masked token $y_i \epsilon Y^{(t)}_{mask}$
3. Update probability score:

$$y^{(t)}_i = \underset{w}{argmax} P(y_i = w | X, Y^{(t)}_{obs}) \tag{4}$$

$$p^{(t)}_i = \max_w P(y_i = w | X, Y^{(t)}_{obs}) \tag{5}$$

4. Values and probabilities of unmasked tokens $Y^{(t)}_{obs}$ remain unchanged

Example

src	Der Abzug der franzsischen Kampftruppen wurde am 20. November abgeschlossen .
$t = 0$	The departure of the French combat completed completed on 20 November .
$t = 1$	The departure of French combat troops was completed on 20 November .
$t = 2$	The withdrawal of French combat troops was completed on November 20th .

Figure: Illustration of text generation by mask-predict. Example from WMT'14 DE-EN corpus. [2]

Deciding Target Sequence Length

- First: compute CMLM's encoder
- Then: use *LENGTH* token's encoding to predict distribution over sequence's length
- Select top l length candidates with highest probabilities
- Decode same example with different lengths in parallel
- Select the sequence with highest average log-probability:

$$\frac{1}{N} \sum logp_i^{(T)} \tag{6}$$

- -> Translating multiple candidates can improve performance

Experiments

Experimental Setup

- Translation Benchmarks
- Hyperparameters
- Model Distillation

Translation Benchmarks

Evaluation on three standard datasets:

1. WNT'14 EN-DE (4.5M sentence pairs)
2. WMT'16 EN-RO (610k pairs)
3. WMT'17 EN-ZH (20M pairs)

- Data is tokenized
- Performance is evaluated with BLEU
- For EN-ZH: SacreBLEU

Hyperparameters

- Mostly standard parameters for transformers for baseline
- Experiments with number of hidden dimensions
- Weight initialization according to BERT
- Detailed information can be found in their paper

Model Distillation

- Train CMLMs on translations produced by a standard left-to-right transformer model
- For comparison, also train standard left-to-right base transformers (EN-DE and EN-ZH)

Translation Quality

Approach is compared to three other parallel decoding translation methods:

1. Fertility based sequence to sequence model of Gu et al. (2018)
2. CTC-loss transformer of Libovicky and Helcl (2018)
3. Iterative refinement approach of Lee et al. (2018)

Translation Quality

Model	Dimensions (Model/Hidden)	Iterations	WMT'14 EN-DE	WMT'14 DE-EN	WMT'16 EN-RO	WMT'16 RO-EN
NAT w/ Fertility (Gu et al., 2018)	512/512	1	19.17	23.20	29.79	31.44
CTC Loss (Libovický and Helcl, 2018)	512/4096	1	17.68	19.80	19.93	24.71
Iterative Refinement (Lee et al., 2018)	512/512	1	13.91	16.77	24.45	25.73
	512/512	10	21.61	25.48	29.32	30.19
(Dynamic #Iterations)	512/512	?	21.54	25.43	29.66	30.30
Small CMLM with Mask-Predict	512/512	1	15.06	19.26	20.12	20.36
	512/512	4	**24.17**	**28.55**	**30.00**	30.43
	512/512	10	**25.51**	**29.47**	**31.65**	**32.27**
Base CMLM with Mask-Predict	512/2048	1	18.05	21.83	27.32	28.20
	512/2048	4	**25.94**	**29.90**	**32.53**	**33.23**
	512/2048	10	**27.03**	**30.53**	**33.08**	**33.31**
Base Transformer (Vaswani et al., 2017)	512/2048	N	27.30	——	——	——
Base Transformer (Our Implementation)	512/2048	N	27.74	31.09	34.28	33.99
Base Transformer (+Distillation)	512/2048	N	27.86	31.07	——	——
Large Transformer (Vaswani et al., 2017)	1024/4096	N	28.40	——	——	——
Large Transformer (Our Implementation)	1024/4096	N	28.60	31.71	——	——

Figure: Performance (BLEU) of CMLMs with mask-predict, compared to other parallel decoding machine translation models. [2]

Translation Quality

Model	Dimensions (Model/Hidden)	Iterations	WMT'17 EN-ZH	WMT'17 ZH-EN
Base CMLM with Mask-Predict	512/2048	1	24.23	13.64
	512/2048	4	32.63	21.90
	512/2048	10	33.19	23.21
Base Transformer (Our Implementation)	512/2048	N	34.31	23.74
Base Transformer (+Distillation)	512/2048	N	34.44	23.99
Large Transformer (Our Implementation)	1024/4096	N	35.01	24.65

Figure: Performance (BLEU) of CMLMs with mask-predict, compared to the standard (sequential) transformer on WMT'17 EN-ZH. [2]

Decoding Speed

Since CMLMs predict in parallel, mask-predict can translate in a constant number of decoding iterations.

Setup

- Base transformer for baseline system with beam search (EN-DE)
- Also use greedy search for faster but less accurate baseline
- Varied number of mask-predict iterations ($T = 4, ...10$)
- Varied number of length candidates ($l = 1, 2, 3$)
- Measure performance (BLEU) and wall time (seconds)
- Calculate relative decoding speed-up (CMLM time / baseline time)

Decoding Speed

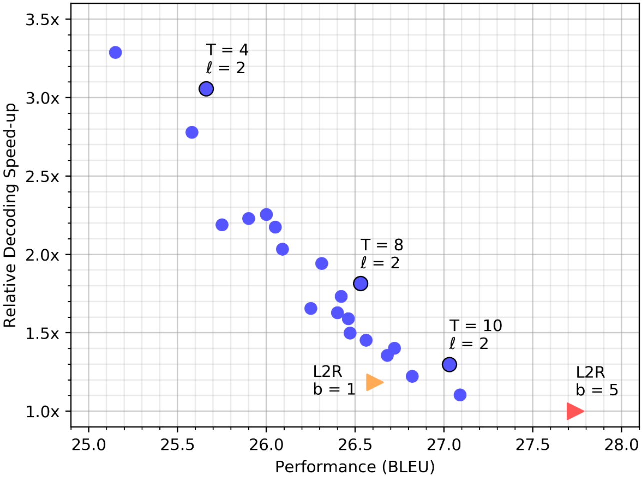

Figure: Trade-off between speed-up and translation quality of a base CMLM with mask-predict, compared to the standard sequentially-decoded base transformer on WMT'14 EN-DE test set. [2]

Analysis

Qualitative Analysis

1. Why are multiple iterations necessary?
2. Do longer sequences need more iterations?
3. Do more length candidates help?
4. Is model distillation necessary?

Why are multiple iterations necessary?

Iterations	WMT'14 EN-DE BLEU	WMT'14 EN-DE Reps	WMT'16 EN-RO BLEU	WMT'16 EN-RO Reps
$T = 1$	18.05	16.72%	27.32	9.34%
$T = 2$	22.91	5.40%	31.08	2.82%
$T = 3$	24.99	2.03%	32.19	1.26%
$T = 4$	25.94	1.07%	32.53	0.87%
$T = 5$	26.30	0.72%	32.62	0.61%

Figure: Percentage of repeating tokens and performance for varying number of iterations (T) [2]

Do longer sequences need more iterations?

	$T = 4$	$T = 10$	$T = N$
$1 \leq N < 10$	21.8	22.4	22.4
$10 \leq N < 20$	24.6	25.9	26.0
$20 \leq N < 30$	24.9	26.7	27.1
$30 \leq N < 40$	24.9	26.7	27.6
$40 \leq N$	25.0	27.5	28.1

Figure: Performance with different number of iterations T, grouped by target sequence length N. [2]

Do more length candidates help?

Length Candidates	WMT'14 EN-DE BLEU	WMT'14 EN-DE LP	WMT'16 EN-RO BLEU	WMT'16 EN-RO LP
$\ell = 1$	26.56	16.1%	32.75	13.8%
$\ell = 2$	27.03	30.6%	33.06	26.1%
$\ell = 3$	**27.09**	43.1%	**33.11**	39.6%
$\ell = 4$	**27.09**	53.1%	32.13	49.2%
$\ell = 5$	27.03	62.2%	33.08	57.5%
$\ell = 6$	26.91	69.5%	32.91	64.3%
$\ell = 7$	26.71	75.5%	32.75	70.4%
$\ell = 8$	26.59	80.3%	32.50	74.6%
$\ell = 9$	26.42	83.8%	32.09	78.3%
Gold	27.27	—	33.20	—

Figure: Performance with 10 iterations varied by the number of length candidates l. Length precision (LP) is the percentage of examples that contain the correct length as one of their candidates.[2]

Is model distillation necessary?

Iterations	WMT'14 EN-DE		WMT'16 EN-RO	
	Raw	Dist	Raw	Dist
$T = 1$	10.64	**18.05**	21.22	**27.32**
$T = 4$	22.25	**25.94**	31.40	**32.53**
$T = 10$	24.61	**27.03**	32.86	**33.08**

Figure: Performance trained with raw data (Raw) or knowledge distillation from an autoregressive model (Dist)[2]

Conclusion

Conclusion

- Approach outperforms previous parallel decoding methods
- Approaches the performance of sequential autoregressive models (decoding faster)
- Problem: need to condition on the target's length
- Problem: dependence on knowledge distillation
- Significant step forward in non-autoregressive and parallel decoding approaches to machine translation
- Also useful for generating text efficiently

References

References I

- Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: **CoRR** abs/1810.04805 (2018). arXiv: 1810.04805. URL: http://arxiv.org/abs/1810.04805.
- Marjan Ghazvininejad et al. "Constant-Time Machine Translation with Conditional Masked Language Models". In: **CoRR** abs/1904.09324 (2019). arXiv: 1904.09324. URL: http://arxiv.org/abs/1904.09324.
- Jiatao Gu et al. "Non-Autoregressive Neural Machine Translation". In: **CoRR** abs/1711.022 (2017). arXiv: 1711.02281. URL: http://arxiv.org/abs/1711.02281.
- Jiatao Gu et al. "Non-Autoregressive Neural Machine Translation". In: **CoRR** abs/1711.022 (2017). arXiv: 1711.02281. URL: http://arxiv.org/abs/1711.02281.
- Guillaume Lample and Alexis Conneau. "Cross-lingual Language Model Pretraining". In: **CoRR** abs/1901.07291 (2019). arXiv: 1901.07291. URL: http://arxiv.org/abs/1901.07291.

References II

Jason Lee, Elman Mansimov, and Kyunghyun Cho. "Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement". In: **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing**. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 1173–1182. DOI: 10.18653/v1/D18-1149. URL: https://www.aclweb.org/anthology/D18-1149.

Jindrich Libovický and Jindrich Helcl. "End-to-End Non-Autoregressive Neural Machine Translation with Connectionist Temporal Classification". In: **CoRR** abs/1811.04719 (2018). arXiv: 1811.04719. URL: http://arxiv.org/abs/1811.04719.

Kaitao Song et al. "MASS: Masked Sequence to Sequence Pre-training for Language Generation". In: **CoRR** abs/1905.02450 (2019). arXiv: 1905.02450. URL: http://arxiv.org/abs/1905.02450.

Ashish Vaswani et al. "Attention Is All You Need". In: **CoRR** abs/1706.03762 (2017). arXiv: 1706.03762. URL: http://arxiv.org/abs/1706.03762.