Fast Structured Decoding for Sequence Models

(Sun et al., NeurIPS 2019)

愛媛大 M1 出口 祥之
e-mail: <deguchi@ai.cs.ehime-u.ac.jp>

第 1 回 NAT 勉強会: 2020/03/05

Links

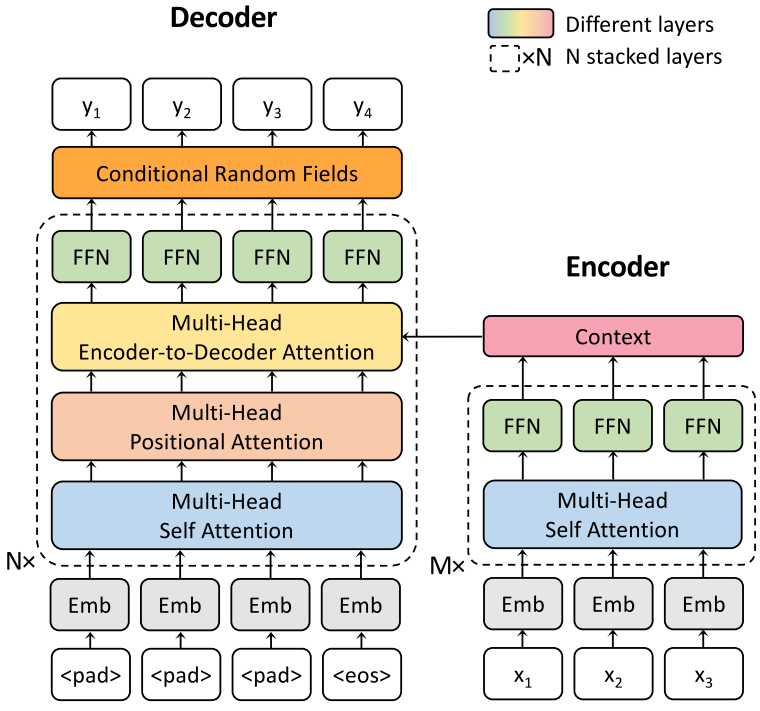
- Fast Structured Decoding for Sequence Models (Sun et al, NeurIPS 2019)
  - `https://papers.nips.cc/paper/8566-fast-structured-decoding-for-sequence-models`
- ソースコード (Fairseq)
  - `nat_crf_transformer.py`
  - `dynamic_crf_layer.py`

Introduction

- 従来の Non-Autoregressive Transformer (NAT) は出力単語同士において条件付き独立を仮定
  - 単語の共起性などをうまく考慮できず, 生成文の一貫性を犠牲にする
- linear-chain Conditional Random Fields (CRF) レイヤを導入することで隣り合った単語間の共起性を考慮
  - 後述する低ランク近似とビーム近似により, 語彙数 $|\mathcal{V}|$ 依存の巨大な計算量 $O(|\mathcal{V}|^2)$ を抑える
  - 提案手法 Dynamic CRF transition により性能改善

Transformer-based Non-autoregressive Translation Model

- (Gu+, 2017) モデルをベースにした翻訳モデル : NART
- デコーダの入力をシンプルに設計
  - fertility による単語コピーを行わない
  - <pad> 系列の末尾に <eos> を付加した系列を用いる
    - 訓練時は目的言語文の文長個
    - 推論時は後述
  - 入力がとてもシンプルだが, 実験ではうまく機能する



(b) Non-Autoregressive Transformer with Conditional Random Fields

Conditional random field

- Non-Autoregressive なモデルのマルチモダリティ問題を構造化推論モジュールにより対処
  - 出力単語をラベルとした系列ラベリング問題と見なし, CRF を用いる

$$P(y|x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^{n} s(y_i, x, i) + \sum_{i=2}^{n} t(y_{i-1}, y_i, x, i)\right)$$

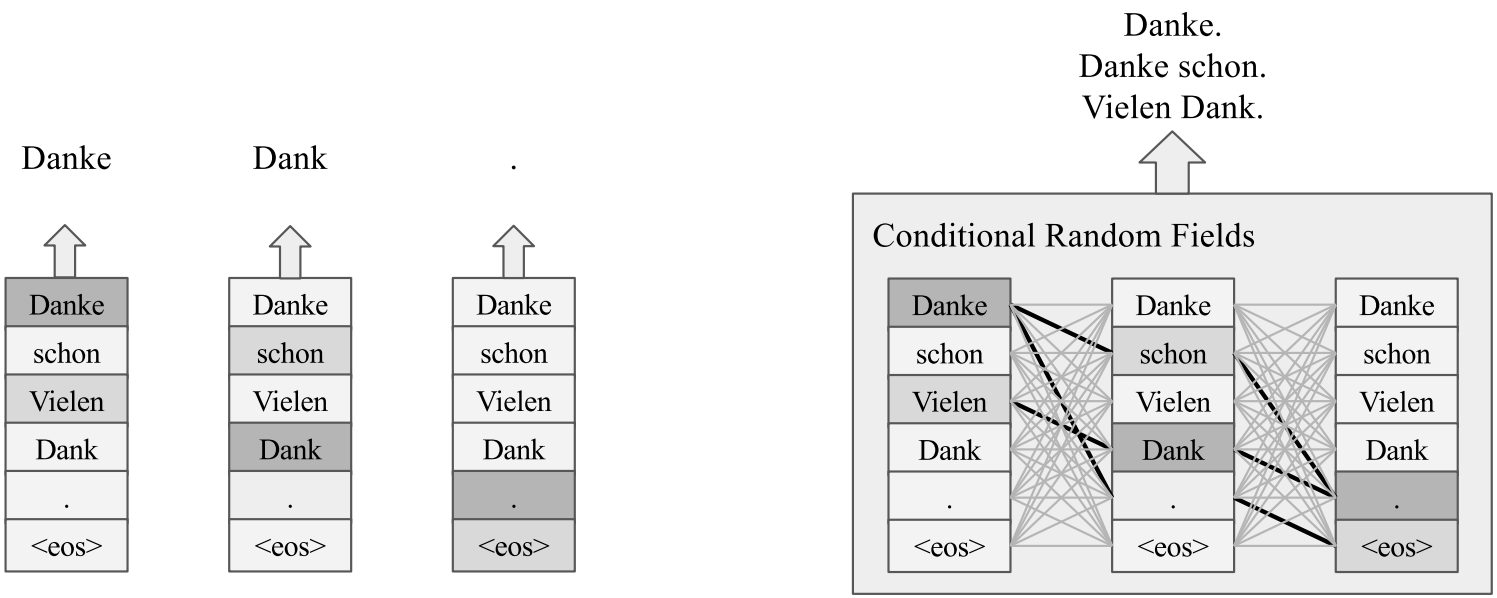


Figure 2: Illustration of the decoding inconsistency problem in non-autoregressive decoding and how a CRF-based structured inference module solves it.

Incorporating CRF into NART model

$$P(y|x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^{n} s(y_i, x, i) + \sum_{i=2}^{n} t(y_{i-1}, y_i, x, i)\right)$$

- CRF をそのままモデルに適用すると遷移行列のサイズが $|\mathcal{V}| \times |\mathcal{V}|$ になる
  - 語彙数 32k のとき, $32k^2$ = 1.024B となり, 実用不可
  - 低ランク近似とビーム近似により解決

Low-rank approximation for transition matrix

- 遷移行列 M を低ランクな行列の積で表現

$$M = E_1 E_2^\top \qquad \text{where} E_1, E_2 \in \mathbb{R}^{|\mathcal{V}| \times d_t}$$

- 分配関数 $Z(x)$ ・ビタビ復号以外の計算効率化

Beam approximation for CRF

- 分配関数 $Z(x)$ ・ビタビ復号時の計算に使用
- 各位置でk-bestを求める
  - 分配関数 $Z(x)$ に正解文の系列を含ませて計算
- k-best遷移行列を用いて計算($|\mathcal{V}| \times |\mathcal{V}| \to k \times k$)
- 低ランク近似とビーム近似によりCRFの計算量は $O(|\mathcal{V}|^2) \to O(nk^2)$

Dynamic CRF transition

- 遷移行列 M は一系列に対して固定
- 位置 i ごとに変化する動的な遷移行列 M^i を定義

$$M^i_{dynamic} = f([h_{i-1}, h_i])$$
$$M^i = E_1 M^i_{dynamic} E_2^\top$$
$$t(y_{i-1}, y_i, x, i) = M^i_{y_{i-1}, y_i}$$

※ $f : \mathbb{R}^{2d_{model}} \to \mathbb{R}^{d_t \times d_t}$ は2層のFeed Forward Network

Joint training with vanilla non-autoregressive loss

- モデル学習を効率的に行うため, 目的関数 $\mathcal{L}$ はCRFの出力のNLL損失($\mathcal{L}_{CRF}$)とNARのNLL損失($\mathcal{L}_{NAR}$)の荷重和とする

$$\mathcal{L} = \mathcal{L}_{CRF} + \lambda \mathcal{L}_{NAR}$$

※ λ は損失の重みをコントロールするハイパーパラメータ

Inference

目的言語文の文長 T'

訓練時 正解文長を使用

推論時 ソース文長 T から決定: $T' = T + C$

- C は訓練データの文長の統計情報により決定される定数
  - 言語ごとの文の平均長によって決定

推論時の rescoreing

- $[(T + C) - B, (T + C) + B]$ の範囲で異なるターゲット文長の翻訳候補を生成
  - B は 4 or 9 に設定 $\rightarrow$ 候補の数は 9 or 19 個
- Autoregressive な Transformer を教師モデルとして最適な翻訳を選択

Experiments

- データセット : WMT14 En-De/De-En, IWSLT14 De-En
- 低ランク近似の遷移行列の埋め込み次元 : $d_t = 32$
- ビーム近似のビーム幅 : $k = 64$
- 目的関数 $\mathcal{L} = \mathcal{L}_{CRF} + \lambda \mathcal{L}_{NAR}$: $\lambda = 0.5$

Results

- 既存のNon-autoregressiveモデルより大幅に性能改善
- WMTのEn-Deにおいて, 強力なAutoregressiveモデル(LSTM-base, CNN-base)よりも優れた性能
- ARTとNARTの性能差(BLEU)を0.61まで縮めた
- latencyについて, NART-CRF/DCRFは, rescoreringなしで11.1/10.4, rescoreringありでも4.45/4.39 倍の速度向上

Table 2: Performance of BLEU score on WMT14 En-De/De-En and IWSLT14 De-En tasks. The number in the parentheses denotes the performance gap between NART models and their ART teachers. "/" denotes that the results are not reported. LSTM-based results are from [2, 27]; CNN-based results are from [5, 28]; Transformer [1] results are based on our own reproduction.[6]

Models	WMT14 En-De	WMT14 De-En	IWSLT14 De-En	Latency	Speedup
Autoregressive models					
LSTM-based [2]	24.60	/	28.53	/	/
CNN-based [5]	26.43	/	32.84	/	/
Transformer [1] (beam size = 4)	27.41	31.29	33.26	$387ms^{\ddagger}$	1.00×
Non-autoregressive models					
FT [6]	17.69 (5.76)	21.47 (5.55)	/	$39ms^{\dagger}$	$15.6\times^{\dagger}$
FT [6] (rescoring 10)	18.66 (4.79)	22.41 (4.61)	/	$79ms^{\dagger}$	$7.68\times^{\dagger}$
FT [6] (rescoring 100)	19.17 (4.28)	23.20 (3.82)	/	$257ms^{\dagger}$	$2.36\times^{\dagger}$
IR [9] (adaptive refinement)	21.54 (3.03)	25.43 (3.04)	/	/	$2.39\times^{\dagger}$
LT [15]	19.80 (7.50)	/	/	$105ms^{\dagger}$	/
LT [15] (rescoring 10)	21.00 (6.30)	/	/	/	/
LT [15] (rescoring 100)	22.50 (4.80)	/	/	/	/
CTC [13]	17.68 (5.77)	19.80 (7.22)	/	/	$3.42\times^{\dagger}$
ENAT-P [29]	20.26 (7.15)	23.23 (8.06)	25.09 (7.46)	$25ms^{\dagger}$	$24.3\times^{\dagger}$
ENAT-P [29] (rescoring 9)	23.22 (4.19)	26.67 (4.62)	28.60 (3.95)	$50ms^{\dagger}$	$12.1\times^{\dagger}$
ENAT-E [29]	20.65 (6.76)	23.02 (8.27)	24.13 (8.42)	$24ms^{\dagger}$	$25.3\times^{\dagger}$
ENAT-E [29] (rescoring 9)	24.28 (3.13)	26.10 (5.19)	27.30 (5.25)	$49ms^{\dagger}$	$12.4\times^{\dagger}$
NAT-REG [8]	20.65 (6.65)	24.77 (6.52)	23.89 (9.63)	$22ms^{\dagger}$	$27.6\times^{\dagger}$
NAT-REG [8] (rescoring 9)	24.61 (2.69)	28.90 (2.39)	28.04 (5.48)	$40ms^{\dagger}$	$15.1\times^{\dagger}$
VQ-VAE [16] (compress 8×)	26.70 (1.40)	/	/	$81ms^{\dagger}$	$4.08\times^{\dagger}$
VQ-VAE [16] (compress 16×)	25.40 (2.70)	/	/	$58ms^{\dagger}$	$5.71\times^{\dagger}$
Non-autoregressive models (Ours)					
NART	20.27 (7.14)	22.02 (9.27)	23.04 (10.22)	$26ms^{\ddagger}$	$14.9\times^{\ddagger}$
NART (rescoring 9)	24.22 (3.19)	26.21 (5.08)	26.79 (6.47)	$50ms^{\ddagger}$	$7.74\times^{\ddagger}$
NART (rescoring 19)	24.99 (2.42)	26.60 (4.69)	27.36 (5.90)	$74ms^{\ddagger}$	$5.22\times^{\ddagger}$
NART-CRF	23.32 (4.09)	25.75 (5.54)	26.39 (6.87)	$35ms^{\ddagger}$	$11.1\times^{\ddagger}$
NART-CRF (rescoring 9)	26.04 (1.37)	28.88 (2.41)	29.21 (4.05)	$60ms^{\ddagger}$	$6.45\times^{\ddagger}$
NART-CRF (rescoring 19)	26.68 (0.73)	29.26 (2.03)	29.55 (3.71)	$87ms^{\ddagger}$	$4.45\times^{\ddagger}$
NART-DCRF	**23.44 (3.97)**	**27.22 (4.07)**	**27.44 (5.82)**	$37ms^{\ddagger}$	$10.4\times^{\ddagger}$
NART-DCRF (rescoring 9)	**26.07 (1.34)**	**29.68 (1.61)**	**29.99 (3.27)**	$63ms^{\ddagger}$	$6.14\times^{\ddagger}$
NART-DCRF (rescoring 19)	**26.80 (0.61)**	**30.04 (1.25)**	**30.36 (2.90)**	$88ms^{\ddagger}$	$4.39\times^{\ddagger}$

Analysis

- Dynamic CRFの効果は, En-Deでは小さく , De-Enで大きい
  - 言語間の特性が原因
- ビーム近似の有効性
  - 完全な遷移行列にどれほど適合するか？
  - ビーム近似のビーム幅を64, 評価時のビーム幅 k を変えて実験
    - $k = 16$ 以降の上がり幅は小さい
    - 訓練時のビーム幅よりも小さい $k = 16$ の時点ですでに良く近似できている

Conclusion and Future Work

Conclusion

- Non-autoregressive なモデルのマルチモダリティ問題を解決するため, linear-chain CRFを導入し, 単語間の共起関係を扱えるようにした
- 計算量が語彙数に依存しない, 低ランク近似, ビーム近似を提案
- 位置毎のコンテキストをモデル化するDynamic CRFを提案

Future Work

- Autoregressiveモデルとのギャップを埋める
- rescoring の処理によってlatencyが増加している
  - 目的言語文の文長を予測するモジュールが役立つかもしれない

Models	WMT14 En-De	WMT14 De-En	IWSLT14 De-En	Latency	Speedup
Autoregressive models					
Transformer [1] (beam size = 4)	27.41	31.29	33.26	$387ms^{\ddagger}$	1.00×
Non-autoregressive models (Ours)					
NART-CRF	23.32 (4.09)	25.75 (5.54)	26.39 (6.87)	$35ms^{\ddagger}$	$11.1\times^{\ddagger}$
NART-CRF (rescoring 9)	26.04 (1.37)	28.88 (2.41)	29.21 (4.05)	$60ms^{\ddagger}$	$6.45\times^{\ddagger}$
NART-CRF (rescoring 19)	26.68 (0.73)	29.26 (2.03)	29.55 (3.71)	$87ms^{\ddagger}$	$4.45\times^{\ddagger}$
NART-DCRF	**23.44 (3.97)**	**27.22 (4.07)**	**27.44 (5.82)**	$37ms^{\ddagger}$	$10.4\times^{\ddagger}$
NART-DCRF (rescoring 9)	**26.07 (1.34)**	**29.68 (1.61)**	**29.99 (3.27)**	$63ms^{\ddagger}$	$6.14\times^{\ddagger}$
NART-DCRF (rescoring 19)	**26.80 (0.61)**	**30.04 (1.25)**	**30.36 (2.90)**	$88ms^{\ddagger}$	$4.39\times^{\ddagger}$