

Recurrent Neural Network Grammers

Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros,
Noah A. Smith

導入

Recurrent Neural Network Grammars (RNNGs)

- ▶ 文の確率的生成モデル
 - ▶ 単語や句の入れ子的・階層的構造を陽に表現
- ▶ 問題：構文解析，文生成
- ▶ 動機：Sequential な Recurrent Neural Networks (RNNs) は自然言語の潜在的な入れ子構造を考慮できていない
- ▶ 構文解析または文生成のアルゴリズム

提案手法

RNNG の定義

$$RNNG := (N, \Sigma, \Theta)$$

$$\left\{ \begin{array}{l} N : \text{非終端記号の有限集合} \\ \Sigma : \text{終端記号の有限集合} (N \cup \Sigma = \emptyset) \\ \Theta : \text{NN のパラメータ} \end{array} \right.$$

提案手法

構文解析のアルゴリズム

$$f : X \rightarrow Y$$

$$\left\{ \begin{array}{l} x : \text{終端記号（単語）の列（入力）} \\ y : \text{構文木（出力）} \\ S : \text{スタック} \\ B : \text{入力バッファ} \end{array} \right.$$

- ▶ スタックの要素：終端記号，open または closed な非終端記号
- ▶ 入力バッファの要素：終端記号

提案手法

構文解析のアルゴリズム

- ▶ 初期状態

$$B = [T_1, \dots, T_n]$$

$$S = \emptyset$$

提案手法

構文解析のアルゴリズム

▶ 遷移の制約

- ▶ n : スタック内の open な非終端記号の数

遷移	制約
nt(X)	$B \neq \emptyset \wedge n < 100$
SHIFT	$B \neq \emptyset \wedge n \geq 1$
REDUCE	スタック内の一番上の要素が open な非終端記号でない $\wedge (n \geq 2 \vee B = \emptyset)$

提案手法

文生成のアルゴリズム

$$f : X \rightarrow Y$$

$$\left\{ \begin{array}{l} x : ? \\ y : \text{終端記号 (単語) の列 (出力)} \\ S : \text{スタック} \\ \textcolor{red}{T} : \text{出力バッファ} \end{array} \right.$$

- ▶ スタックの要素：終端記号，open または closed な非終端記号
- ▶ 出力バッファの要素：終端記号

提案手法

文生成のアルゴリズム

- ▶ 初期状態

$$B = \emptyset$$

$$S = \emptyset$$

提案手法

文生成のアルゴリズム

▶ 遷移の制約

- ▶ n : スタック内の open な非終端記号の数
- ▶ $A_G(T, S, n)$

遷移	制約
GEN(X)	$n \geq 1$
REDUCE	スタック内の一番上の要素が open な非終端記号でない $\wedge n \geq 1$