

目的言語側における係り受け構造 を考慮したニューラル機械翻訳

江里口瑛子[†], Kyunghyun Cho^{††}, 鶴岡慶雅[†]

[†]東京大学, ^{††} New York University

背景

- ニューラル機械翻訳モデル (NMT)
 - Encoder-Decoderモデルによる系列データ変換タスクとしてモデル化 (Cho et al., 2014; Sutskever et al., 2014)
 - 欧州言語翻訳タスク (WMT), アジア言語翻訳タスク (WAT)において高い翻訳性能が報告
 - 構文情報を利活用することで性能改善 (Eriguchi et al., 2016; Sennrich et al., 2016)

白い犬がソファの上で寝ている。



The white dog is sleeping on the sofa.

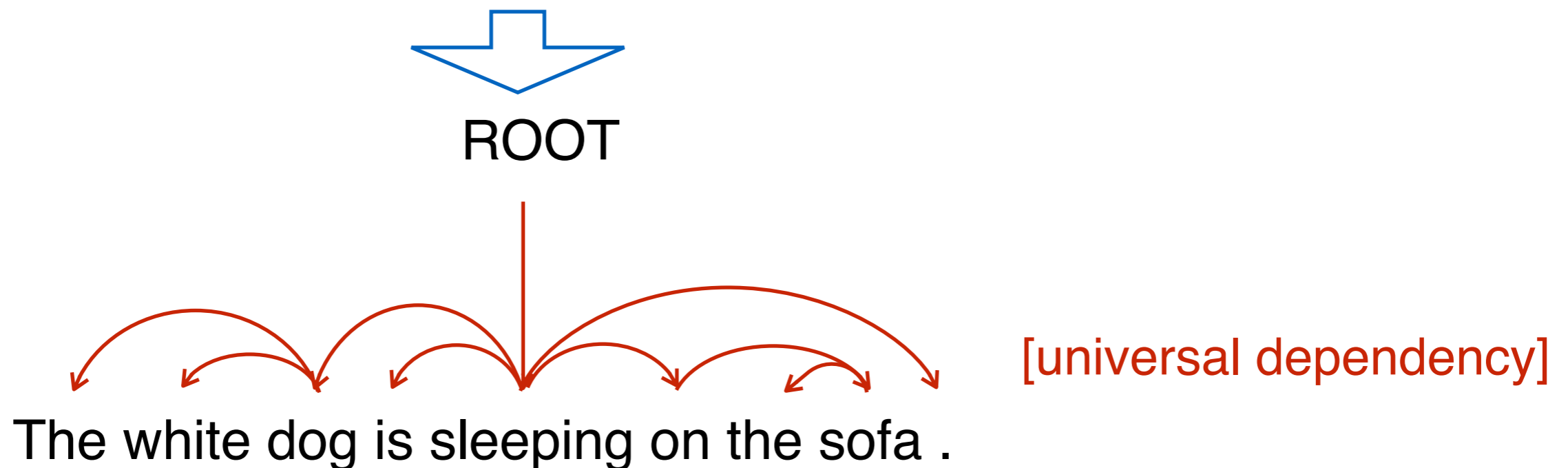
関連研究

- 構文情報を扱うNMTモデル
 - [原言語] 句構造情報 ([Eriguchi et al., 2016](#))
 - [原言語] 構文ラベル情報 (POSタグ, 係り受けラベル, レンマ, BIOタグ) ([Sennrich et al., 2016](#))
 - [原言語] 潜在的構造を自動推定 ([Hashimoto and Tsuruoka, 2017](#); [Kim et al., 2017](#))
 - [原言語・目的言語] CCGパーサーのスーパータグ ([Nadejde et al., 2017](#))
 - [目的言語] 係り受け木構造をルートからトップダウンに生成 ([Alvarez-Melis and Jaakkola, 2017](#))
 - テスト時、構文木候補の探索が困難なため貪欲探索

本発表

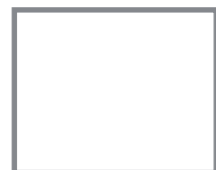
- 目的言語側の構文情報を考慮したNMTモデルの提案
 - 構文情報: 単語間の係り受け構造
 - 遷移型構文解析モデルをNMTモデル内で (選択的に) モデル化
 - テスト時の翻訳スピードは既存モデルと同一

白い犬がソファの上で寝ている。



遷移型構文解析による係り受け構文解析

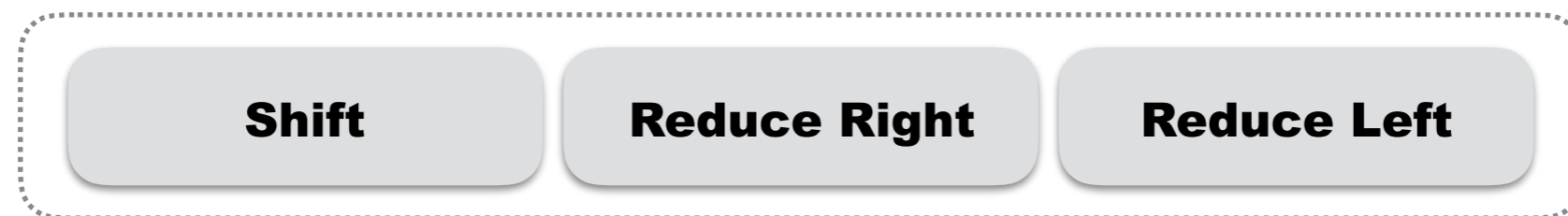
- 文の構文木構造を、3種のAction系列状態で表現
- Actionに従い、Stack上に係り受け構文木を構成
 - Bufferには、入力文の単語列 + ROOTが存在



The white dog is sleeping on the sofa . ROOT

Stack

Buffer



Action

遷移型構文解析による係り受け構文解析

- 文の構文木構造を、3種のAction系列状態で表現
- Actionに従い、Stack上に係り受け構文木を構成
 - Bufferには、入力文の単語列 + ROOTが存在

The

white dog is sleeping on the sofa . ROOT

Stack

Buffer

Shift

Reduce Right

Reduce Left

Action

遷移型構文解析による係り受け構文解析

- 文の構文木構造を、3種のAction系列状態で表現
- Actionに従い、Stack上に係り受け構文木を構成
 - Bufferには、入力文の単語列 + ROOTが存在

The white

dog is sleeping on the sofa . ROOT

Stack

Buffer

Shift

Reduce Right

Reduce Left

Action

遷移型構文解析による係り受け構文解析

- 文の構文木構造を、3種のAction系列状態で表現
- Actionに従い、Stack上に係り受け構文木を構成
 - Bufferには、入力文の単語列 + ROOTが存在

The white dog

is sleeping on the sofa . ROOT

Stack

Buffer

Shift

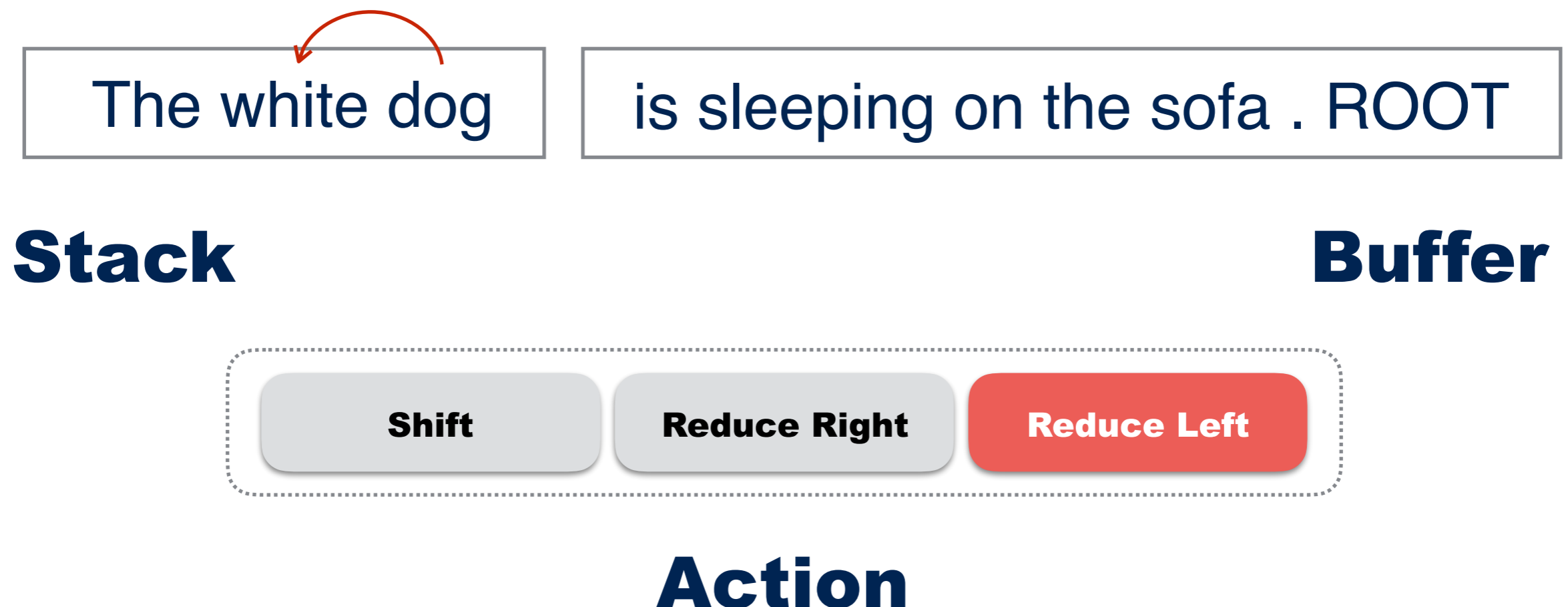
Reduce Right

Reduce Left

Action

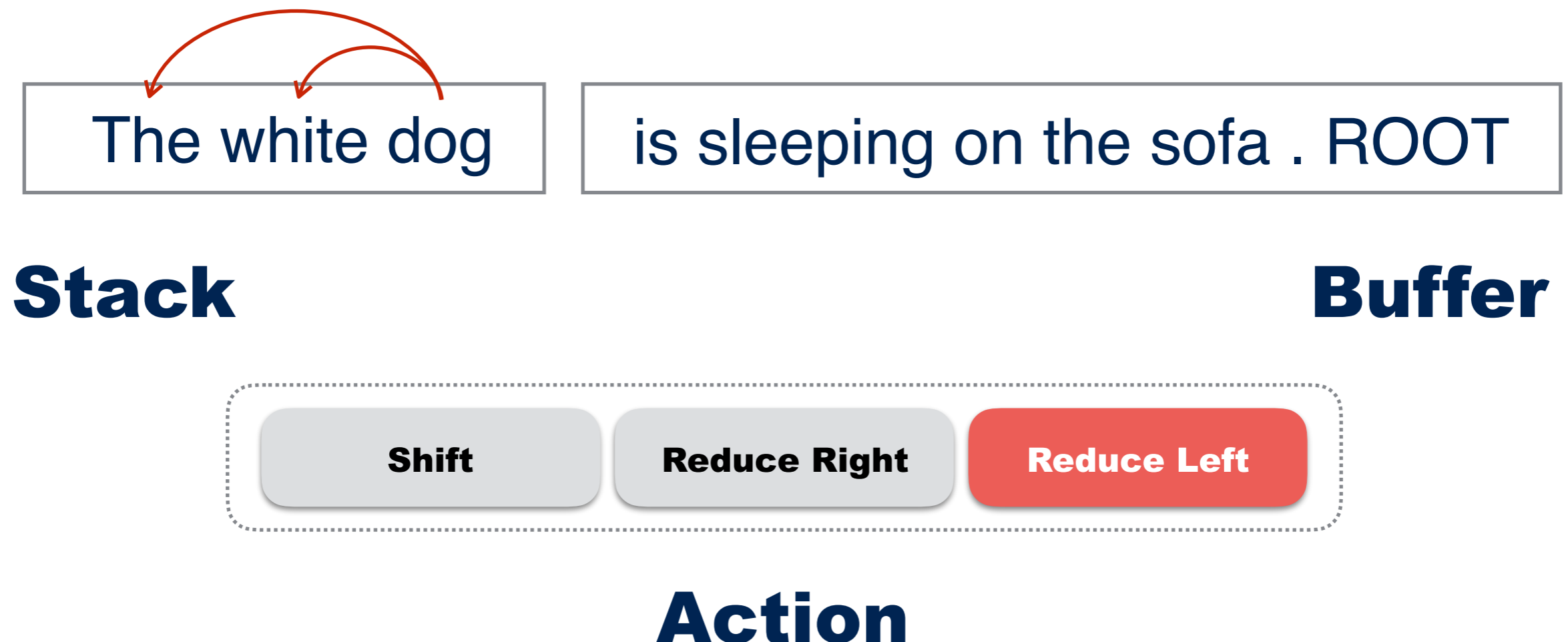
遷移型構文解析による係り受け構文解析

- 文の構文木構造を、3種のAction系列状態で表現
- Actionに従い、Stack上に係り受け構文木を構成
 - Bufferには、入力文の単語列 + ROOTが存在



遷移型構文解析による係り受け構文解析

- 文の構文木構造を、3種のAction系列状態で表現
- Actionに従い、Stack上に係り受け構文木を構成
 - Bufferには、入力文の単語列 + ROOTが存在



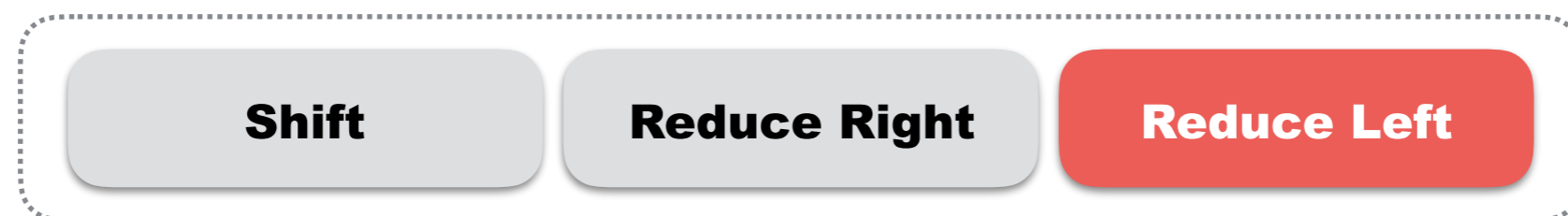
遷移型構文解析による係り受け構文解析

...途中過程省略...



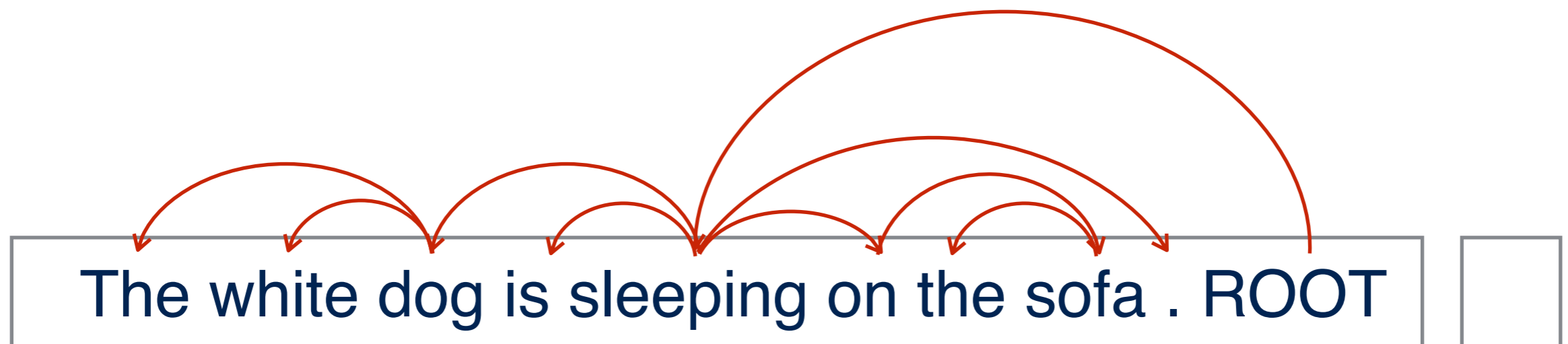
Stack

Buffer



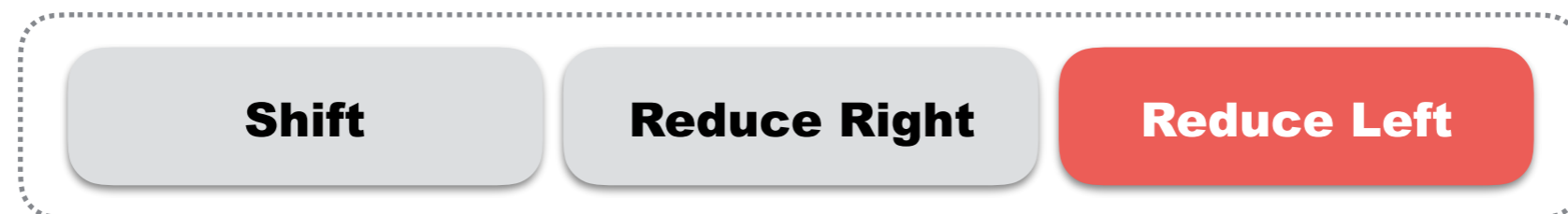
Action

遷移型構文解析による係り受け構文解析



Stack

Buffer

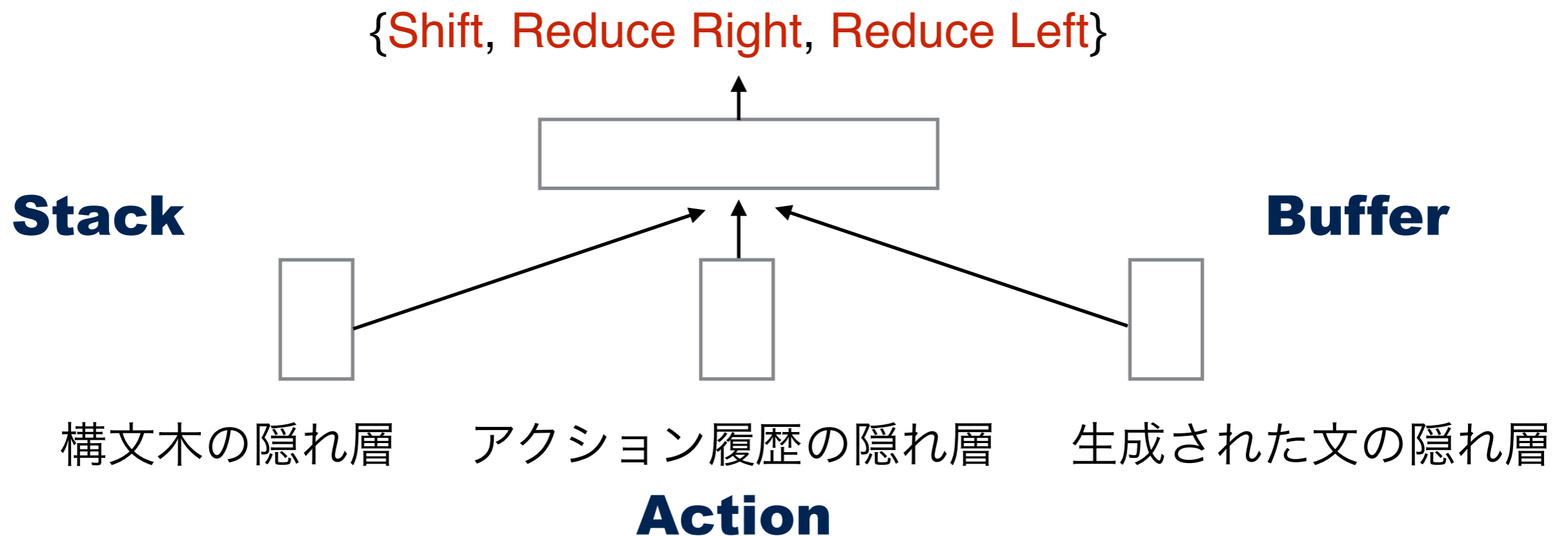


Action

Recurrent Neural Network Grammar (RNNG)

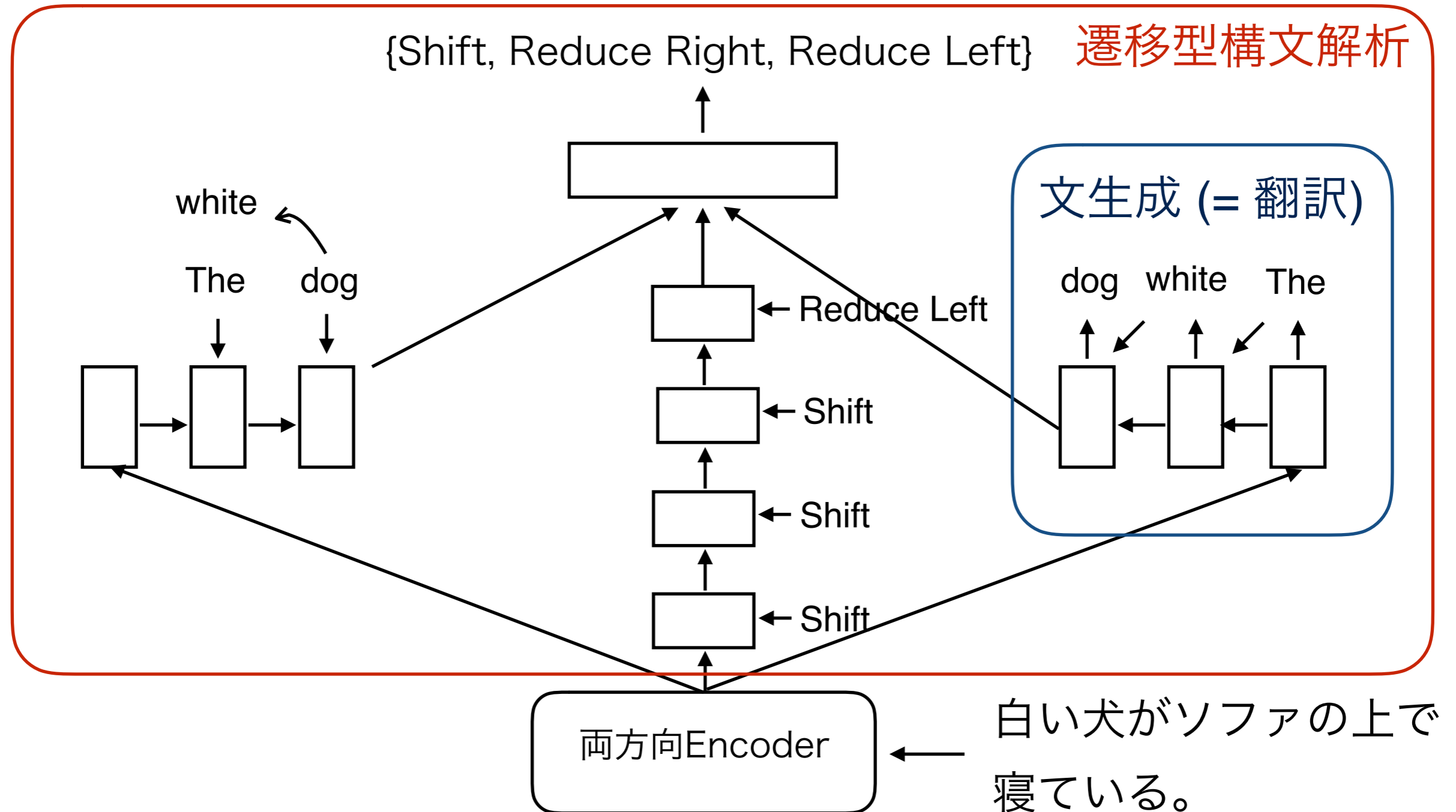
(Dyer et al., 2016)

- ・ 遷移型構文解析と文生成の同時学習モデル
 - ・ アクション系列を主体に単語系列も予測
 - ・ Shiftが選択された時、1単語生成
 - ・ 言語モデル、係り受け構文解析タスクで最高精度 (Kuncoro et al., 2017)



提案手法 (NMT + RNNG)

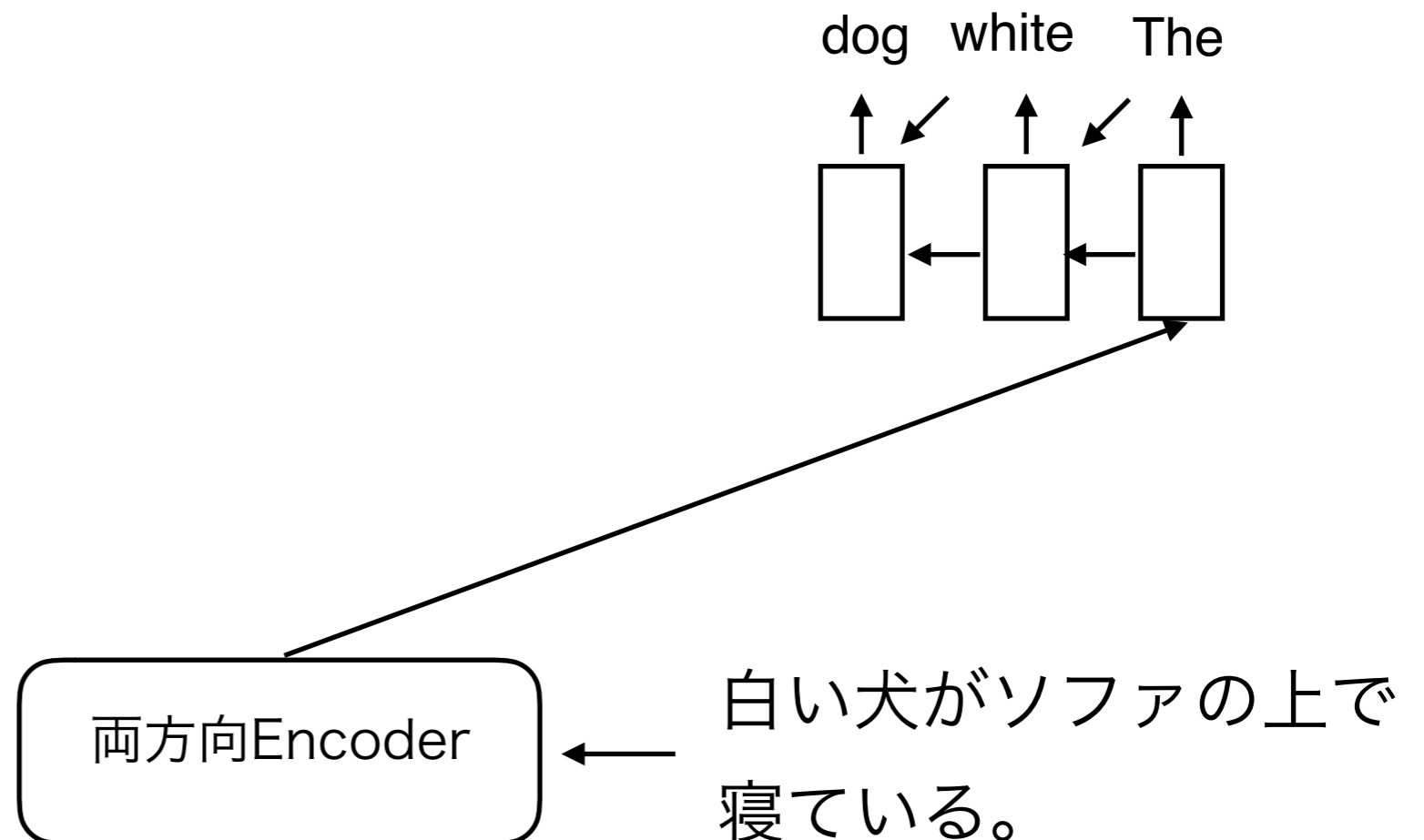
- NMTとRNNGを単一ネットワークでモデル化



提案手法 (NMT + RNNG)

- NMTとRNNGを単一ネットワークでモデル化
 - アテンション構造を有するNMTデコーダ (Luong et al., 2015)

アテンションに基づくDecoder



提案手法 (NMT + RNNG)

- Stack-LSTM (Dyer et al., 2015) を用いて、各隠れ層を算出

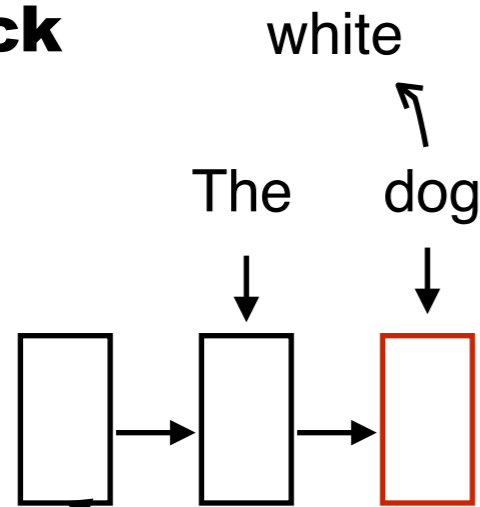
構文木の隠れ層

アクション履歴の隠れ層

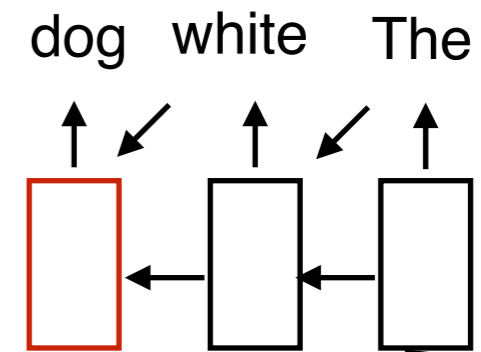
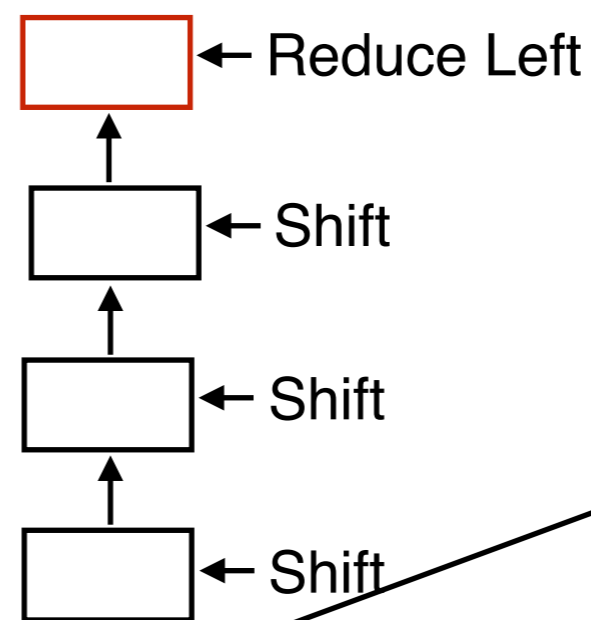
生成された文の隠れ層

アテンションに基づく Decoder
(= Buffer)

Stack



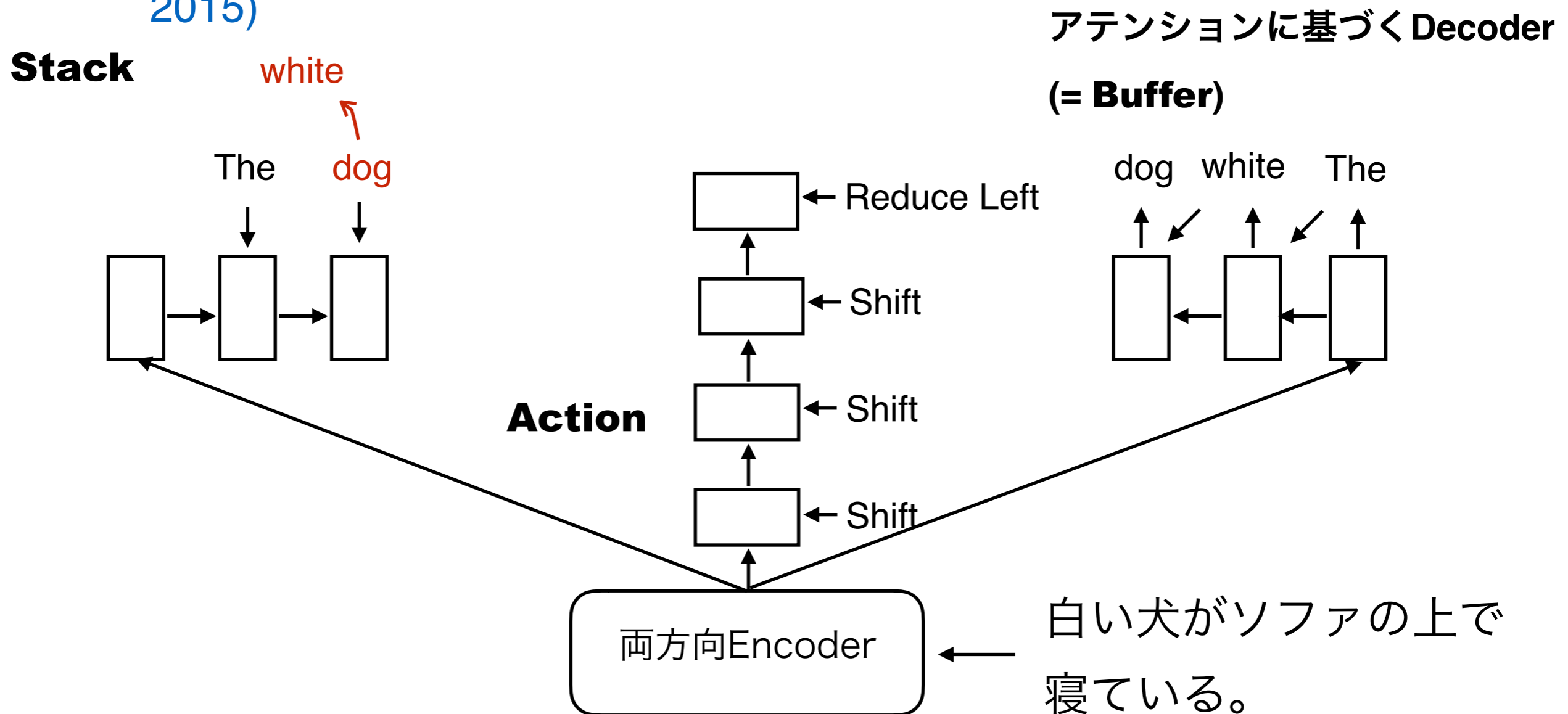
Action



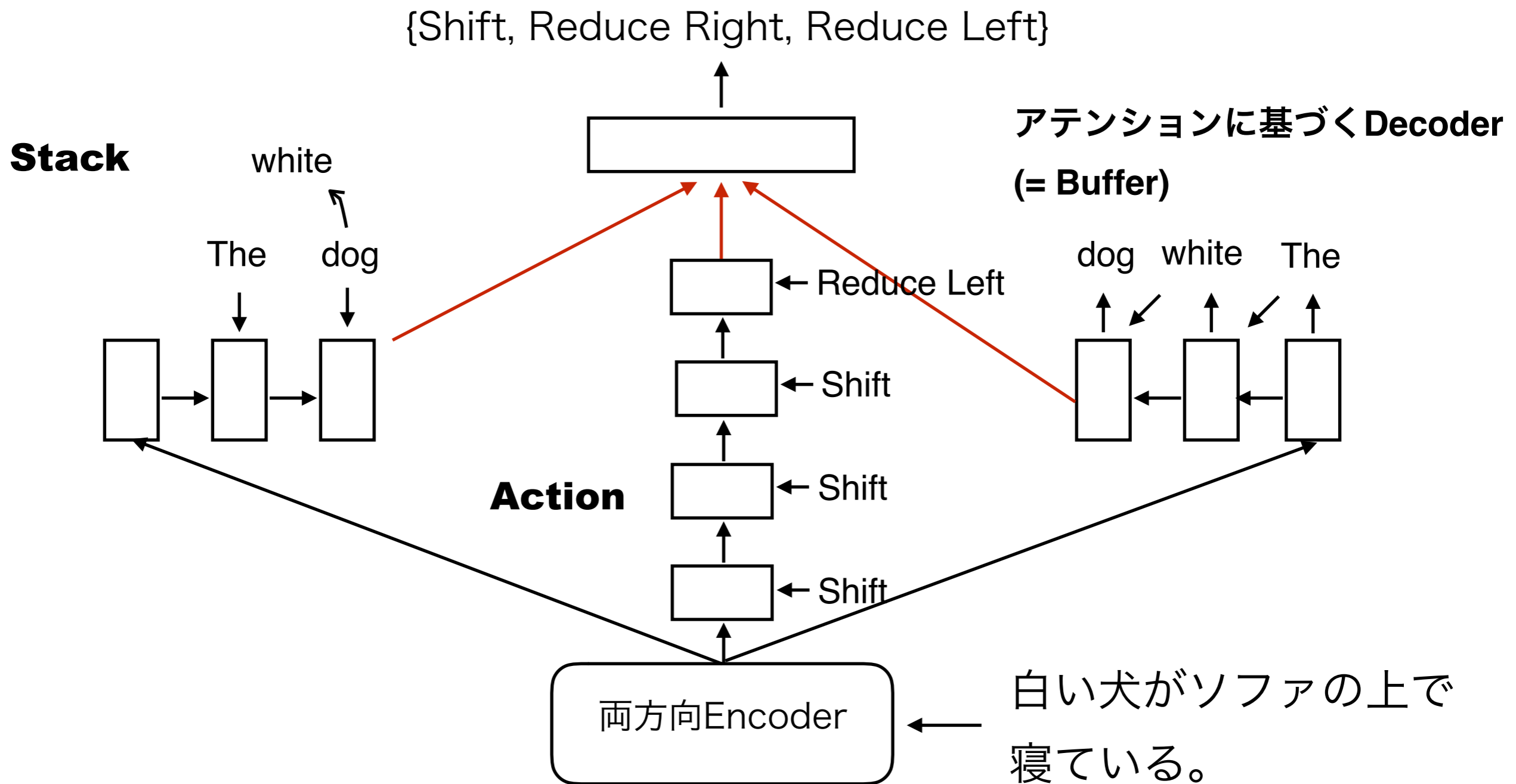
白い犬がソファの上で
寝ている。

提案手法 (NMT + RNNG)

- Stack-LSTM (Dyer et al., 2015) を用いて、各隠れ層を算出
 - 係り受けの構成表現は親子, 係り受けラベルから構成 (Dyer et al., 2015)

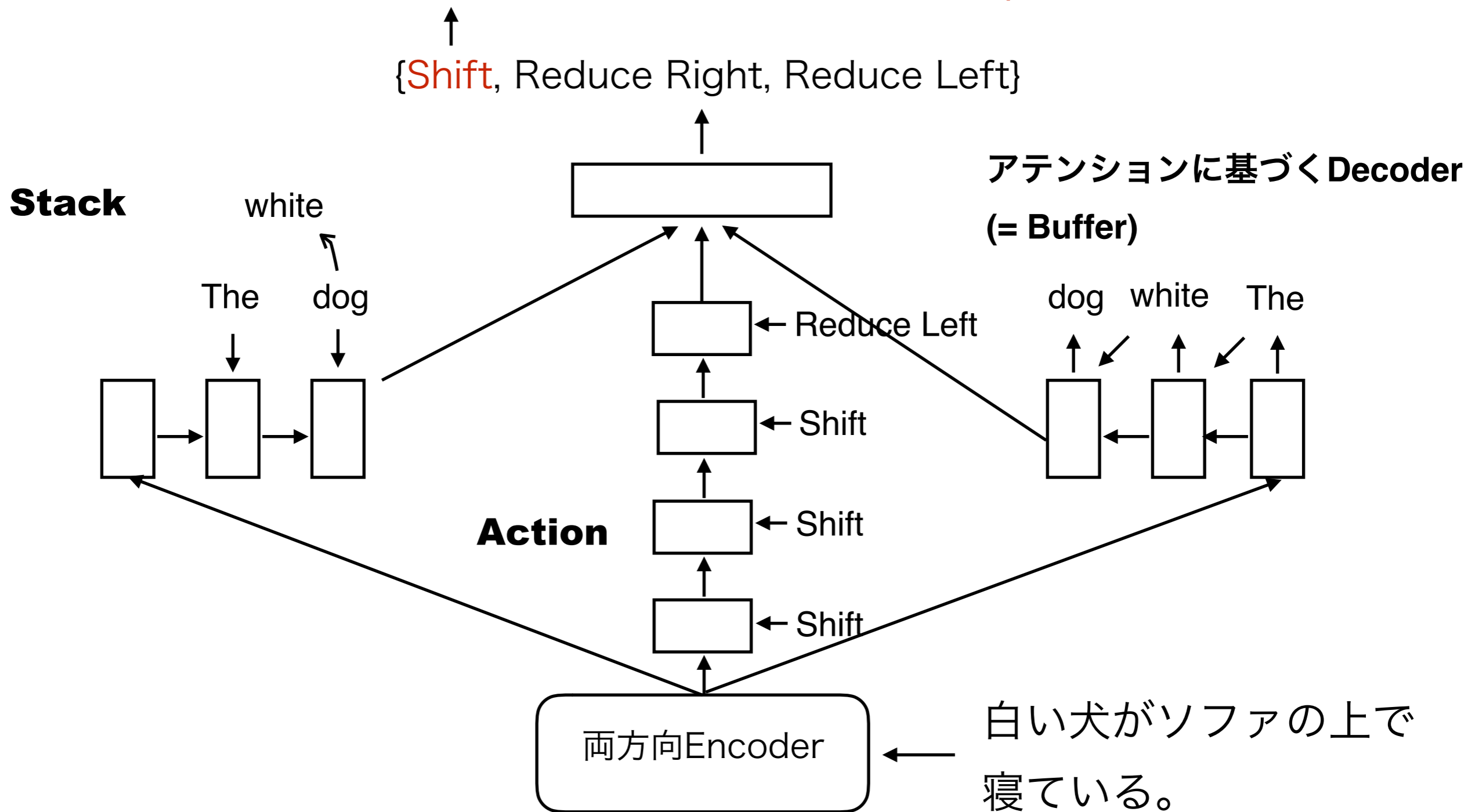


提案手法 (NMT + RNNG)



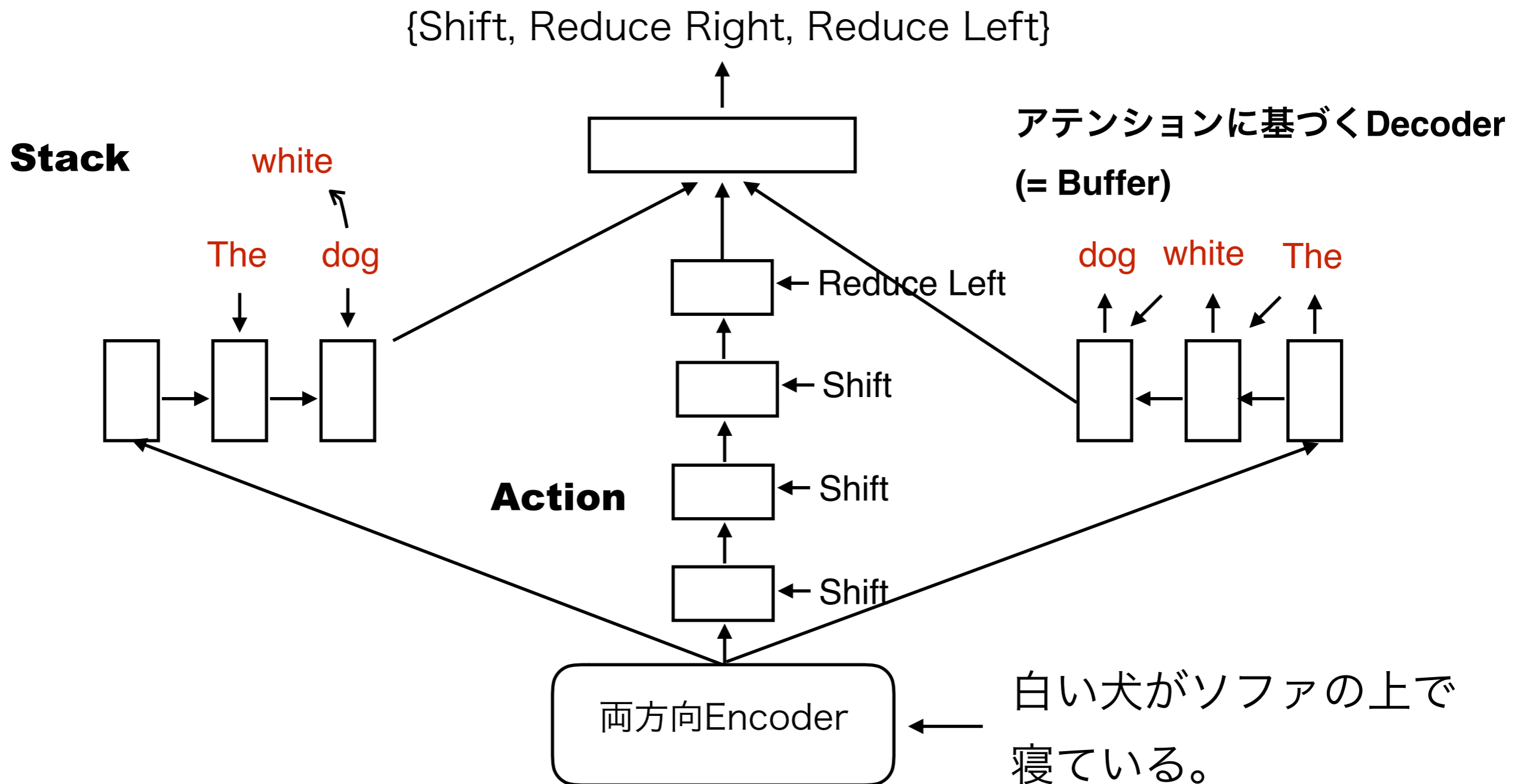
提案手法 (NMT + RNNG)

Buffer側で目的言語の単語を1つ生成し、Stackにpush



提案手法 (NMT + RNNG)

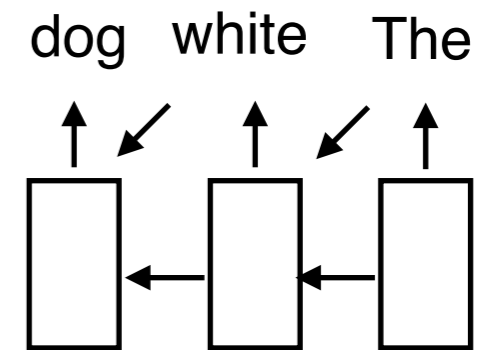
- 予備実験にて、パラメータ共有による性能改善を確認



テスト時

- RNNGモデルは選択的に使用可能
 - 翻訳の学習データ: 1M~, 構文解析の学習データ: 10K~
 - 翻訳でのアノテーションは非現実的
- 用いない場合、NMTモデルと同一速度
 - 系列データの生成のためビーム探索可

アテンションに基づくDecoder



両方向Encoder

白い犬がソファの上で
寝ている。

実験設定

- {JP, RU, CS, DE}-ENの4言語組で翻訳性能を比較
 - JP-EN: ASPEC (科学技術論文系) ([Nakazawa et al., 2016](#))
 - {RU, CS, DE}-EN: News Commentary v8 (ニュース)
 - 係り受け構造解析器: SyntaxNet ([Andor et al., 2016](#))
 - ラベル付きアクションを利用

言語組	Train	Dev.	Test	語彙 (src, tgt, action)
JP-EN	100K	1,790	1,812	(23K, 28K, 80)
RU-EN	131K	2,818 (newstest2015)	2,999 (newstest2016)	(32K, 27K, 82)
CS-EN	134K	2,656 (newstest2015)	2,998 (newstest2016)	(33K, 27K, 82)
DE-EN	166K	2,169 (newstest2015)	2,999 (newstest2016)	(33K, 30K, 80)

実験設定

- 翻訳モデルのパラメータ設定
 - パラメータ初期化: 範囲 $[-0.1, 0.1]$ の一様分布からランダム初期化
 - LSTM, Stack-LSTMの隠れ層, 単語埋め込みベクトル: 256次元
 - アクション埋め込みベクトル: 128次元
 - 目的関数: $J(\theta) = \sum \log p(\mathbf{y}, \mathbf{a} | \mathbf{x})$ ((\mathbf{x}, \mathbf{y}) は翻訳ペア, \mathbf{a} はアクション)
 - 確率的勾配降下法によるパラメータ最適化
 - BlackOut (Ji et al., 2016) による単語softmax学習の高速化
 - テスト時はビーム探索による翻訳文生成
 - 最適ビーム幅は開発データに従って決定

実験結果

- 自動評価指標による性能評価
 - BLEU (Papineni et al., 2002), RIBES (Isozaki et al., 2010)
 - ボールド字は、Bootstrapリサンプリング (Koehn, 2004) による有意差有 ($p < 0.005$)

BLEU

	JP-EN	RU-EN	CS-EN	DE-EN
NMT	17.88	12.03	11.22	16.61
NMT + RNNG	18.84	12.46	12.06	16.41

RIBES

	JP-EN	RU-EN	CS-EN	DE-EN
NMT	71.27	69.56	69.59	73.75
NMT + RNNG	72.25	71.04	70.39	75.03

考察

- RNNGモデルの構成要素をそれぞれ切除してモデルを学習
 - 構成要素のうち、Stackが最も重要
 - 構文解析タスクでは、StackのみからなるRNNGモデルが最高精度 (Kuncoro et al., 2017)

	JP-EN (Dev.)
NMT + RNNG	18.60
w/o Buffer	18.02
w/o Action	17.94
w/o Stack	17.58
NMT	17.75

翻訳例

- (発表日までに追加)

まとめ

- 目的言語側の構文情報を利用したNMTモデルの提案
 - NMTモデルと、RNNGモデル (遷移型構文解析と文生成モデル) を1つのニューラルネットワークで表現
- 4言語組の翻訳実験で、提案モデルによる性能改善を確認
 - RNNGモデルの3つの構成要素のうち、構文木を構成するStackが性能改善に寄与
- 今後の課題は、提案モデルに対して、外部構文解析結果が与える影響に関する調査