NAACL2018 参加報告

東京工業大学
上垣外 英剛

自己紹介

上垣外 英剛 （かみがいと ひでたか）

- 東工大高村研で博士取得 (2017/3)
- 昨年までNTTコミュニケーション科学基礎研究所でポスドク (2017/4~2018/3)
- 現在東工大奥村研助教 (2018/4~)

今まで行ってきた研究
機械翻訳・自動要約・句構造解析

自己紹介

上垣外 英剛 （かみがいと ひでたか）

- 東工大高村研で博士取得 (2017/3)
- 昨年までNTTコミュニケーション科学基礎研究所でポスドク (2017/4~2018/3)
- 現在東工大奥村研助教 (2018/4~)

NAACL2018では下記論文を発表

Hidetaka Kamigaito, Katsuhiko Hayashi, Hirao Tsutomu and Masaaki Nagata.
“**Higher-order Syntactic Attention Network for Long Sentence Compression**” (long paper)

Annual Conference of the North American Chapter of the Association for Computational Linguistics

略称 **NAACL**

北米で開催される**自然言語処理分野の**
トップ国際会議の一つ

他にはACLやEMNLP等が存在

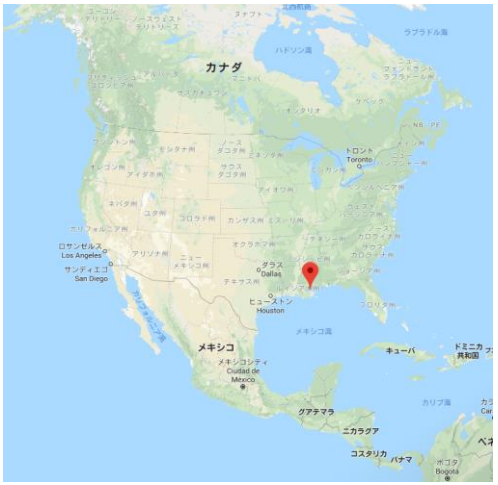
NAACL2018 (6/1~6/6)

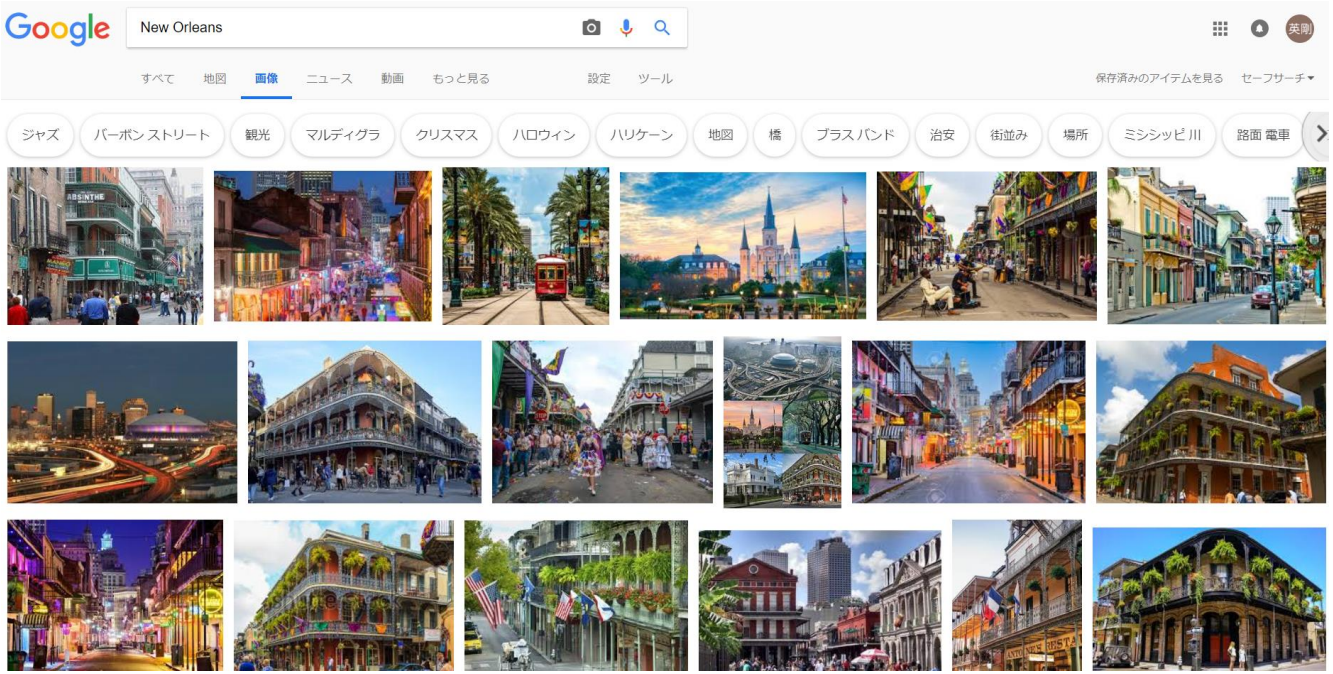
16回目となる**NAACL**

NAACL2018

16回目となる**NAACL**

開催地は**ニューオーリンズ**

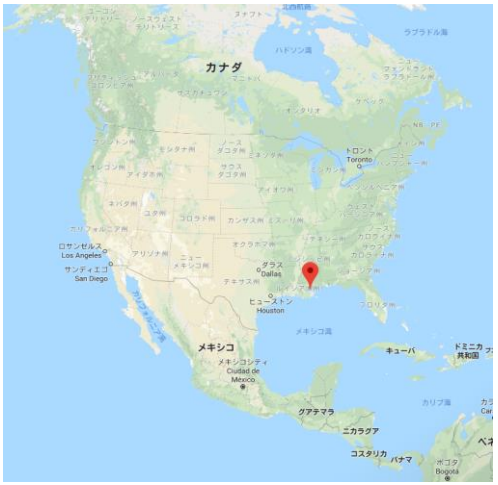




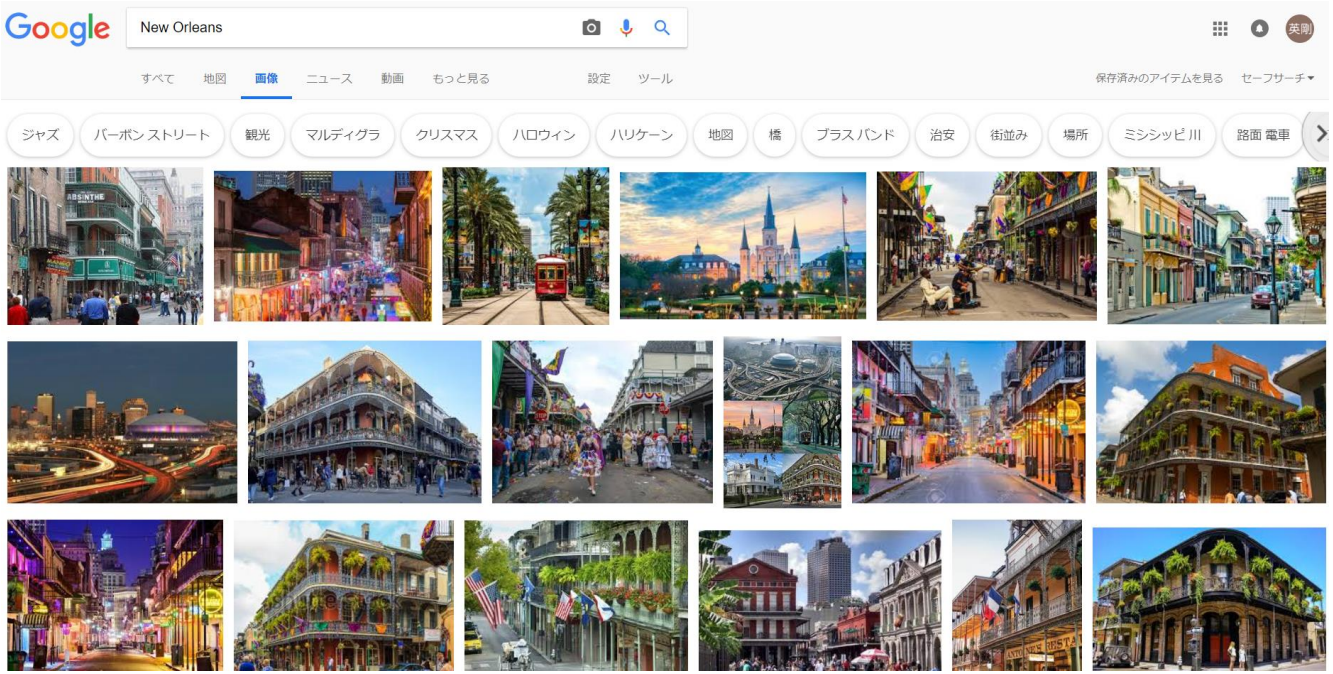
NAACL2018

16回目となる**NAACL**

開催地は**ニューオーリンズ**



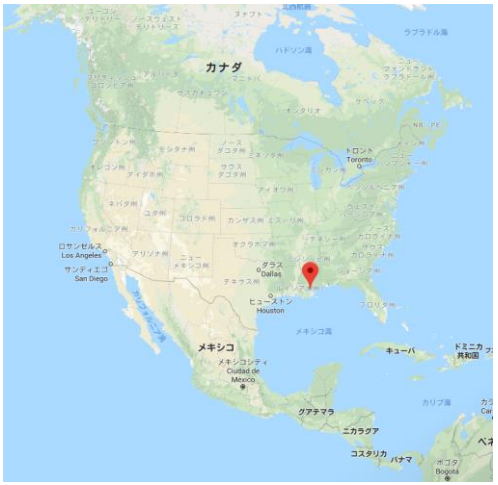
歓楽街



NAACL2018

16回目となる**NAACL**

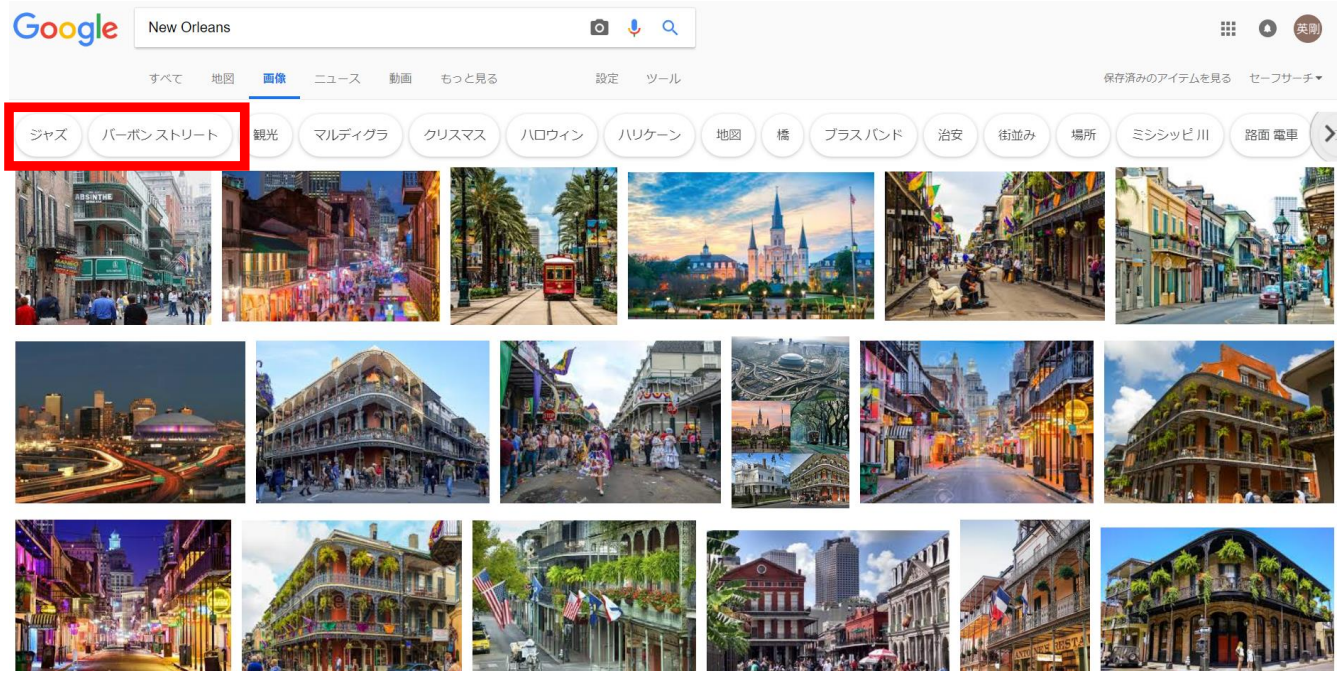
開催地は**ニューオーリンズ**



歓楽街

音楽

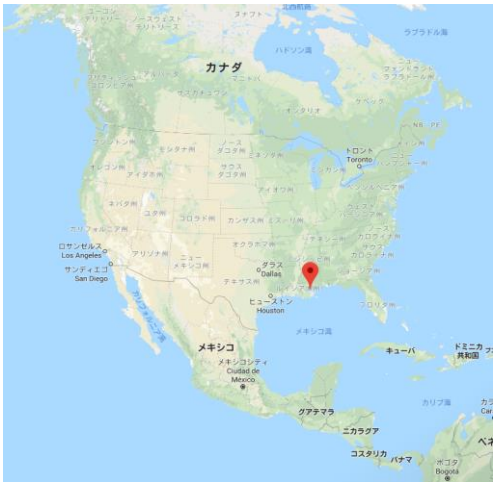
お酒



NAACL2018

16回目となる**NAACL**

開催地は**ニューオーリンズ**



海産物

NAACL2018

16回目となる**NAACL**

開催地は**ニューオーリンズ**

会場は**Hyatt Regency New Orleans**

NAACL2018

16回目となる**NAACL**

開催地は**ニューオーリンズ**

会場は**Hyatt Regency New Orleans**

会場の朝食

投稿数・採択率

Long Paper	投稿数：**647**	採択数：**207**	採択率：**32%**
Short Paper	投稿数：**425**	採択数：**125**	採択率：**29%**
合計	投稿数：**1072**	採択数：**332**	採択率：**31%**

投稿数・採択率

Long Paper	投稿数：**647**	採択数：**207**	採択率：**32%**
Short Paper	投稿数：**425**	採択数：**125**	採択率：**29%**
合計	投稿数：**1072**	採択数：**332**	採択率：**31%**

参考　NAACL2016

Long Paper	投稿数：**396**	採択数：**100**	採択率：**25%**
Short Paper	投稿数：**284**	採択数：**82**	採択率：**29%**
合計	投稿数：**680**	採択数：**182**	採択率：**27%**

投稿数・採択率

Long Paper	投稿数：**647**	採択数：**207**	採択率：**32%**
Short Paper	投稿数：**425**	採択数：**125**	採択率：**29%**
合計	投稿数：**1072**	採択数：**332**	採択率：**31%**

投稿数は約1.5倍に！

参考　NAACL2016

Long Paper	投稿数：**396**	採択数：**100**	採択率：**25%**
Short Paper	投稿数：**284**	採択数：**82**	採択率：**29%**
合計	投稿数：**680**	採択数：**182**	採択率：**27%**

投稿数・採択率

Long Paper	投稿数：**647**	採択数：**207**	採択率：**32%**
Short Paper	投稿数：**425**	採択数：**125**	採択率：**29%**
合計	投稿数：**1072**	採択数：**332**	採択率：**31%**

採択率は上昇

参考　NAACL2016

Long Paper	投稿数：**396**	採択数：**100**	採択率：**25%**
Short Paper	投稿数：**284**	採択数：**82**	採択率：**29%**
合計	投稿数：**680**	採択数：**182**	採択率：**27%**

NAACL2018: Outstanding Paper

タスク・手法ともに多様
実用性が高いELMoがBest Paperに選ばれる

- ***Deep Contextualized Word Representations.*** Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee and Luke Zettlemoyer Best Paper
- ***Learning to Map Context-Dependent Sentences to Executable Formal Queries***. Alane Suhr, Srinivasan Iyer and Yoav Artz
- ***Neural Text Generation in Stories using Entity Representations as Context***. Elizabeth Clark, Yangfeng Ji and Noah A. Smith
- ***Recurrent Neural Networks as Weighted Language Recognizers***. Yining Chen, Sorcha Gilroy, Andreas Maletti, Jonathan May and Kevin Knight

NAACL2018以降で感じた最近の自然言語処理（手法）

- 分散表現 → 定着
- ニューラル → 定着
- RNN → 定着
- Seq2Seq → 定着
- End-to-End → 定着
- 強化学習 → 成長中
- 伝統的な手法 → 復活中？

特性を調べる等の研究が増加

複雑なモデルが増加

省リソース設定・データの作成などの研究が増加

複数のタスクで使用

探索・枝刈り・クラスタリング等

NAACL2018以降で感じた最近の自然言語処理（手法）

- 分散表現 → 定着
- ニューラル → 定着
- RNN → 定着
- Seq2Seq → 定着
- End-to-End → 定着
- 強化学習 → 成長中
- 伝統的な手法 → 復活中？

特性を調べる等の研究が増加

複雑なモデルが増加

省リソース設定・データの作成などの研究が増加

複数のタスクで使用

探索・枝刈り・クラスタリング等

NAACL2018以降で感じた最近の自然言語処理（手法）

分散表現 → 定着

ニューラル → 定着

RNN → 定着

Seq2Seq → 定着

End-to-End → 定着

強化学習 → 成長中

伝

今回の紹介対象

特性を調べる等の研究が増加

複雑なモデルが増加

省リソース設定・データの作成などの研究が増加

複数のタスクで使用

探索・枝刈り・クラスタリング等

今回紹介する論文

- **Deep contextualized word representations**
  - Best paper
  - 大規模コーパスによる精度向上を複数タスクで実現

- **Recurrent Neural Networks as Weighted Language Recognizers**
  - Outstanding paper
  - RNNの理論限界を明らかに

- **Fortification of Neural Morphological Segmentation Models for Polysynthetic Minimal-Resource Languages**
  - 小規模コーパスで学習するための手法を検証

Deep contextualized word representations

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer

概要

- 単語ベクトルで文脈情報を考慮可能な事前学習手法**ELMo**を提案
  - **ELMo**: Embeddings from Language Models
  - 両方向のLSTM言語モデルを使用

- 自然言語処理の複数タスクで大幅な性能の改善を報告

複数タスクで最高精度を達成

ELMo未使用

Task	Previous SOTA		Our baseline	ELMo + Baseline	Increase (Absolute/Relative)
SQuAD	SAN	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al (2017)	88.6	88.0	88.7 +/- 0.17	0.7 / 5.8%
SRL	He et al (2017)	81.7	81.4	84.6	3.2 / 17.2%
Coref	Lee et al (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al (2017)	91.93 +/- 0.19	90.15	92.22 +/- 0.10	2.06 / 21%
Sentiment (5-class)	McCann et al (2017)	53.7	51.4	54.7 +/- 0.5	3.3 / 6.8%

https://allennlp.org/elmo より引用

複数タスクで最高精度を達成

ELMo未使用　　ELMo使用

Task	Previous SOTA		Our baseline	ELMo + Baseline	Increase (Absolute/Relative)
SQuAD	SAN	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al (2017)	88.6	88.0	88.7 +/- 0.17	0.7 / 5.8%
SRL	He et al (2017)	81.7	81.4	84.6	3.2 / 17.2%
Coref	Lee et al (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al (2017)	91.93 +/- 0.19	90.15	92.22 +/- 0.10	2.06 / 21%
Sentiment (5-class)	McCann et al (2017)	53.7	51.4	54.7 +/- 0.5	3.3 / 6.8%

複数タスクで最高精度を達成

Task	Previous SOTA		Our baseline	ELMo + Baseline	Increase (Absolute/Relative)
SQuAD	SAN	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al				0.7 / 5.8%
SRL					3.2 / 17.2%
Coref	Lee et al (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al (2017)	91.93 +/- 0.19	90.15	92.22 +/- 0.10	2.06 / 21%
Sentiment (5-class)	McCann et al (2017)	53.7	51.4	54.7 +/- 0.5	3.3 / 6.8%

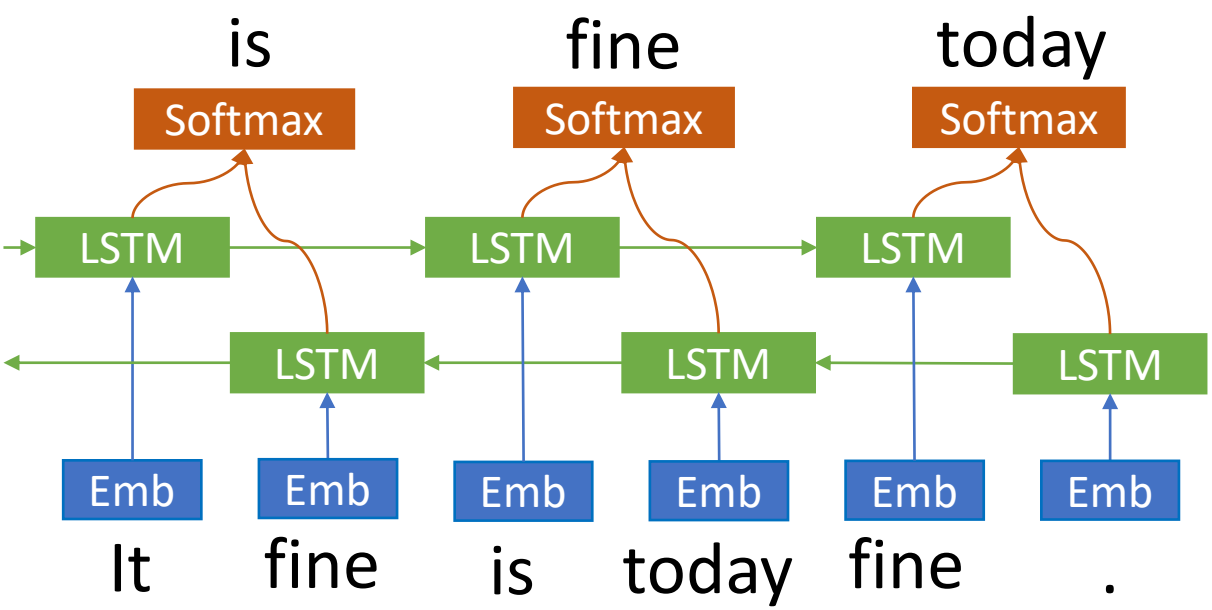
大幅な精度向上を達成

ELMoの構成

基本的には両方向のLSTM言語モデル

目的関数

$$\sum_{k=1}^{N} \left(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \right)$$



順方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

逆方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

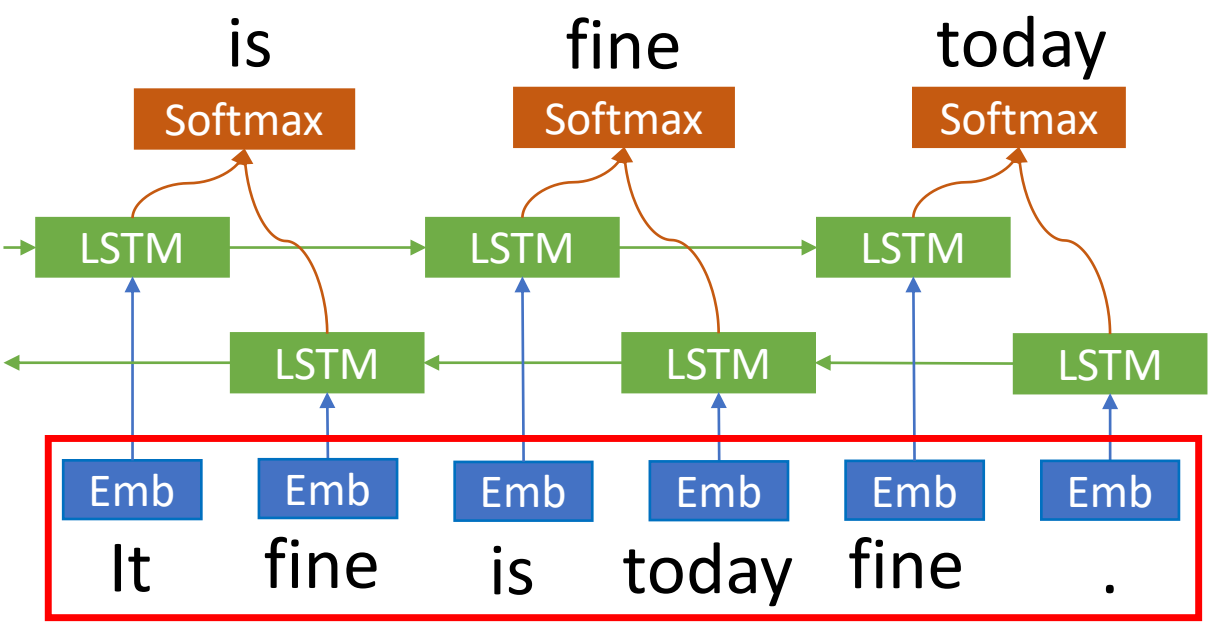
ELMoの構成

基本的には両方向のLSTM言語モデル

未知語に対処するため
単語ベクトルは文字CNNで学習

目的関数

$$\sum_{k=1}^{N} \left(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \right)$$



順方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

逆方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

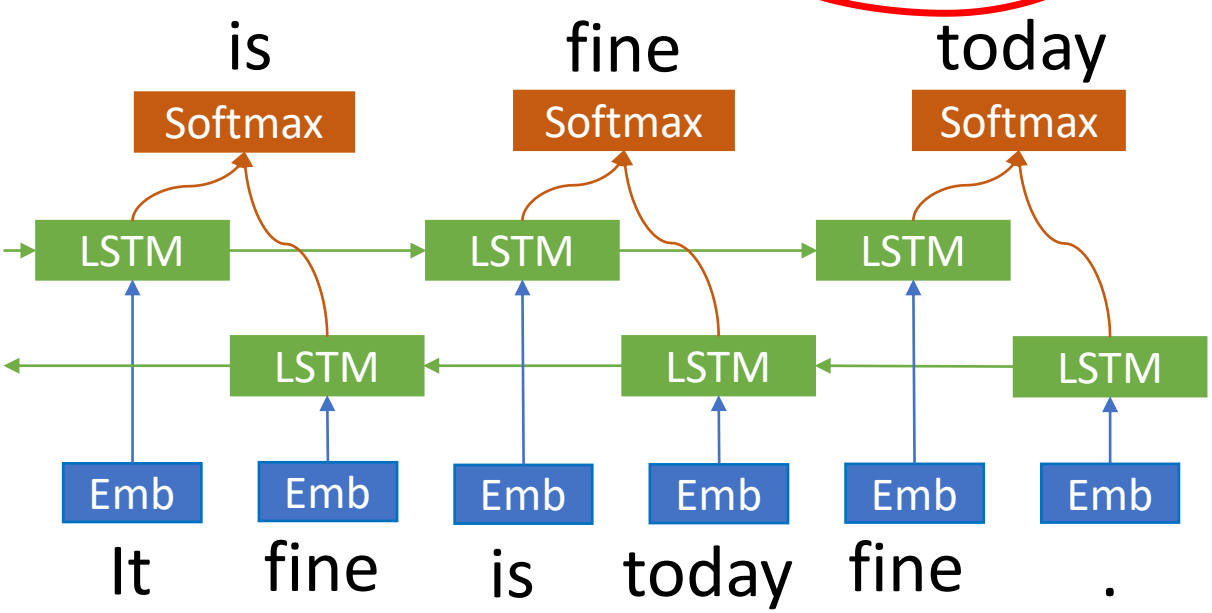
ELMoの構成

基本的には両方向のLSTM言語モデル

目的関数

$$\sum_{k=1}^{N} \Big(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \Big)$$

両方向モデルでパラメータをシェア



順方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

逆方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

ELMoの構成

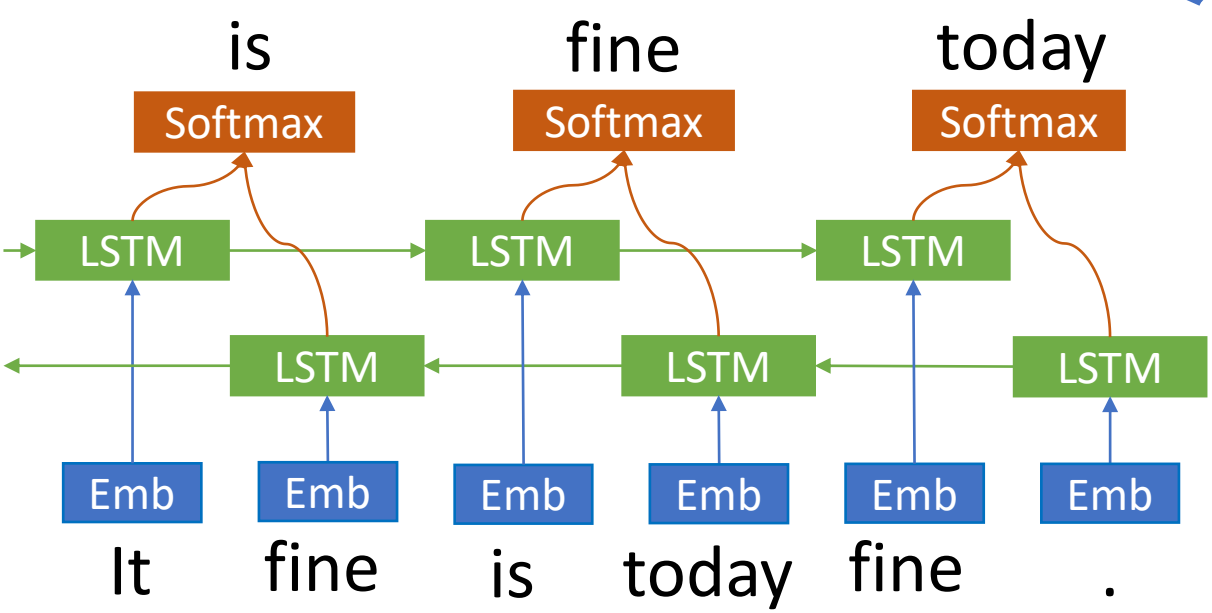
基本的には両方向のLSTM言語モデル

目的関数

$$\sum_{k=1}^{N} \Big(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \boxed{\overrightarrow{\Theta}_{LSTM}}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \boxed{\overleftarrow{\Theta}_{LSTM}}, \Theta_s) \Big)$$

両方向モデルで
パラメータをシェア

LSTMはそれぞれ別



順方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

逆方向言語モデル

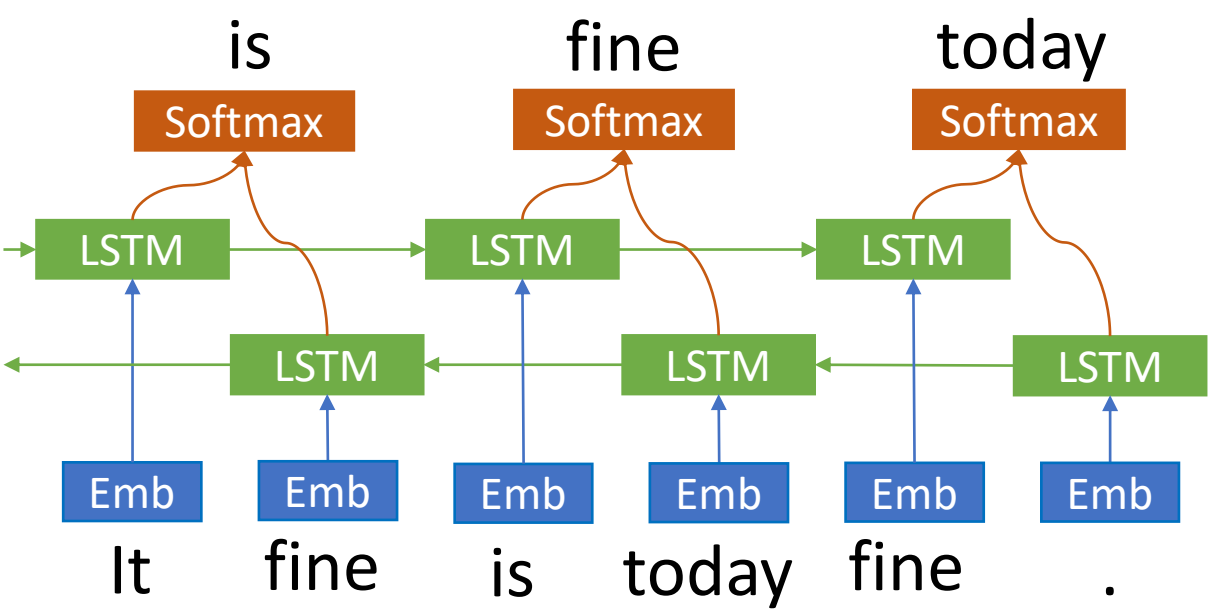
$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

ELMoの学習

言語モデルなので**ラベル付きデータは不要**

目的関数

$$\sum_{k=1}^{N} \left(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \right)$$



順方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

逆方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

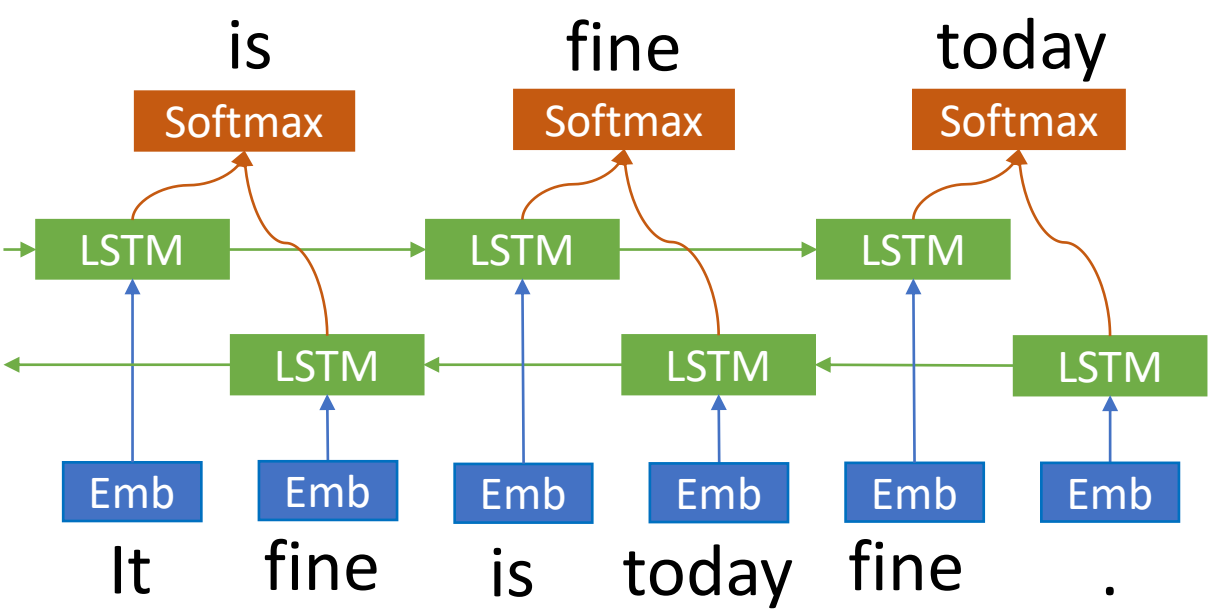
ELMoの学習

言語モデルなのでラベル付きデータは不要

目的関数

$$\sum_{k=1}^{N} \left(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \right)$$

大規模データを利用可能



順方向言語モデル

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

逆方向言語モデル

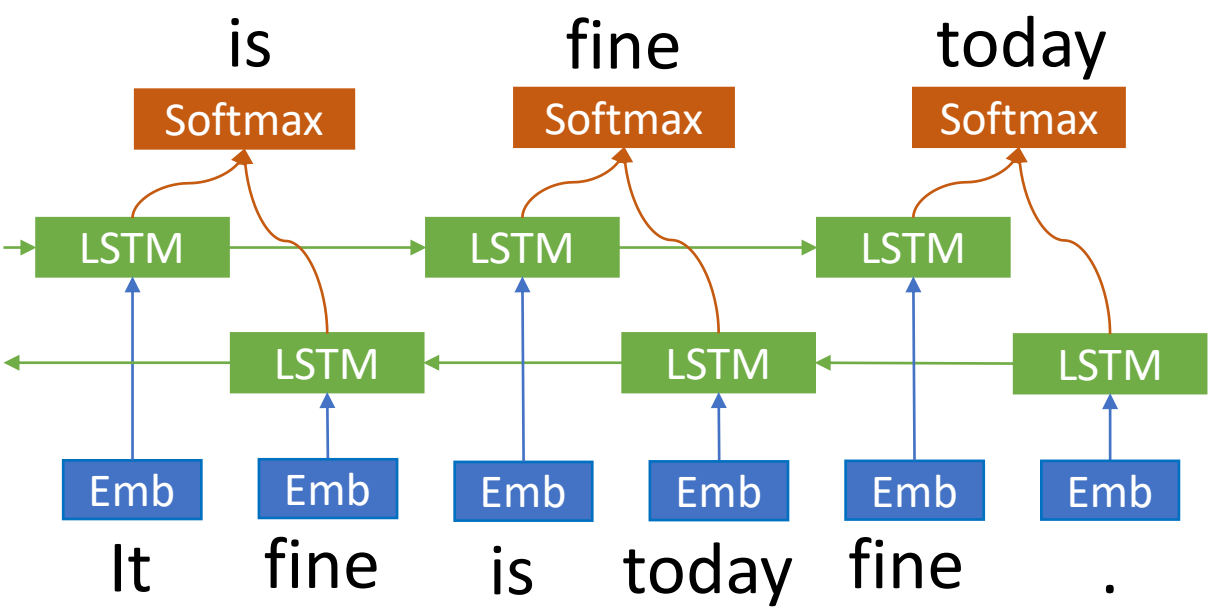
$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N)$$

ELMoの利用

単語ベクトルと対応するLSTMの隠れ層を
重み付け加算して利用

$$\mathbf{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} \mathbf{h}_{k,j}^{LM}$$

$$\mathbf{h}_{k,j}^{LM} = [\overrightarrow{\mathbf{h}}_{k,j}^{LM}; \overleftarrow{\mathbf{h}}_{k,j}^{LM}]$$



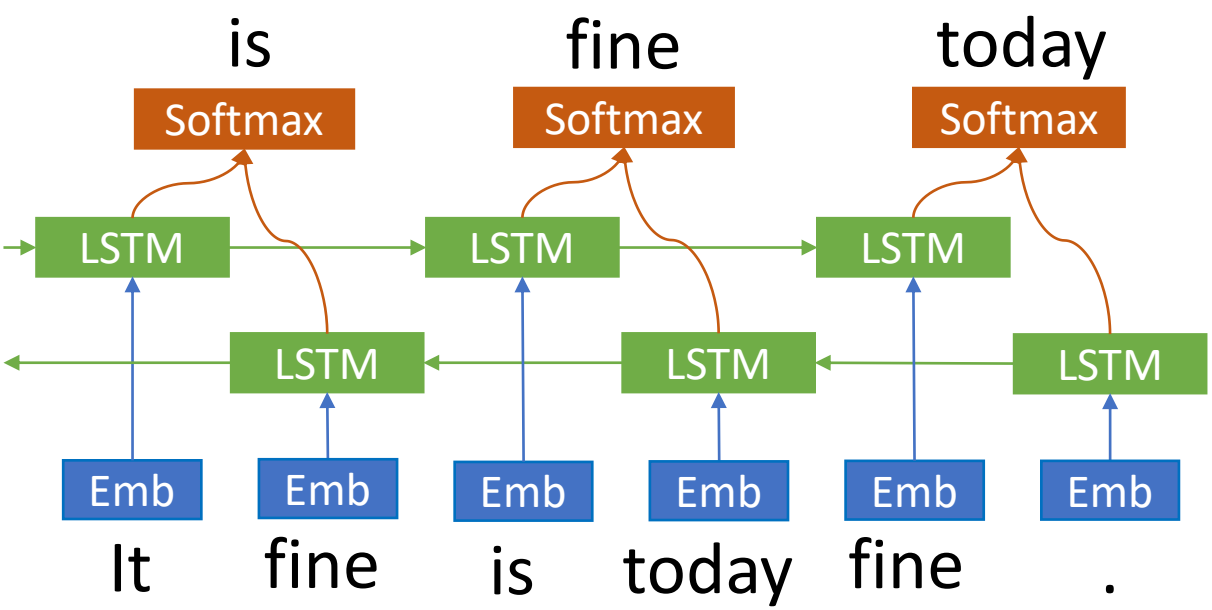
ELMoの利用

単語ベクトルと対応するLSTMの隠れ層を
重み付け加算して利用

$$\mathbf{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} \mathbf{h}_{k,j}^{LM}$$

$$\mathbf{h}_{k,j}^{LM} = [\overrightarrow{\mathbf{h}}_{k,j}^{LM}; \overleftarrow{\mathbf{h}}_{k,j}^{LM}]$$

単語ベクトルも含む



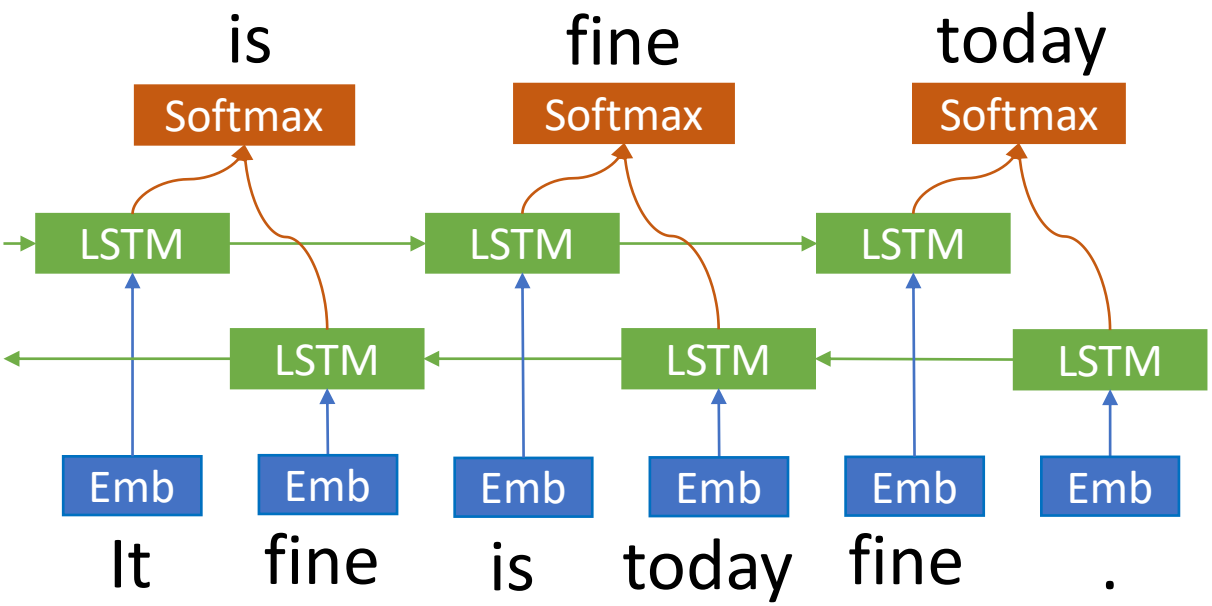
ELMoの利用

単語ベクトルと対応するLSTMの隠れ層を
重み付け加算して利用

$$\mathbf{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \boxed{\gamma^{task}} \sum_{j=0}^{L} s_j^{task} \mathbf{h}_{k,j}^{LM}$$

$$\mathbf{h}_{k,j}^{LM} = [\overrightarrow{\mathbf{h}}_{k,j}^{LM}; \overleftarrow{\mathbf{h}}_{k,j}^{LM}]$$

ハイパーパラメータ
タスクに応じて調整



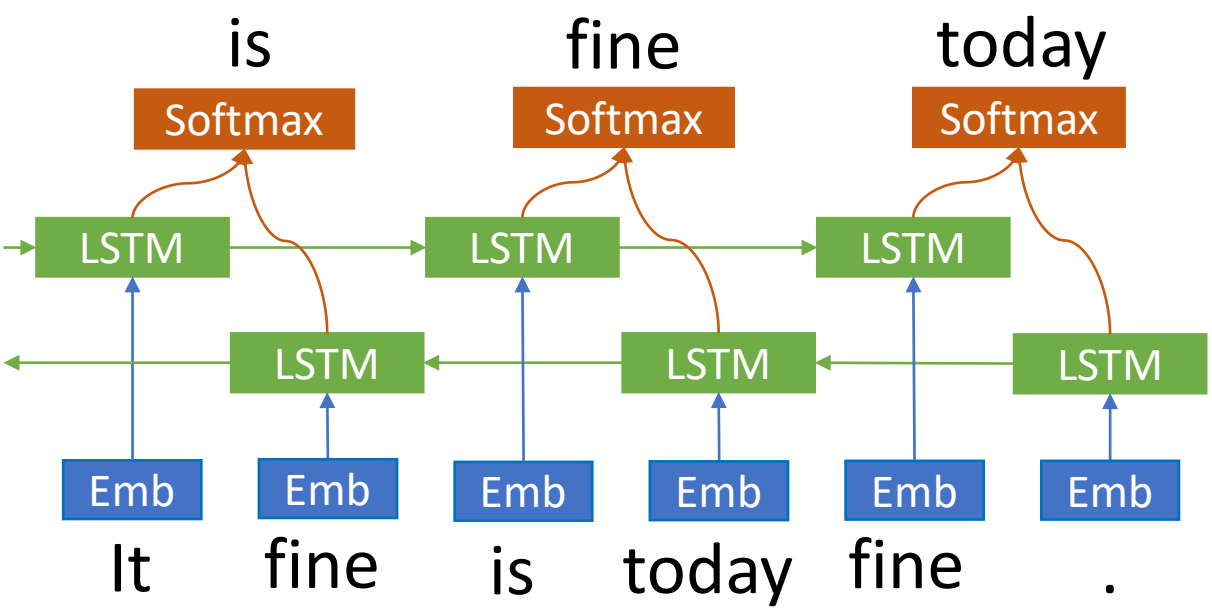
ELMoの利用

単語ベクトルと対応するLSTMの隠れ層を
重み付け加算して利用

$$\mathbf{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} \boxed{s_j^{task}} \mathbf{h}_{k,j}^{LM}$$

$$\mathbf{h}_{k,j}^{LM} = [\overrightarrow{\mathbf{h}}_{k,j}^{LM}; \overleftarrow{\mathbf{h}}_{k,j}^{LM}]$$

Softmax層で重み付け
自動調整



ELMoの利用

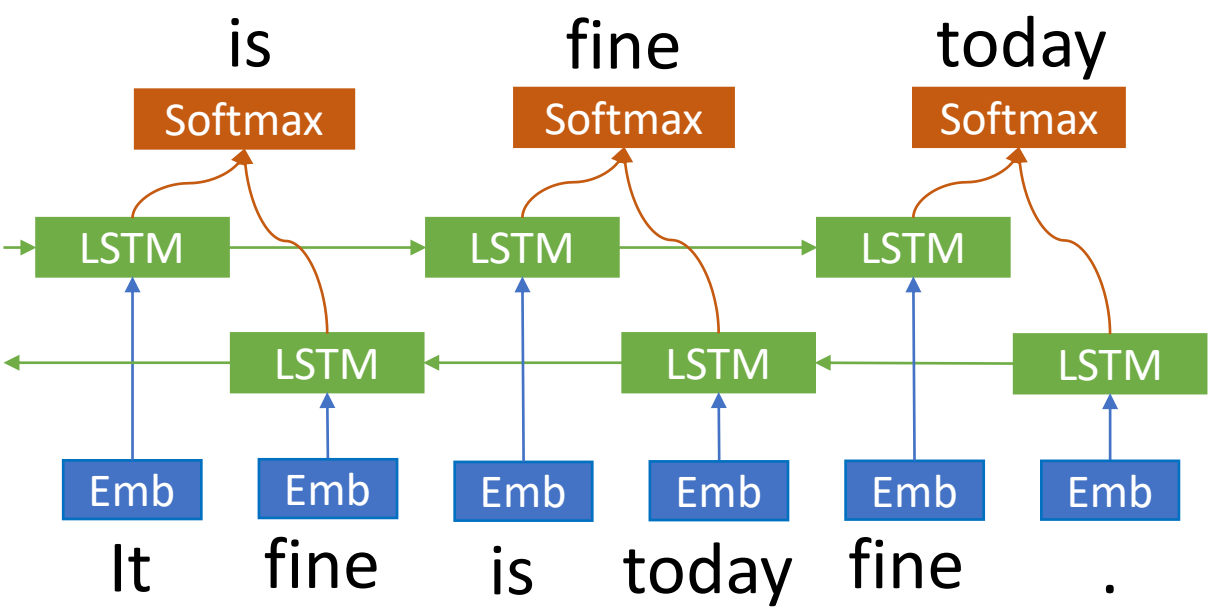
ELMoの単語表現ベクトルはRNNに基づくモデルの入力や隠れ層に結合して利用

$$\mathbf{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} \mathbf{h}_{k,j}^{LM}$$

$$\mathbf{h}_{k,j}^{LM} = [\overrightarrow{\mathbf{h}}_{k,j}^{LM}; \overleftarrow{\mathbf{h}}_{k,j}^{LM}]$$

$$[\bar{\mathbf{h}}_k; \bar{\mathbf{ELMo}}_k^{task}]$$

$$[\mathbf{x}_k; \bar{\mathbf{ELMo}}_k^{task}]$$



再掲：複数タスクで最高精度

Task	Previous SOTA		Our baseline	ELMo + Baseline	Increase (Absolute/Relative)
SQuAD	SAN	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al (2017)	88.6	88.0	88.7 +/- 0.17	0.7 / 5.8%
SRL	He et al (2017)	81.7	81.4	84.6	3.2 / 17.2%
Coref	Lee et al (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al (2017)	91.93 +/- 0.19	90.15	92.22 +/- 0.10	2.06 / 21%
Sentiment (5-class)	McCann et al (2017)	53.7	51.4	54.7 +/- 0.5	3.3 / 6.8%

モデルとコードは公開中

https://github.com/allenai/bilm-tf

bilm-tf

Tensorflow implementation of the pretrained biLM used to compute ELMo representations from "Deep contextualized word representations".

This repository supports both training biLMs and using pre-trained models for prediction.

We also have a pytorch implementation available in AllenNLP.

You may also find it easier to use the version provided in Tensorflow Hub if you just like to make predictions.

Citation:

```
@inproceedings{Peters:2018,
  author={Peters, Matthew E. and  Neumann, Mark and Iyyer, Mohit and Gardner, Matt and Clark, Christopher and Lee, Ken
  title={Deep contextualized word representations},
  booktitle={Proc. of NAACL},
  year={2018}
}
```

Installing

Install python version 3.5 or later, tensorflow version 1.2 and h5py:

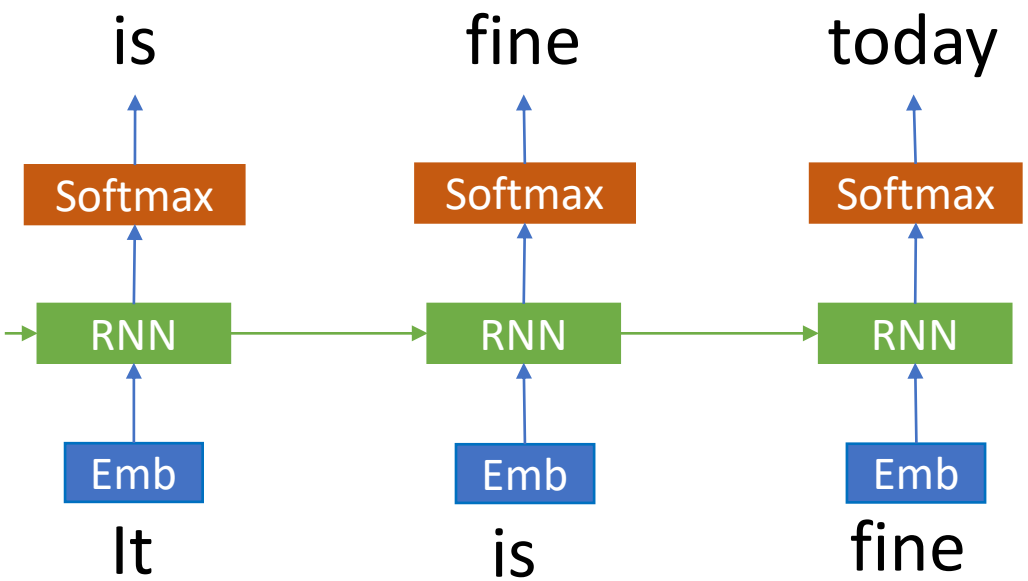
入手は容易→今年度からの標準的な事前学習法になる？

Recurrent Neural Networks as Weighted Language Recognizers

Yining Chen, Sorcha Gilroy, Andreas Maletti, Jonathan May, Kevin Knight

RNNの特性を証明

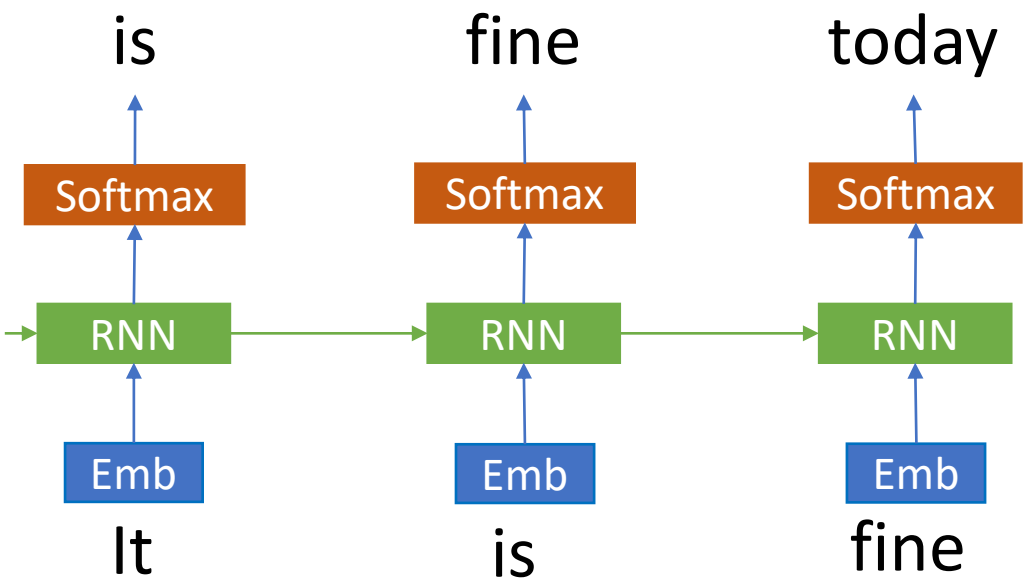
1-layerでReLUとSoftmax層から構成されるRNN：



を対象にRNNの特性を証明

RNNの特性を証明

1-layerでReLUとSoftmax層から構成されるRNN：



を対象にRNNの特性を証明

本発表では証明された事実のみを紹介

証明された特性

1-layerのRNNを確率オートマトンと考え理論限界を調査
以下を明らかに（LSTMでも成立すると考えられる）

1. 導出規則の総和が1にならない、一貫性を持たないRNNが存在
2. このRNNの一貫性は保証できない
3. 任意のRNNにおいて、最も高い重みを持つ文字列を探し出すことは非決定的

証明された特性

1-layerのRNNを確率オートマトンと考え理論限界を調査
以下を明らかに（LSTMでも成立すると考えられる）

1. 導出規則の総和が1にならない、一貫性を持たないRNNが存在
2. このRNNの一貫性は保証できない
3. 任意のRNNにおいて、最も高い重みを持つ文字列を探し出すことは非決定的

任意のRNNを用いて文字列を生成する場合には
近似解法や**ヒューリスティックな手法**が**必要**

証明された特性

1-layerのRNNを確率オートマトンと考え理論限界を調査
以下を明らかに（LSTMでも成立すると考えられる）

1. 導出規則の総和が1にならない、一貫性を持たないRNNが存在
2. このRNNの一貫性は保証できない
3. 任意のRNNにおいて、最も高い重みを持つ文字列を探し出すことは非決定的

任意のRNNを用いて文字列を生成する場合には
近似解法や**ヒューリスティックな手法**が**必要**

一貫性が保証された場合は？

証明された特性

1-layerのRNNを確率オートマトンと考え理論限界を調査
以下を明らかに（LSTMでも成立すると考えられる）

1. 導出規則の総和が1にならない、一貫性を持たないRNNが存在
2. このRNNの一貫性は保証できない
3. 任意のRNNにおいて、最も高い重みを持つ文字列を探し出すことは非決定的
4. 一貫性が保証されたRNNで最も高い重みを持つ文字列を探し出すことは決定的

一貫性が保証されていればRNNで最も高い重みを持つ文字列を探し出すことは可能

証明された特性

1-layerのRNNを確率オートマトンと考え理論限界を調査
以下を明らかに（LSTMでも成立すると考えられる）

1. 導出規則の総和が1にならない、一貫性を持たないRNNが存在
2. このRNNの一貫性は保証できない
3. 任意のRNNにおいて、最も高い重みを持つ文字列を探し出すことは非決定的
4. 一貫性が保証されたRNNで最も高い重みを持つ文字列を探し出すことは決定的
5. 4.において探索する長さの計算量の上界はその他の上界を上回る
6. 解空間が多項式長に制限された場合に最も高い重みを持つ文字列を探し出すことはNP完全

一貫性が保証されていればRNNで最も高い重みを持つ文字列を探し出すことは可能**だが計算量的に困難**

証明された特性

1-layerのRNNを確率オートマトンと考え理論限界を調査
以下を明らかに（LSTMでも成立すると考えられる）

1. 導出規則の総和が1にならない、一貫性を持たないRNNが存在
2. このRNNの一貫性は保証できない
3. 任意のRNNにおいて、最も高い重みを持つ文字列を探し出すことは非決定的
4. 一貫性が保証されたRNNで最も高い重みを持つ文字列を探し出すことは決定的
5. 4.において探索する長さの計算量の上界はその他の上界を上回る
6. 解空間が多項式長に制限された場合に最も高い重みを持つ文字列を探し出すことはNP完全

さらにRNNの等価性についても証明

証明された特性

1-layerのRNNを確率オートマトンと考え理論限界を調査
以下を明らかに（LSTMでも成立すると考えられる）

1. 導出規則の総和が1にならない、一貫性を持たないRNNが存在
2. このRNNの一貫性は保証できない
3. 任意のRNNにおいて、最も高い重みを持つ文字列を探し出すことは非決定的
4. 一貫性が保証されたRNNで最も高い重みを持つ文字列を探し出すことは決定的
5. 4.において探索する長さの計算量の上界はその他の上界を上回る
6. 解空間が多項式長に制限された場合に最も高い重みを持つ文字列を探し出すことはNP完全
7. RNNの等価性（同じ文字列に同じ重みを与えること）は非決定的
8. 等価なRNNを最小のニューロンで構成することは非決定的

対象のRNNの挙動を保存しつつよりコンパクトなモデルにするには**近似解法**や**ヒューリスティックな手法**が**必要**

まとめ

以下の事実が理論的に示された

RNNを用いて文字列を生成する場合に**近似解法やヒューリスティックな手法が必要**とされる。

対象のRNNの挙動を保存しつつよりコンパクトなモデルにするには**近似解法**や**ヒューリスティックな手法**が**必要**

まとめ

以下の事実が理論的に示された

RNNを用いて文字列を生成する場合に**近似解法やヒューリスティックな手法が必要**とされる。

対象のRNNの挙動を保存しつつよりコンパクトなモデルにするには**近似解法**や**ヒューリスティックな手法**が**必要**

今後もデコーダー周りの探索法の研究は続く？

Fortification of Neural Morphological Segmentation Models for Polysynthetic Minimal-Resource Languages

Katharina Kann, Manuel Mager, Ivan Meza-Ruiz, Hinrich Schütze

抱合語

長い**複合動詞**を多く含む言語

持ち上げる

持ち｜上げる

日本語の複合動詞の例

抱合語

長い**複合動詞**を多く含む言語

持ち上げる

持ち｜上げる

日本語の複合動詞の例

onemococowaya

o｜ne｜mo｜cocowa｜ya

抱合語の複合動詞の例

対象とする抱合語

本論文では4つの言語を扱う
下記の例はそれぞれの言語での *I was sick* の表現

Mexicanero

u|ni|ye MALO

スペイン語の語彙が含まれる

Wixarica

ne|p+|ti|kuye|kai

長音・短音の母音を含む

語順はSOV

Nahuatl

o|ne|mo|cocowa|ya

語順はSOV

Yorem Nokki

ko’kore|ye|ne

複雑な動詞が多い

抱合語の形態素解析

抱合語の形態素解析データセットは少数しか存在しない

	Mexicanero	Nahuatl	Wixarika	Yorem N.
train	427	540	665	511
dev	106	134	176	127
test	355	449	553	425
total	888	1123	1394	1063

データ分割と含まれる文数の内訳

抱合語の形態素解析

抱合語の形態素解析データセットは少数しか存在しない

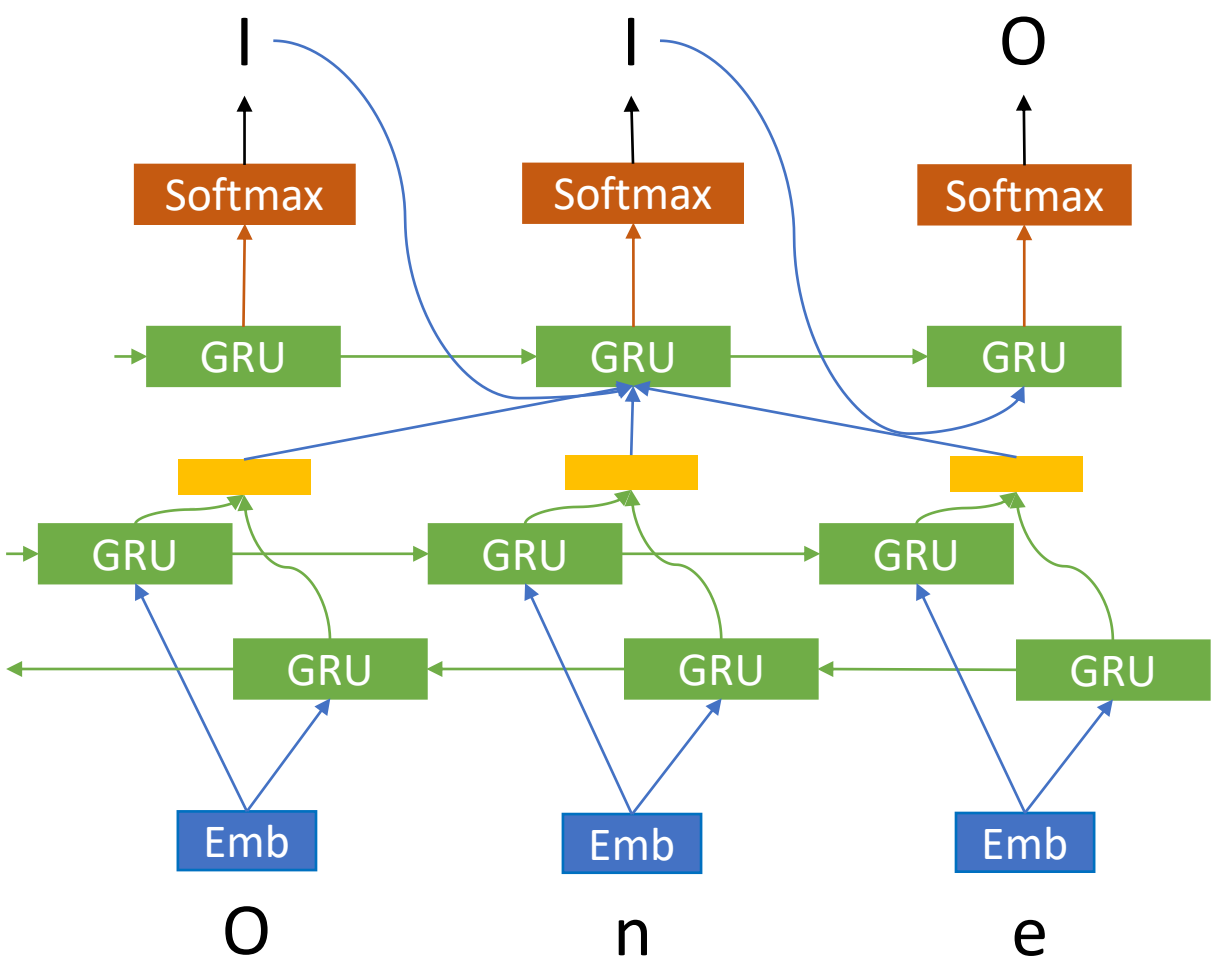
	Mexicanero	Nahuatl	Wixarika	Yorem N.
train	427	540	665	511
dev	106	134	176	127
test	355	449	553	425
total	888	1123	1394	1063

データ分割と含まれる文数の内訳

少資源コーパスでも学習可能な手法を提案

使用するモデル

GRUを用いたEncoder-Decoder



出力ラベル列

Decoder
w/ Attention

Encoder

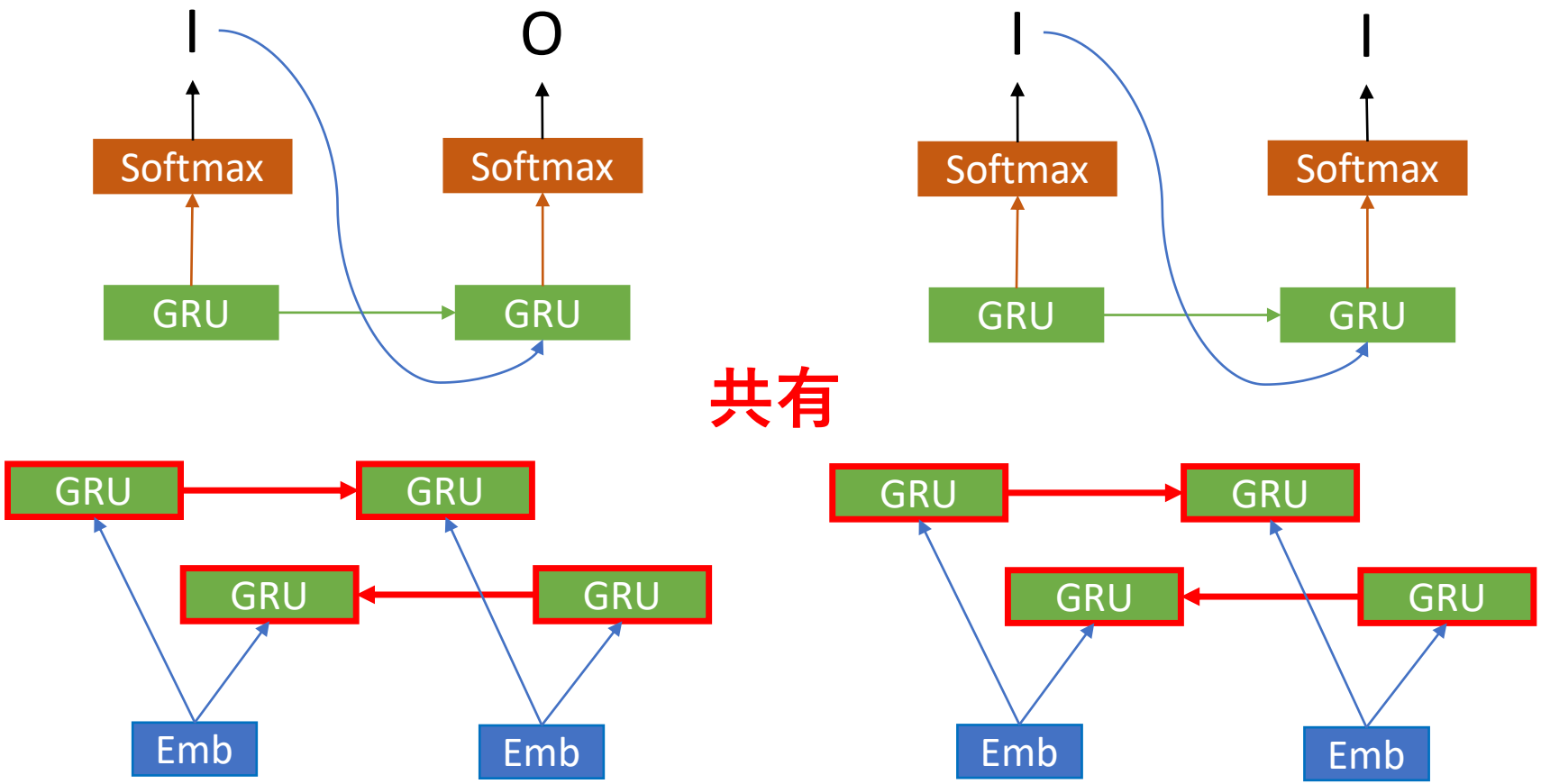
入力文字列

手法1：マルチタスクラーニング

教師ありと**教師なし**の二種類の手法を提案

マルチタスクラーニング

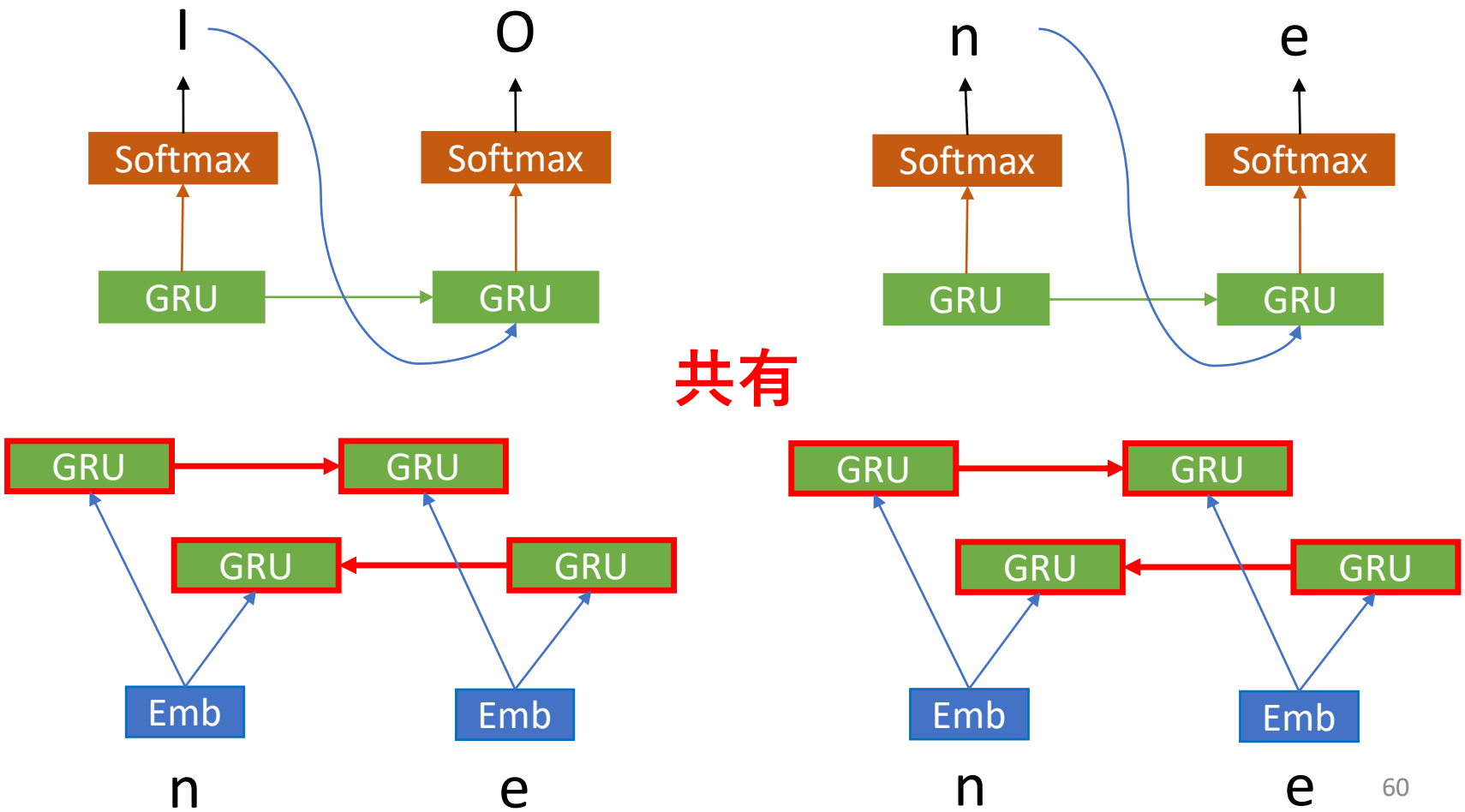
教師あり：同一エンコーダで複数言語を学習



Mexicanero **Wixarica**

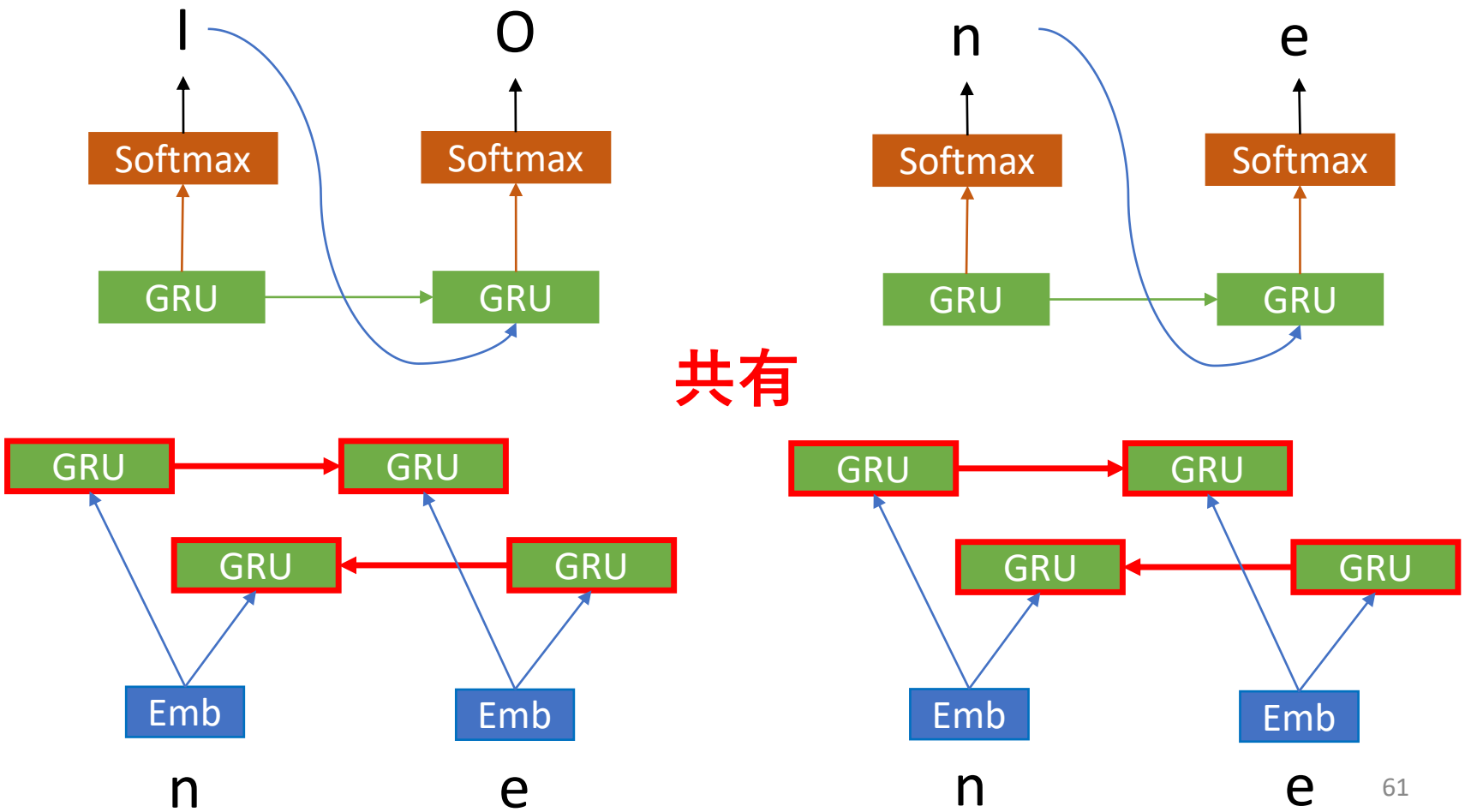
マルチタスクラーニング

教師なし：同一エンコーダで疑似正解と同時に学習



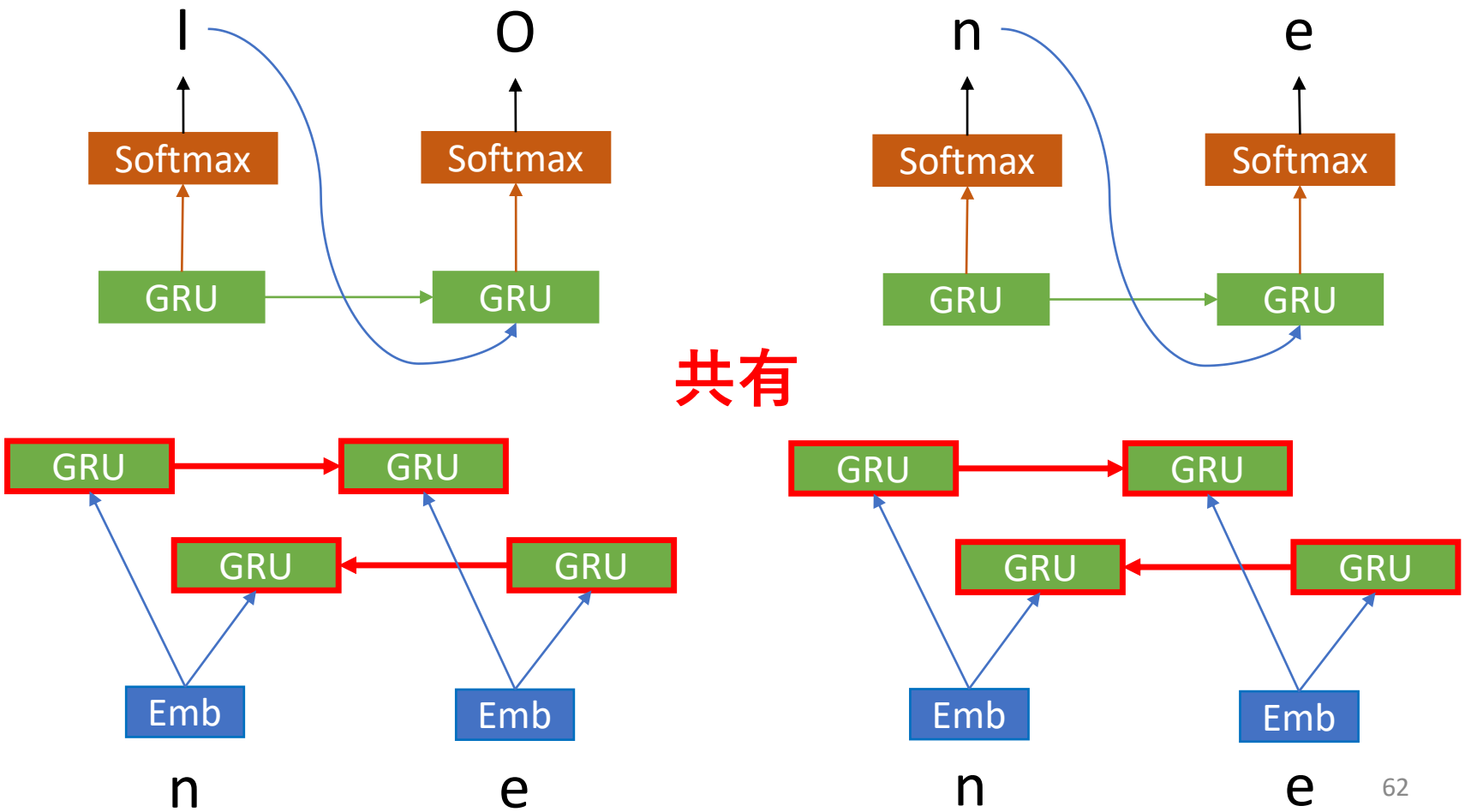
マルチタスクラーニング

教師なし：同一エンコーダで疑似正解と同時に学習



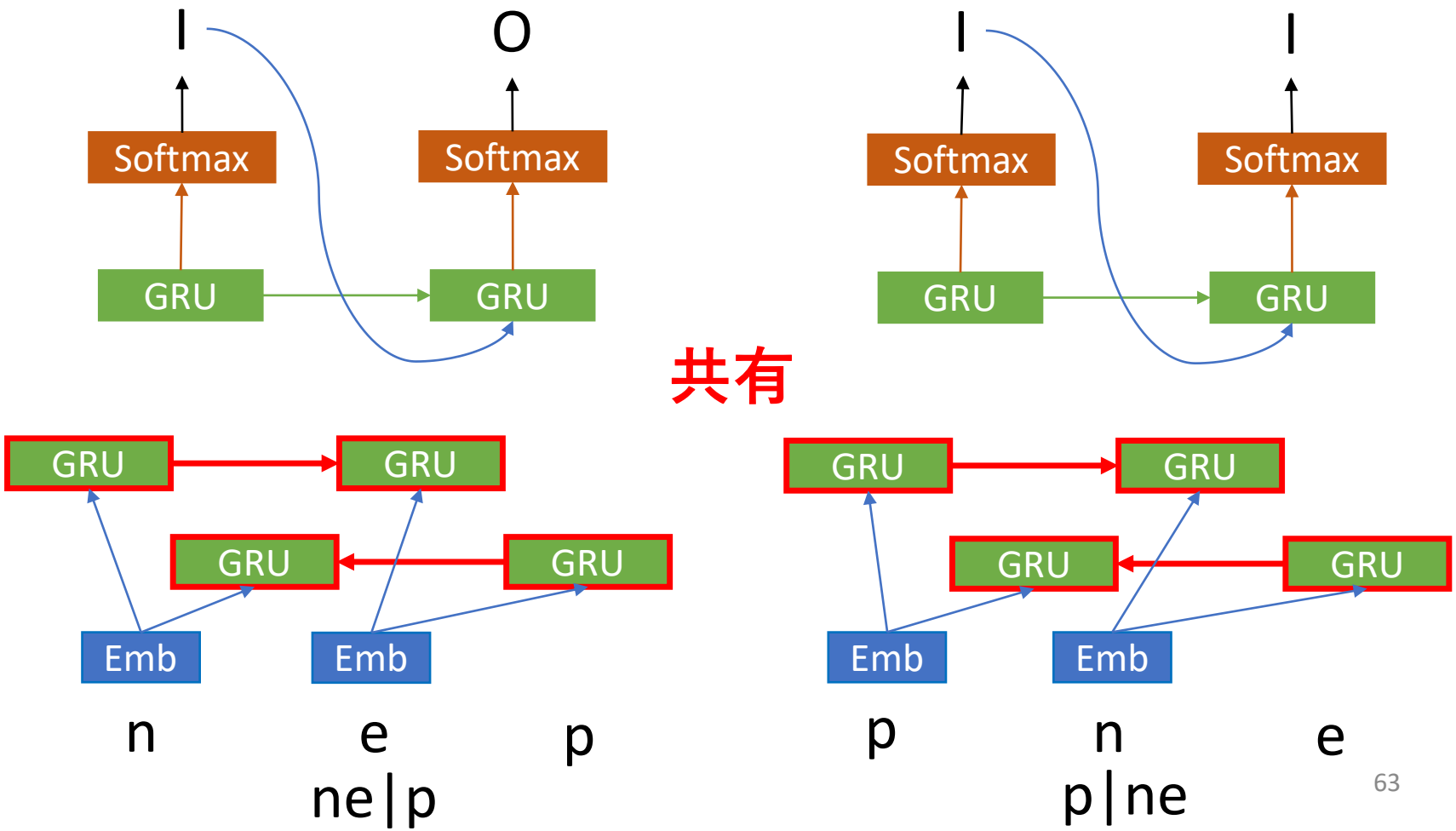
手法2：疑似正解の作成

入力列を出力列にする



手法2：疑似正解の作成

単語内の文字列を入れ替えてデータを増加



実験結果

先ほど紹介したEncoder-Decoder（ベースライン）

	F1							
	MTT-U	MTT-R	DA-U	DA-R	S2S	MORF	CRF	FC
Mex.	**.8786**	.8694	.6715	.8683	.8618	.5121	.8639	.5621
Nahuatl	.7388	.7367	.6865	.7328	.7266	.4154	**.7487**	.5185
Wixarika	.7949	.8024	.7109	**.8161**	.7961	.4426	.7932	.5568
Yorem N.	.7887	**.8076**	.7133	.7923	.7730	.3528	.7736	.6139

入力と出力が同じ＋マルチタスク学習

入力をランダム入れ替え＋マルチタスク学習

入力と出力が同じ

入力をランダム入れ替え

実験結果

先ほど紹介したEncoder-Decoder（ベースライン）

	F1							
	MTT-U	MTT-R	DA-U	DA-R	S2S	MORF	CRF	FC
Mex.	**.8786**	.8694	.6715	.8683	.8618	.5121	.8639	.5621
Nahuatl	.7388	.7367	.6865	.7328	.7266	.4154	**.7487**	.5185
Wixarika	.7949	.8024	.7109	**.8161**	.7961	.4426	.7932	.5568
Yorem N.	.7887	**.8076**	.7133	.7923	.7730	.3528	.7736	.6139

Nahuatlを除いてCRF（素性ベース）に勝利

入力と出力が同じ＋マルチタスク学習

入力をランダム入れ替え＋マルチタスク学習

入力と出力が同じ

入力をランダム入れ替え

実験結果

さすがに効果はない

先ほど紹介したEncoder-Decoder（ベースライン）

	F1							
	MTT-U	MTT-R	DA-U	DA-R	S2S	MORF	CRF	FC
Mex.	**.8786**	.8694	.6715	.8683	.8618	.5121	.8639	.5621
Nahuatl	.7388	.7367	.6865	.7328	.7266	.4154	**.7487**	.5185
Wixarika	.7949	.8024	.7109	**.8161**	.7961	.4426	.7932	.5568
Yorem N.	.7887	**.8076**	.7133	.7923	.7730	.3528	.7736	.6139

入力と出力が同じ＋マルチタスク学習

入力をランダム入れ替え＋マルチタスク学習

入力と出力が同じ

入力をランダム入れ替え

実験結果

以外にも効果的

先ほど紹介したEncoder-Decoder（ベースライン）

	F1							
	MTT-U	MTT-R	DA-U	DA-R	S2S	MORF	CRF	FC
Mex.	**.8786**	.8694	.6715	.8683	.8618	.5121	.8639	.5621
Nahuatl	.7388	.7367	.6865	.7328	.7266	.4154	**.7487**	.5185
Wixarika	.7949	.8024	.7109	**.8161**	.7961	.4426	.7932	.5568
Yorem N.	.7887	**.8076**	.7133	.7923	.7730	.3528	.7736	.6139

入力と出力が同じ＋マルチタスク学習

入力をランダム入れ替え＋マルチタスク学習

入力と出力が同じ

入力をランダム入れ替え

まとめ

- Seq2Seqにおいて少数のデータセットから抱合語（長い複合動詞を多く含む言語）の形態素解析を学習する手法を提案
  - マルチタスク学習
  - データの増加法
- 小規模データでは強力なベースラインである素性に基づくCRFを超える性能を実現

今回紹介した論文

- **Deep contextualized word representations**
  - Best paper
  - 大規模コーパスによる精度向上を複数タスクで実現
- **Recurrent Neural Networks as Weighted Language Recognizers**
  - Outstanding paper
  - RNNの理論限界を明らかに
- **Fortification of Neural Morphological Segmentation Models for Polysynthetic Minimal-Resource Languages**
  - 小規模コーパスで学習するための手法を検証

ご清聴ありがとうございました

参考文献

- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer, "**Deep contextualized word representations**", *In Proc of NAACL 2018*
- Yining Chen, Sorcha Gilroy, Andreas Maletti, Jonathan May, Kevin Knight, "**Recurrent Neural Networks as Weighted Language Recognizers**", *In Proc of NAACL 2018*
- Katharina Kann, Manuel Mager, Ivan Meza-Ruiz, Hinrich Schütze, "**Fortification of Neural Morphological Segmentation Models for Polysynthetic Minimal-Resource Languages**", *In Proc of NAACL 2018*

参考文献

- Alane Suhr, Srinivasan Iyer and Yoav Artz, "***Learning to Map Context-Dependent Sentences to Executable Formal Queries***", *In Proc of NAACL2018*
- Elizabeth Clark, Yangfeng Ji and Noah A. Smith, ***"Neural Text Generation in Stories using Entity Representations as Context"***, *In Proc of NAACL2018*

参考文献

- *https://naacl2018.wordpress.com/*
- *https://allennlp.org/elmo*
- *https://github.com/allenai/bilm-tf*