講演番号 A-14

目的言語の文書文脈を用いたニューラル機械翻訳

17890543　山岸駿秀
指導教員: 小町守 准教授

首都大学東京主催　平成30年度情報通信システム学域　修士論文発表会
平成31年2月6日首都大学東京日野キャンパス2号館 401・402教室

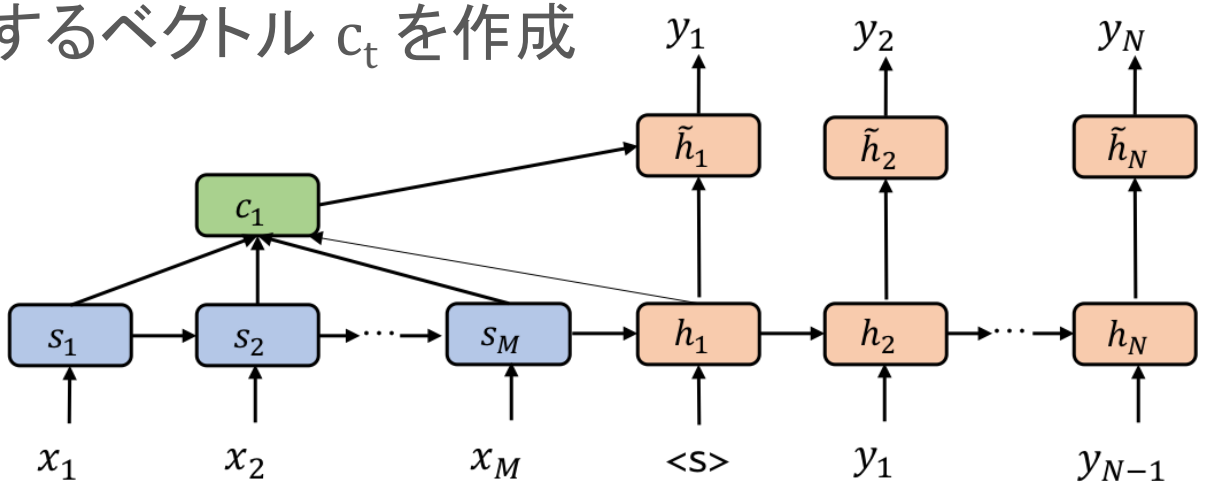
本研究の概要

- 機械翻訳: 言語Fで書かれた文を言語Eで書き換える
- 近年はニューラル機械翻訳（NMT）が研究・開発の主流
  - NMTは遠い単語間の関係を捉えられる
- 文を超えた関係（文書文脈）を導入する研究が増加

  → **文脈つきニューラル機械翻訳（文脈つきNMT）**
- 文脈つきNMTでは、原言語側の文脈が有用とされる
- 本研究: 目的言語側の文書文脈を活用する手法の提案
  - Decoder側の計算履歴を用いる

NMT（dot型 Global Attention）[Luong+, 2015]

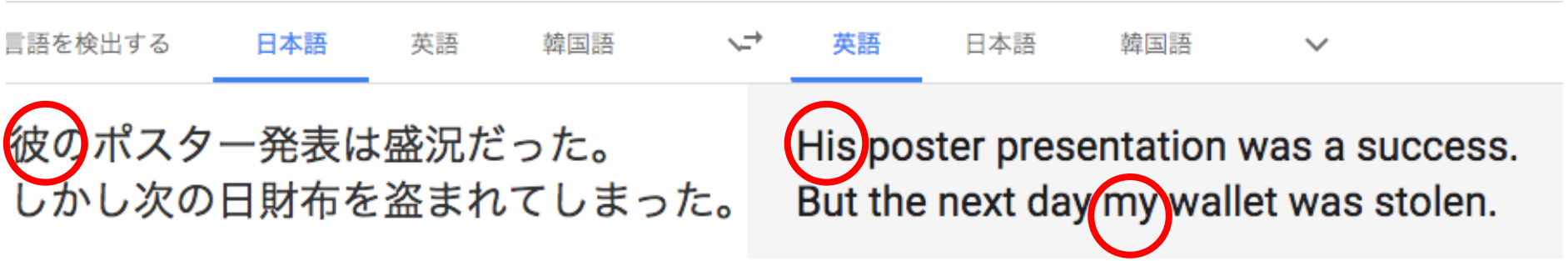
- Encoder（青部）
  - 入力単語 x_t の列を隠れ状態ベクトル s_t の列へ変換
- Decoder（橙部）
  - Attentionの情報 c_t 、時刻 t-1 の隠れ状態ベクトル h_{t-1} 、生成単語 y_{t-1} の情報から時刻 t での単語 y_t を生成
- Attention（緑部）
  - Encoderで得られたベクトル列から適宜必要な情報を参照するベクトル c_t を作成



機械翻訳は文脈情報を扱えない

- 人間の翻訳時は文書を俯瞰で見ることができる
  - 文脈を自由に使うことができる
  - 訳語・文体の一貫性の担保

- Google翻訳（https://translate.google.co.jp, 2019/01/23現在）
  - “財布”の所有者は“彼” → “my wallet”に翻訳
  - 現在の機械翻訳システムでは、文脈を扱えない



文脈つきNMTの先行研究

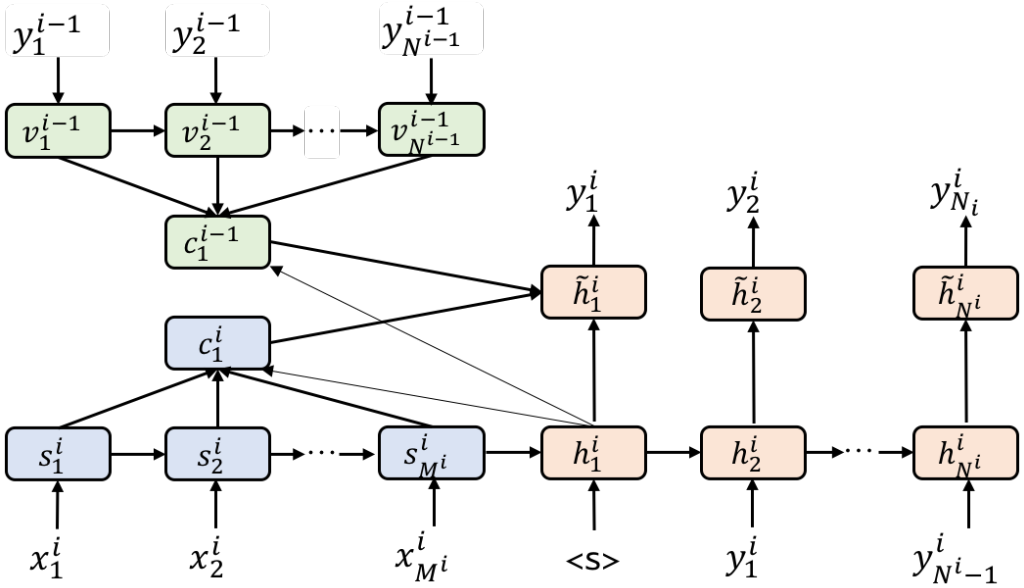
- 階層型Encoder [Wang+, 2017, 中英]
  - 下位のEncoderで単語ベクトルを文ベクトルへ変換
  - 上位のEncoderで文ベクトルを文書ベクトルへ変換
  - 複数文の情報を扱うことが可能

- キャッシュ [Tu+, 2018, 中英]
  - 生成単語とそのときの状態ベクトルをキャッシュで保存
  - キャッシュを適宜参照することで翻訳履歴を考慮
  - 概ね5文以上前の情報は必要性が薄い

→ いずれの手法も、大規模なニューラルネットが必要

Separated Multi-Encoder [Bawden+, 2018]

- 現在の文脈つきNMT研究で最も多用 [Müller+, 2018, 英独等]
- 2つのEncoderを使って2文の情報を取得
  - Encoder: 入力文を読み込む
  - 文脈Encoder: 1つ前の文（文脈文）を読み込む
- それぞれに対してAttentionを計算して生成時に使用



Separated型の知見

- 共参照解析の痕跡 [Tiedemann+, 2017, 独英], [Voita+, 2018, 英露]
  - 共参照解析: 単語Aが指すものと同じものを指す単語Bを特定
  - 英露翻訳: ロシア語の代名詞は指す名詞の性によって変化
  - Multi-Encoder型が文脈情報を扱えている可能性を示唆

- [Bawden+, 2018] での見解
  - 原言語側の文脈文(1つ前の文)は有用
  - 目的言語側の文脈文(1つ前の文)は有用でない

目的言語側の文脈文は不要？

- [Bawden+, 2018] は英仏翻訳
  - 日英翻訳など、言語間の類似性が低い場合も同様？

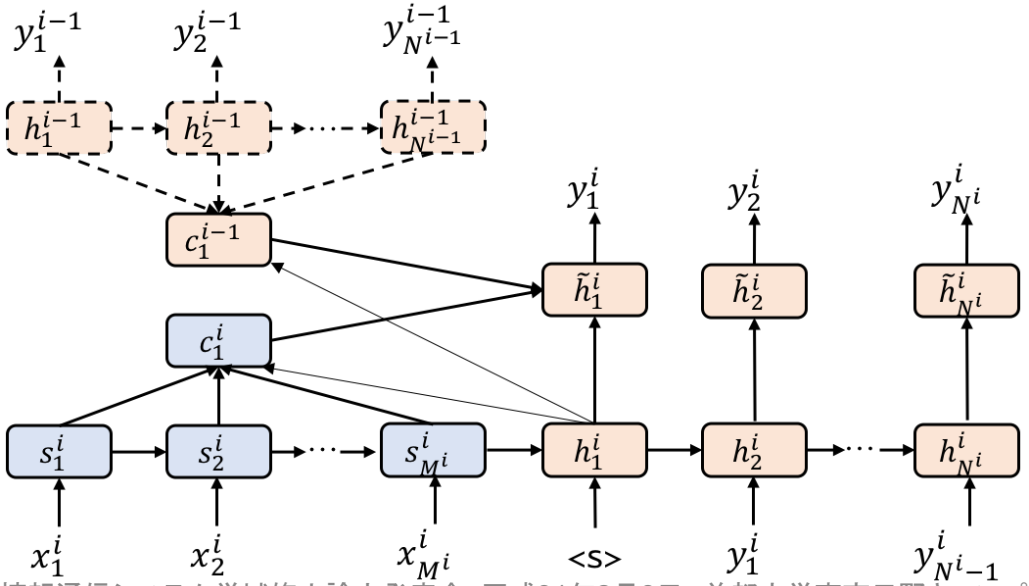
- 構造的な問題があるのでは？
  - 原言語側の文脈を使うとき
    → 文脈文も入力文も **Encoder** の枠組みで扱う
  - 目的言語側の文脈を使うとき
    → 文脈文は**Encoder**、出力文は**Decoder**の枠組みで扱う

- 本研究では、以下の仮説を調査する
  - 「目的言語側の文脈情報はDecoderを用いて得るべき」
    （**文脈文も出力文もDecoder**で扱うべき）

Shared型（提案手法）

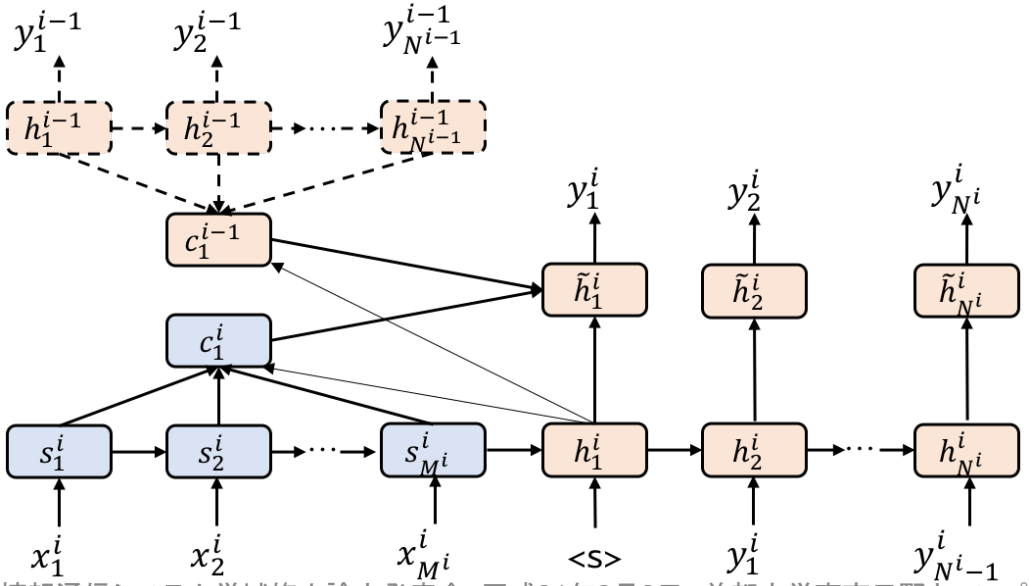
- 文脈文を翻訳した際の隠れ状態ベクトルを保存
- 保存したベクトル列を文脈Encoderの計算結果とする
  - 文脈Encoderの計算が不要
  - Encoderと文脈Encoderの重み行列を共有しているとみなせる

- 文脈Encoder（保存したベクトル列）にAttentionを張る



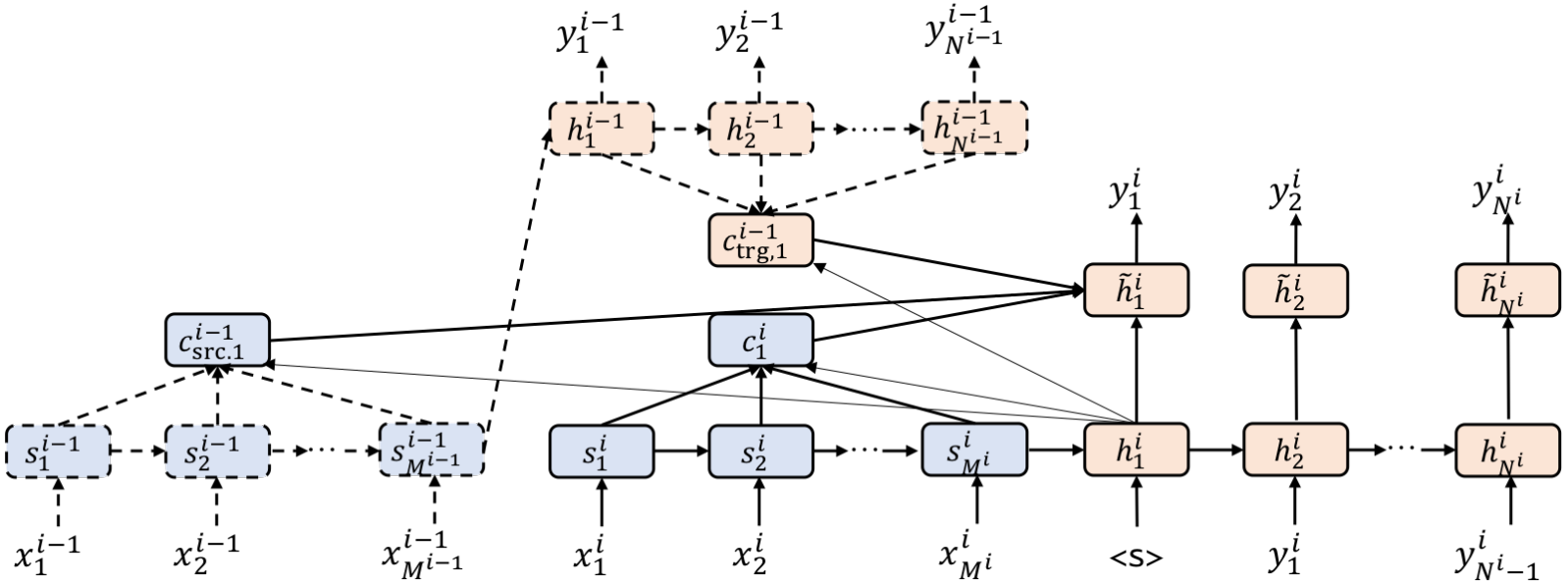
Shared型の文脈文

- Shared Source型（原言語側の文脈を扱う）
  - 入力文も文脈文もEncoderで扱う（Separated型と同様）
  - 重み共有によりパラメータ数を抑えることができる
- Shared Target型（目的言語側の文脈を扱う）
  - Decoderの隠れ状態ベクトルを保存する
  - 目的言語側の文脈情報をDecoderから取得できる



Shared Mix型（提案手法その2）

- 両方の文脈文を使ったときの結果についても調査
- Shared Mix型
  - 文脈文の翻訳時の隠れ状態ベクトルを両側で保存しておく
  - それぞれに対してAttentionを計算し、和をAttentionとして使用
  - パラメータ数は他のShared型と同等



実験

コーパス

- TEDコーパス（Ted Talksにあるプレゼン動画の字幕）
  - 独英・英独: 同じ語族
  - 中英・英中: 違う語族、同じ語順（SVO）
  - 日英・英日: 違う語族、違う語順（SOVとSVO）、省略の有無
- Recipeコーパス（ユーザがCookpadに投稿したレシピ）
  - 日英・英日のみ

コーパス	単語分割器	学習データ	検証データ	評価データ
TED 独英	Moses Tokenizer	203,998	888	1,305
TED 中英	Jieba / Moses Tokenizer	226,196	879	1,297
TED 日英	MeCab / Moses Tokenizer	194,170	871	1,285
Recipe 日英	MeCab / Moses Tokenizer	108,990	3,303	2,804

実験設定（共通）

- Baseline: dot Global Attention型 NMT [Luong+, 2015]
  - ハイパーパラメータは下表のとおり（提案手法も同様）
- 評価尺度: BLEU

ハイパーパラメータ	設定した値
単語分散表現の次元数	512
隠れ状態ベクトルの次元数	512
語彙空間のサイズ	TED: 32,000, Recipe: 8,000（BPE適用）
ミニバッチのサイズ	128文書
Encoder / Decoder	2層 双方向 / 単方向 LSTM
ビーム探索の探索幅	5
最適化	AdaGrad（初期学習率: 0.01）
学習のEpoch	30

実験設定（提案手法）

- Baselineの学習で得られた重み行列を初期値に使用
- 文書の1文目の翻訳時は 文脈Attention = **0**

- Separated型
  - 文脈Encoderはランダム値で初期化
  - Separated Target型の学習時は文脈文として正解文を入力
- Shared型
  - 学習時・検証/評価時のどちらでも保存したベクトル列を使用

- 乱数シードを変えて各3回実験
  - 平均と標準偏差
  - Bootstrap Resamplingによる統計的有意差（Baseline比）

実験結果: 目的言語側の文脈

- Shared Target型がほぼ全ての言語対で**最高性能**
  - 英日翻訳でも最高性能の手法と同程度の結果
- Separated Target型の一部ではBaselineより悪化
- 目的言語側の文脈文はDecoder側から得るべき

実験	Baseline	Separated 型		Shared 型		
		Source	Target	Source	Target	Mix
TED 独英	26.55	26.29 ± .37	26.52 ± .12	*27.20 ± .11	***27.34** ± .11	27.18 ± .21
TED 英独	21.26	21.04 ± .64	20.77 ± .10	21.63 ± .27	**21.83** ± .30	21.50 ± .29
TED 中英	12.54	12.52 ± .33	12.63 ± .24	*13.36 ± .41	***13.52** ± .10	*13.23 ± .09
TED 英中	8.97	8.94 ± .11	8.71 ± .06	9.45 ± .22	***9.58** ± .13	9.42 ± .19
TED 日英	5.84	*6.64 ± .26	*6.37 ± .12	*6.95 ± .07	***6.96** ± .18	*6.81 ± .16
TED 英日	8.40	8.58 ± .12	8.26 ± .00	8.51 ± .31	8.59 ± .08	**8.66** ± .14
Recipe 日英	25.34	*26.51 ± .09	*26.69 ± .15	*26.90 ± .17	***26.92** ± .10	*26.78 ± .11
Recipe 英日	20.81	*21.87 ± .12	*21.45 ± .14	***22.02** ± .20	*21.97 ± .09	*21.81 ± .15

実験結果: 原言語側の文脈

- 原言語側でもSeparated型よりShared型の方が高性能
- 重み共有自体に意味があるのでは？
  - 層ごとの重みを共有しても性能が悪化しない [Dabre+, 2019]
  - マルチタスク学習の枠組みで考えられる可能性もある

実験	Baseline	Separated 型		Shared 型		
		Source	Target	Source	Target	Mix
TED 独英	26.55	26.29 ± .37	26.52 ± .12	*27.20 ± .11	***27.34** ± .11	27.18 ± .21
TED 英独	21.26	21.04 ± .64	20.77 ± .10	21.63 ± .27	**21.83** ± .30	21.50 ± .29
TED 中英	12.54	12.52 ± .33	12.63 ± .24	*13.36 ± .41	***13.52** ± .10	*13.23 ± .09
TED 英中	8.97	8.94 ± .11	8.71 ± .06	9.45 ± .22	***9.58** ± .13	9.42 ± .19
TED 日英	5.84	*6.64 ± .26	*6.37 ± .12	*6.95 ± .07	***6.96** ± .18	*6.81 ± .16
TED 英日	8.40	8.58 ± .12	8.26 ± .00	8.51 ± .31	8.59 ± .08	**8.66** ± .14
Recipe 日英	25.34	*26.51 ± .09	*26.69 ± .15	*26.90 ± .17	***26.92** ± .10	*26.78 ± .11
Recipe 英日	20.81	*21.87 ± .12	*21.45 ± .14	***22.02** ± .20	*21.97 ± .09	*21.81 ± .15

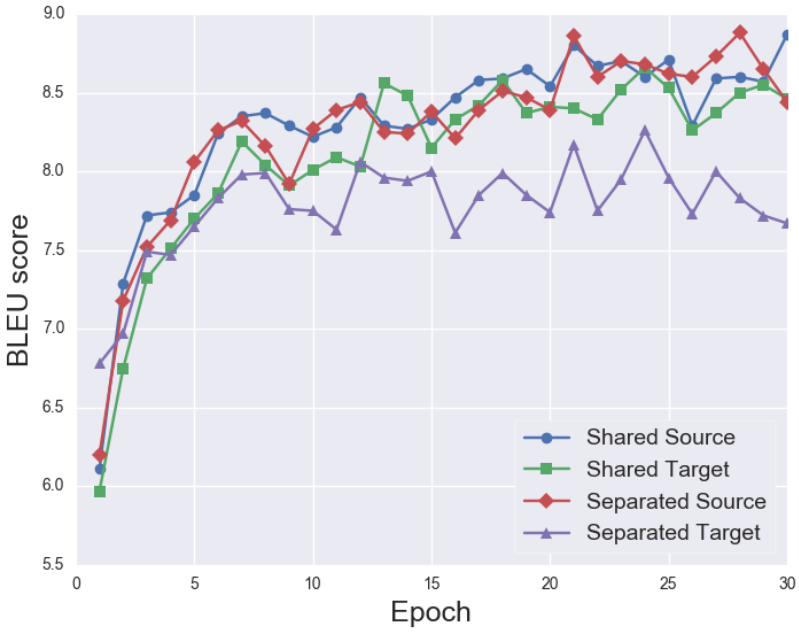
言語対に関する考察

- 独・英・中のどれかを使用 → 目的言語側の方が有用
  - 語族の違いは関係なく、語順（SVO）が同じならば似た傾向？
- 日本語を使用 → どちらも同程度の有用性
  - 日本語の省略の多さが関係している？
  - 語順がSOVであるから？

実験	Baseline	Separated 型		Shared 型		
		Source	Target	Source	Target	Mix
TED 独英	26.55	26.29 ± .37	26.52 ± .12	*27.20 ± .11	***27.34** ± .11	27.18 ± .21
TED 英独	21.26	21.04 ± .64	20.77 ± .10	21.63 ± .27	**21.83** ± .30	21.50 ± .29
TED 中英	12.54	12.52 ± .33	12.63 ± .24	*13.36 ± .41	***13.52** ± .10	*13.23 ± .09
TED 英中	8.97	8.94 ± .11	8.71 ± .06	9.45 ± .22	***9.58** ± .13	9.42 ± .19
TED 日英	5.84	*6.64 ± .26	*6.37 ± .12	*6.95 ± .07	***6.96** ± .18	*6.81 ± .16
TED 英日	8.40	8.58 ± .12	8.26 ± .00	8.51 ± .31	8.59 ± .08	**8.66** ± .14
Recipe 日英	25.34	*26.51 ± .09	*26.69 ± .15	*26.90 ± .17	***26.92** ± .10	*26.78 ± .11
Recipe 英日	20.81	*21.87 ± .12	*21.45 ± .14	***22.02** ± .20	*21.97 ± .09	*21.81 ± .15

学習の安定性

- グラフ: 各EpochごとのBLEU変化（左: 中英　右:英中）
- Separated Target型のみ不安定（英中がより顕著）
  - 文脈文の質と文脈Encoderの質が学習過程でともに変化
  - Shared型はDecoderの質と文脈Encoderの質が同じ
    → 安定性が高い？



実験	文（後処理済み、それぞれ上下が1文目と2文目に対応）
入力文	わかめはよく洗って塩を落とし、10分ほど水に浸けておいてからざく切りにする。長ねぎは小口切りにする。
	熱した鍋にごま油をひき、わかめと長ねぎを入れて30秒ほど軽く炒める。
参照訳	Wash the **wakame** well to remove the salt, put into a bowl of water for 10 minutes and drain. Cut into large pieces. Slice the **Japanese leek**.
	Heat a pan and pour the sesame oil. Stir fry the **wakame** and **leek** for 30 seconds.
Baseline	Wash the **wakame seaweed** well and remove the salt. Soak in water for 10 minutes, then roughly chop. Cut the **Japanese leek** into small pieces.
	Heat sesame oil in a heated pot, add the **wakame** and **leek**, and lightly sauté for about 30 seconds.
Separated Target	Wash the **wakame** well, soak in water for about 10 minutes. Cut into small pieces. Cut the **Japanese leek** into small pieces.
	Heat the sesame oil in a frying pan, add the **wakame** and **leek**, and stir-fry for about 30 seconds.

実験	文（後処理済み、それぞれ上下が1文目と2文目に対応）
入力文	わかめはよく洗って塩を落とし、10分ほど水に浸けておいてからざく切りにする。長ねぎは小口切りにする。
	熱した鍋にごま油をひき、わかめと長ねぎを入れて30秒ほど軽く炒める。
参照訳	Wash the **wakame** well to remove the salt, put into a bowl of water for 10 minutes and drain. Cut into large pieces. Slice the **Japanese leek**.
	Heat a pan and pour the sesame oil. Stir fry the **wakame** and **leek** for 30 seconds.
Baseline	Wash the **wakame seaweed** well and remove the salt. Soak in water for 10 minutes, then roughly chop. Cut the **Japanese leek** into small pieces.
	Heat sesame oil in a heated pot, add the **wakame** and **leek**, and lightly sauté for about 30 seconds.
Shared Target	Wash the **wakame** well, remove the salt, soak in water for about 10 minutes, then roughly chop. Chop the **Japanese leek** into small pieces.
	Heat sesame oil in a heated pan, add the **wakame** and **Japanese leek**, and lightly stir-fry for about 30 seconds.

まとめ

- 既存のNMTに文脈情報を取り入れた
  - 重み共有を用いて、目的言語側の文脈文をDecoder側から取り込む手法を提案
  - 原言語側でも重み共有の性能を調査

- 6言語対の実験結果から、提案手法の有用性を確認
  - どちら側の文脈が必要かは言語対に依存
  - 重み共有自体に意味がある可能性が高い
  - 学習は先行研究より安定している
  - 出力例から訳語の統一ができている可能性を確認
  - 文脈つきNMTをよりコンパクトに実現可能