Optimizing Statistical Machine Translation for Text Simplification

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, Chris Callison-Burch. TACL (4) pp.401-415, 2016.

首都大（小町研 D2）梶原 智之

タスク：Text Simplification

- English Wikipedia: Alfonso Perez
  - Alfonso Perez ~~Munoz, usually referred to as Alfonso,~~ is a former Spanish footballer~~, in the striker position~~.
- Simple English Wikipedia: Alfonso Perez
  - Alfonso Perez is a former Spanish football player.

読みやすくなるように文を書き換えるタスク

- 応用１：自然言語処理のために入力文の複雑さを減らす
- 応用２：言語学習者など人々の文章読解を助ける

問題点：言い換えモデル

- Text Simplification の3つのサブタスク
  1. Splitting: 長い文を連続した短い文に分割する
  2. Deletion: 文のあまり重要でない部分を削除する
  3. Paraphrasing: 並び替え、語彙的/構文的な言い換え

- 新しい言い換えモデルを開発する研究は少ない
  - 統計的機械翻訳ツールをブラックボックスとして使う
    - Coster and Kauchak, 2011; Štajner et al., 2015; など
  - SMTのある一部分のみを変更する
    - 翻訳モデル：Zhu et al., 2010; Woodsend and Lapata, 2011
    - リランキング：Wubben et al., 2012

Optimizing Statistical Machine Translation for Text Simplification

- Text Simplificationを言い換えの問題と考える
  - 語句を平易に書き換える語彙平易化に焦点を当てる
  - 文分割/文圧縮は前/後処理として加えることができる

- SMTのパイプラインに4つの修正を加える
  1. Text Simplificationのための自動評価尺度
  2. 大規模な言い換え規則（翻訳モデルに相当）
  3. Text Simplificationのための素性
  4. マルチリファレンスのデータセット

もくじ

SARI: Text Simplificationのための自動評価尺度

- that compares System output Against References and against the Input sentence.
- システム出力 と リファレンス と 入力 を比較

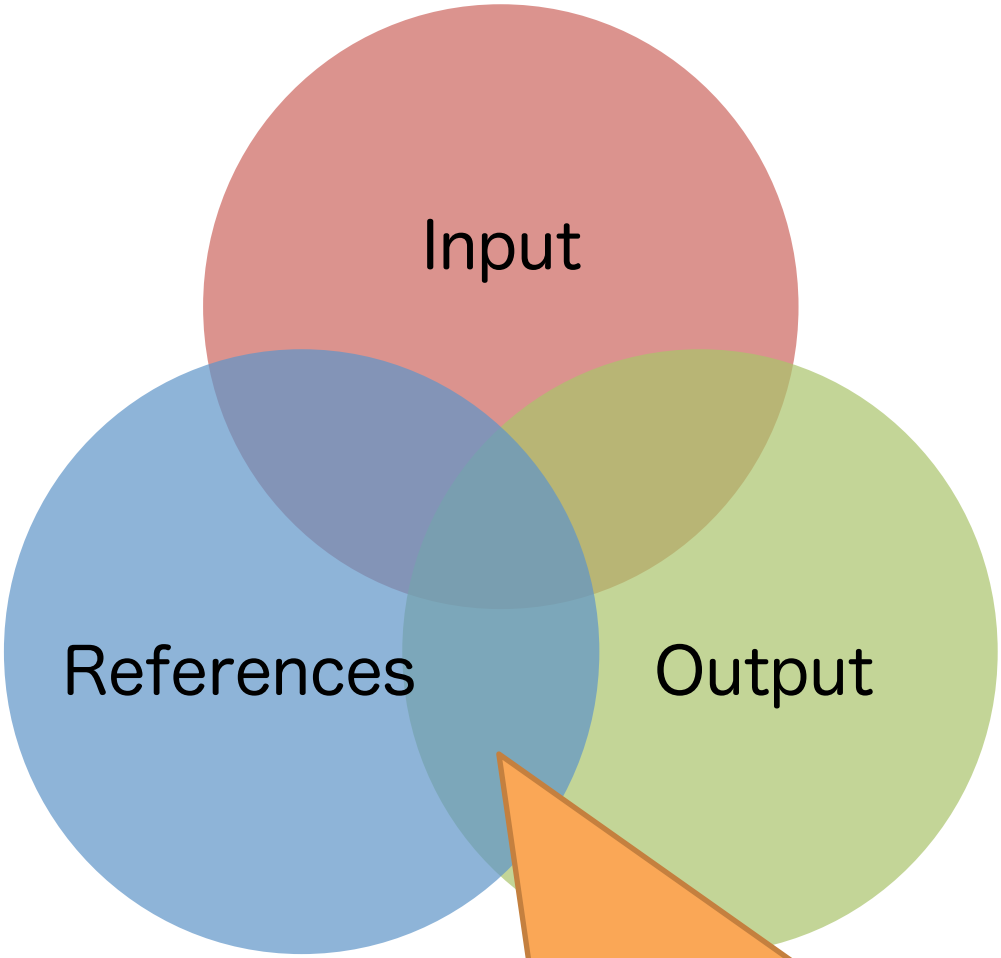
システムによって

- 追加（add）
- 保持（keep）
- 削除（del）

された単語の良さを測定する自動評価尺度

$$SARI = \frac{1}{3}F_{add} + \frac{1}{3}F_{keep} + \frac{1}{3}P_{del}$$

SARI = (F_{add} + F_{keep} + P_{del}) / 3



$$p_{add}(n) = \frac{\bar{I} \cap O \cap R}{\bar{I} \cap O}$$

$$r_{add}(n) = \frac{\bar{I} \cap O \cap R}{\bar{I} \cap R}$$

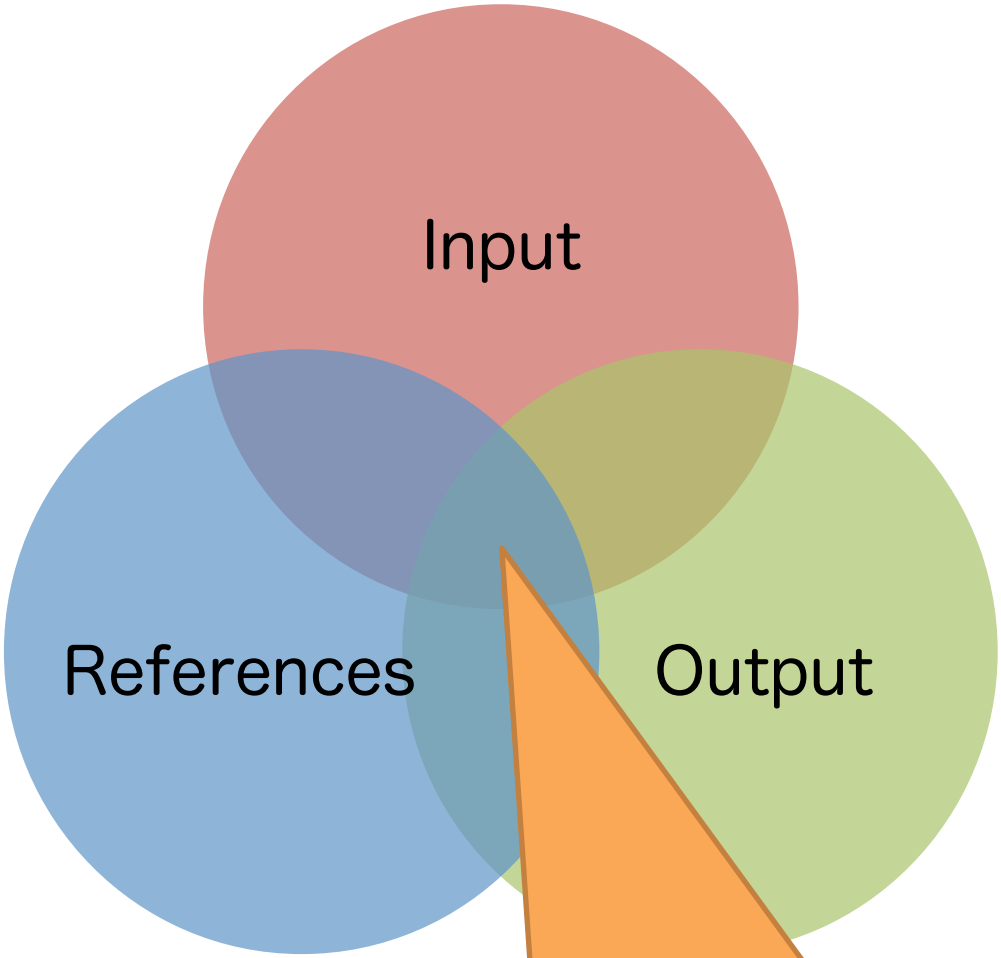
$$P_{add} = \frac{1}{4} \sum_{n=[1,2,3,4]} p_{add}(n)$$

$$R_{add} = \frac{1}{4} \sum_{n=[1,2,3,4]} r_{add}(n)$$

$$F_{add} = \frac{2 \times P_{add} \times R_{add}}{P_{add} + R_{add}}$$

システムが正しく追加した単語

SARI = (F_{add} + F_{keep} + P_{del}) / 3



$$p_{keep}(n) = \frac{I \cap O \cap R}{I \cap O}$$

$$r_{keep}(n) = \frac{I \cap O \cap R}{I \cap R}$$

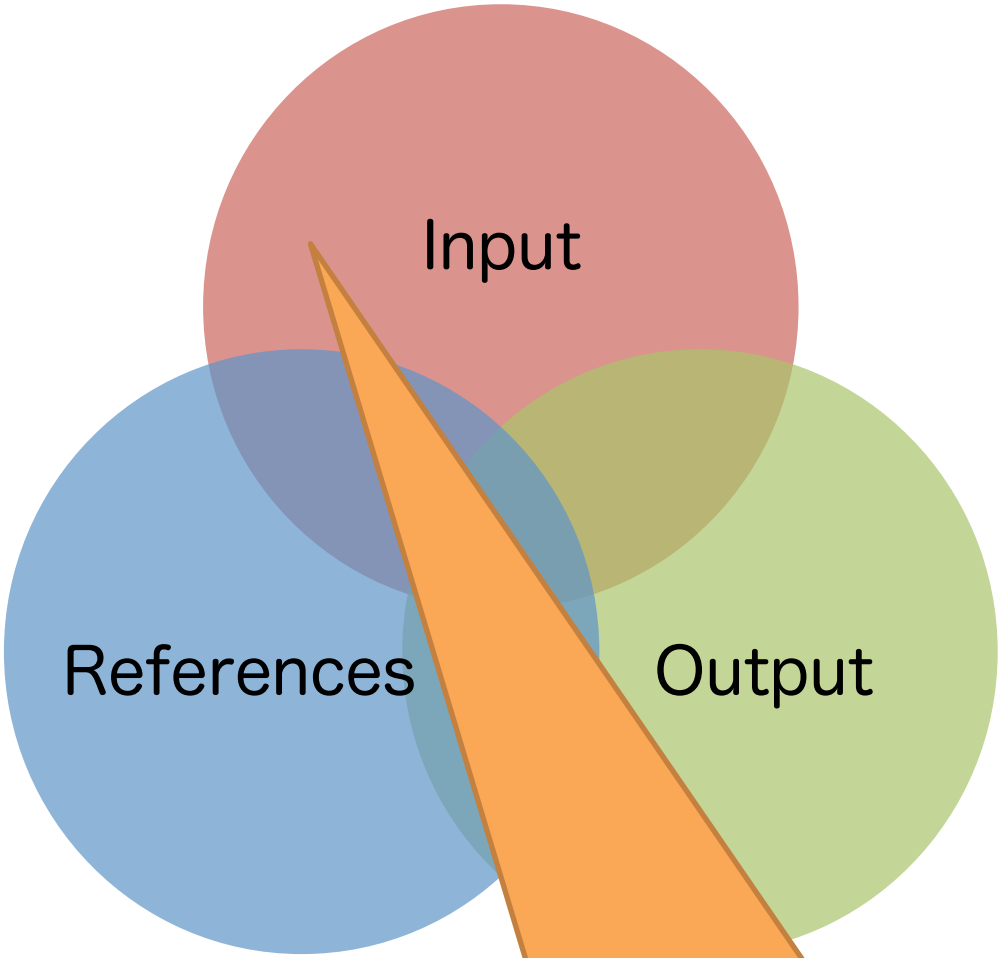
$$P_{keep} = \frac{1}{4} \sum_{n=[1,2,3,4]} p_{keep}(n)$$

$$R_{keep} = \frac{1}{4} \sum_{n=[1,2,3,4]} r_{keep}(n)$$

$$F_{keep} = \frac{2 \times P_{keep} \times R_{keep}}{P_{keep} + R_{keep}}$$

システムが正しく保持した単語

SARI = (F_{add} + F_{keep} + P_{del}) / 3



$$p_{del}(n) = \frac{I \cap \overline{O} \cap \overline{R}}{I \cap \overline{O}}$$

$$P_{del} = \frac{1}{4} \sum_{n=[1,2,3,4]} p_{del}(n)$$

削除しすぎると削除しない
よりもリーダビリティを
損なうので、削除については
適合率のみを考慮する

システムが正しく削除した単語

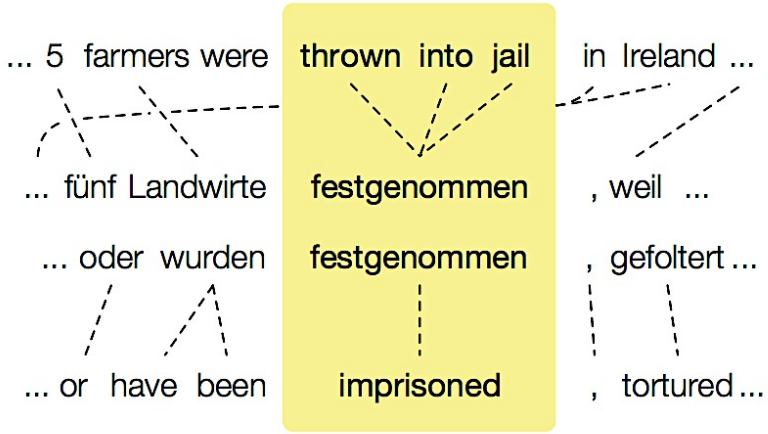
もくじ

1. ~~Text Simplificationのための自動評価尺度~~
2. 大規模な言い換え規則
3. Text Simplificationのための素性
4. マルチリファレンスのデータセット

PPDB: 大規模な言い換え規則

- ~2009年：WordNetや国語辞典などの辞書
- 2010年~：コーパスから書き換え規則を抽出
  （WikipediaとSimple Wikipedia）

- PPDB: 大規模な言い換え規則 (Pavlick et al., 2015)
  - パラレルWikipediaコーパス：10万文対/200万語
  - PPDB：1億文対/20億語（800万語、7,300万句、1.4億構文）



$$p(e_2 \mid e_1) \approx \sum_f p(e_2 \mid f) p(f \mid e_1)$$

もくじ

1. ~~Text Simplificationのための自動評価尺度~~
2. ~~大規模な言い換え規則~~
3. Text Simplificationのための素性
4. マルチリファレンスのデータセット

平易に言い換えるための素性

- 言い換えのための33素性（PPDBと同じ素性）
  - 言い換え確率
  - 分布類似度　など
- 平易化のための5素性
  - 言語モデル（Gigaword corpus + Simple English Wikipedia）
  - 平易な単語の割合
    http://www.manythings.org/vocabulary/lists/l/noll-about.php
  - 文字数
  - 単語数
  - 音節数

もくじ

1. ~~Text Simplificationのための自動評価尺度~~
2. ~~大規模な言い換え規則~~
3. ~~Text Simplificationのための素性~~
4. マルチリファレンスのデータセット

人手で平易化したマルチリファレンス

- Wikipediaから無作為抽出した2,350文 × 8人
  （チューニング：2,000文、テスト：350文）
- クラウドソーシング（Amazon Mechanical Turk）
- 品質管理
  - それぞれの作業者の初めの数文をチェック
  - 初めの数文から全体の作業品質を予測できる
    （Gao et al., 2015）

https://github.com/cocoxu/simplification

実験設定

- デコーダ：Joshua、チューニング：PRO
- 比較手法：Moses + 10-bestのリランキング
- 人手評価
  - Grammar、Meaning、Simplicity
  - 0から4までの5段階評価
  - 5人の評価者による評価の平均値
  - 基準：Normal Wikipedia
- 自動評価
  - FK（リーダビリティ、小さいほど良い）、BLEU、SARI
  - 基準：Mechanical Turk（人手で平易化したデータ）

実験結果

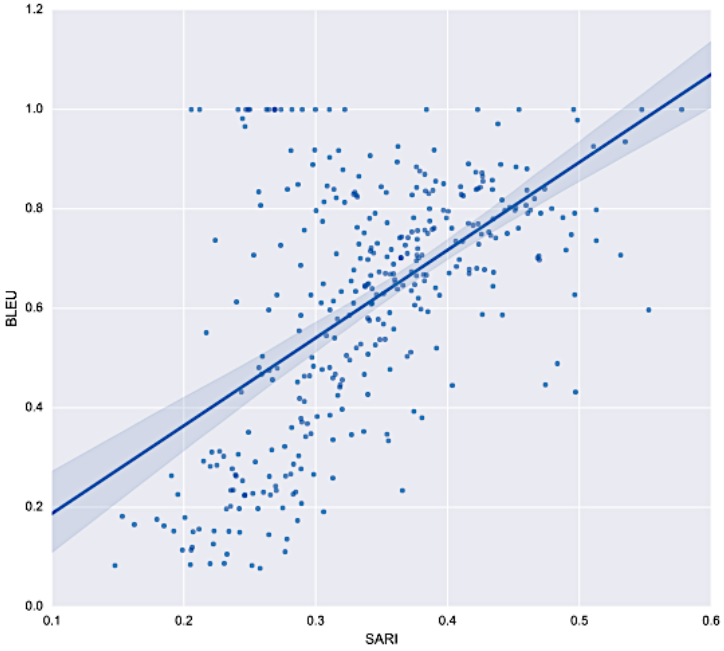
人手評価	Grammar	Meaning	Simplicity	#tokens	#chars	Edit Dist.
Normal Wikipedia	4.00	4.00	0.00	23	125	0.00
Simple Wikipedia	3.72	3.24	1.03	22	116	6.69
Mechanical Turk	3.70	3.36	1.35	19	104	8.25
Wubben et al., 2012	3.18	2.83	0.47	20	108	5.96
PPDB + BLEU	3.30	3.05	0.48	21	107	4.03
PPDB + SARI	**3.50**	**3.16**	**0.65**	23	118	3.98

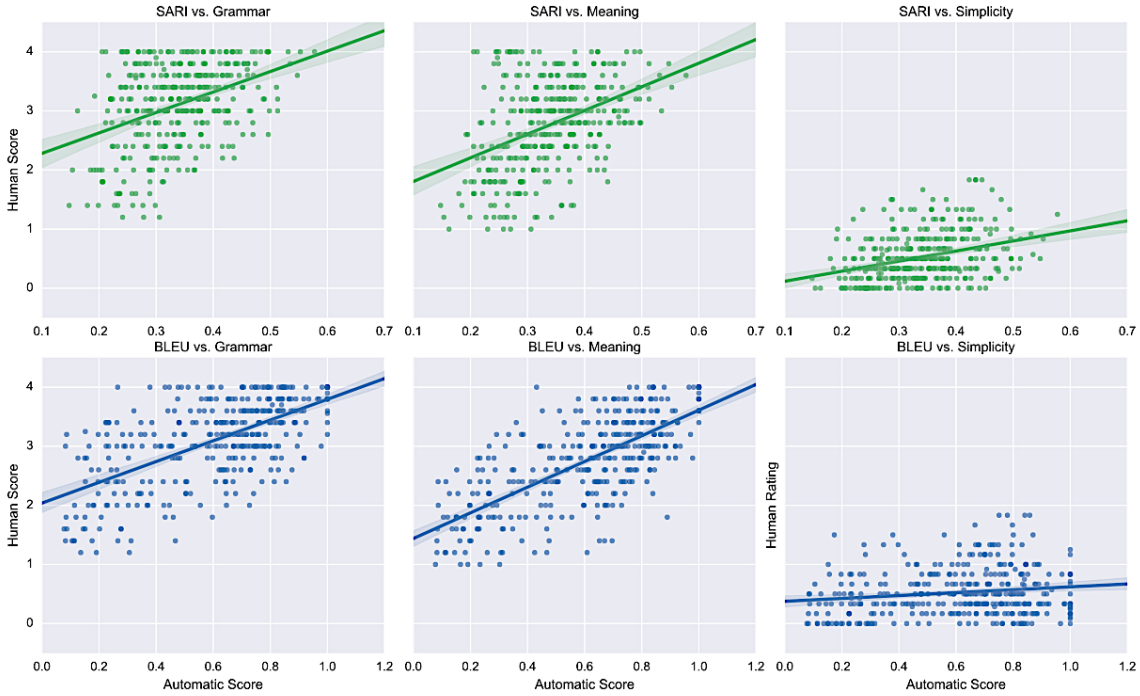
自動評価	FK	BLEU	SARI
Normal Wikipedia	12.88	99.05	26.05
Simple Wikipedia	11.25	66.75	38.42
Mechanical Turk	10.80	100.0	43.71
Wubben et al., 2012	11.10	63.12	33.77
PPDB + BLEU	10.75	74.48	34.18
PPDB + SARI	10.90	72.36	37.91

最適化の計算時間	Time [ms]
BLEU	0.125
SARI	0.155

自動評価尺度と人手評価の相関

Spearman's ρ	ref.	Grammar	Meaning	Simplicity
FK	none	-0.002	0.136	0.147
BLEU	single	0.366	0.459	0.151
BLEU	multiple	0.589	0.701	0.111
SARI	multiple	0.342	0.397	0.343





Optimizing Statistical Machine Translation for Text Simplification

- Simple Wikipedia は英語しか存在しない
- 機械翻訳の自動評価尺度は人手評価との相関が低い

1. Text Simplificationのために
   チューニング尺度 と 素性 を設計したので
   - 人手で構築した平易化コーパスは不要
   - 大規模な言い換え規則（≒平易化規則）で代替可能

   http://paraphrase.org/#/download

2. 人手評価との相関が見られる初めての
   Text Simplificationのための 自動評価尺度 を提案

   https://github.com/cocoxu/simplification

Future Work

- Text-to-Text Generation タスクにおける
  汎用的な自動評価尺度の設計
  - Text Simplification
  - Sentence Compression
  - Error Correction

  いずれも、入力文とシステム出力とマルチリファレンスを比較する自動評価尺度が必要

- Text Simplification のためのNMTモデルの設計
  - BLEUで評価するとPBSMTに負ける