Vol.2014-NL-216 No.11
Vol.2014-SLP-101 No.11
2014/5/22

機械翻訳システムの詳細な誤り分析のための誤り順位付け手法

赤部 晃一[1,a)]　Graham Neubig[1,b)]　Sakriani Sakti[1,c)]　戸田 智基[1,d)]　中村 哲[1,e)]

概要：
本論文では，機械翻訳システムの誤り分析の手法を提案し，先行研究との比較を行う．従来より，誤り傾向は誤り頻度によって示されることが多いが，さらに詳細な分析を誤り単語列の出現頻度に従って行った場合，その単語列は正しい翻訳文の中にも頻繁に出現するため，評価者は正解文も多数評価することとなり，分析の効率は低下する．我々は上述の問題点を解決することを目的として，識別言語モデルを利用した手法を提案した．本論文では，提案手法の更なる拡張，及びモデルに依らない誤りの順位付け手法との比較を行う．

1. はじめに

機械翻訳システムの性能は年々向上している一方，システムの内部は非常に複雑化している．その結果，システムへの改良が翻訳結果に与える影響は必ずしも事前に把握できるわけではなく，実際に翻訳を行ってその結果を評価し，システムを改善することが広く行われている．しかし多くの場合，誤り傾向を捉えるには大量の文を評価する必要があり，非常に時間のかかる作業となる．さらに，目を通す文字列の多くは誤りを含んでいないか，誤りだとしてもシステム全体に影響を及ぼすとは限らない．しかし，重要な誤りに自動的にランク付けを行うことができれば，重要な誤りをより効率的に見つけ出すことが可能になると考えられる．

先行研究 [11], [15] では，機械翻訳システムの誤りを自動的に分類し，頻度順に提示する手法が提案されている．この手法により特定される誤りの種類は，機械翻訳の一般的な誤り（例として活用誤り，並べ替え，単語削除，挿入誤りなど）を網羅しており，この誤りの多さを把握することにより，全体的な誤り傾向を分析の前に把握することができる．一方で，誤りの種類ごとの傾向を掴むだけでなく，個別の誤りに着目した分析を行うことも，システムを改善する上で重要な作業となる．しかし，先行研究のように誤りを単純に頻度で選択した場合，選択された誤りは頻出する言語現象に支配されてしまい，翻訳誤りそのものを捉える目的では有効な手法とは言えない．この一例として，表 1 に機械翻訳システムの出力に含まれて，かつ参照文には含まれない n-gram のリストを示す．表を見て分かる通り，頻繁に出現する誤り n-gram は，単純に英語に頻出する n-gram に過ぎず，この表だけからはシステムの特徴的な誤りが分からない．

表 1　頻繁に出現する誤り n-gram

1-gram		2-gram	
the	61	(BOS) the	42
,	47	. (EOS)	41
and	43	, and	32
of	42	of the	27
:	42	in the	21

この問題の解決策として，我々は n-best による正則化された識別言語モデルに基づく誤り分析手法を提案した [16]．識別言語モデルは，システムの一般的な誤りを修正するように学習される言語モデルである．誤り分析の観点から見れば，識別言語モデルを n-gram 素性によって学習し，学習された重みを調べることにより，負または正に大きい重みが付くことは，すなわち過剰に生成されている，もしくは十分に生成されていない n-gram を表現していると言える．これは重みが誤りを修正するように学習されることに由来し，それらの修正パターンは単純に頻度で選択された誤りに比べて有用であると言える．さらに，識別言語モデルの様々な付帯機能を利用することで，重点的な分析が可能となる．先行研究では L1 正則化を利用しているが，こ

[1] 奈良先端科学技術大学院大学
Nara Institute of Science and Technology
[a)] akabe.koichi.zx8@is.naist.jp
[b)] neubig@is.naist.jp
[c)] ssakti@is.naist.jp
[d)] tomoki@is.naist.jp
[e)] s-nakamura@is.naist.jp

Vol.2014-NL-216 No.11
Vol.2014-SLP-101 No.11
2014/5/22

表 2　評価者によって記録される誤りの種類

誤りの種類
文脈依存置換
文脈非依存置換
挿入
削除
並べ替え
活用

れにより多くの素性が削除され，重要な修正パターンだけがモデルに残る．

本研究では，n-best に依らず単純に出現頻度に基づく誤り選択手法を新たに複数提示し，n-best による識別言語モデルに基づく手法の有効性を検証する．実験では，提示された複数の手法を n-best による識別言語モデルに基づく手法と比較し，誤り箇所選択率と選択される誤りの傾向の分析を行う．また，正則化された識別言語モデルでは，システム出力の 1-best と参照文を利用した学習も行い，n-best で学習した場合との比較を行う．さらに，オラクル文の選択に利用する評価尺度の変更により，特定の分野の誤りに絞った分析が可能となることを示す．

2. 機械翻訳のための詳細な誤り分析

本節では，機械翻訳システムの誤り分析を詳細に行う手法について説明する．

機械翻訳の大雑把な誤り傾向だけではなく，個々の誤りに目を向けた分析をするには，機械翻訳が実際に出力した翻訳文について，どのような誤りが含まれているか個別に判断していけば良い．ある特定の誤り現象が多くの翻訳文の中に含まれている時，その誤り現象はシステムの特徴的な誤りと言える．本論文では，n-gram を用いて上述の特徴的な誤り現象を捉える．

ある n-gram がシステム出力の誤りを捉えている時，その n-gram を誤り n-gram とする．ある基準に基づいて特徴的な誤り n-gram が選択された際，以下に示す枠組によって，選択された n-gram が示すシステムの誤りを調査する．

(1) 各手法によって代表的な誤り n-gram を優先度の高いものから順に選択する．
(2) 選択された n-gram を含む機械翻訳の 1-best 出力を 1 つ以上選択する．
(3) 選択された翻訳文を選択要因となった n-gram とともに評価者に提示する．
(4) 評価者は提示された n-gram が翻訳文中で誤り箇所を捉えているかどうか評価する．誤り箇所を捉えている場合は，表 2 に示す誤りの種類などの情報を記録する．

実際に評価者に提示する分析シートの例を図 1 に示す．

3. 出現頻度を用いた誤り n-gram 選択法

本節では，分析の対象とする誤り n-gram を，システム出力の 1-best E_k と正解文 C_k を利用して決定する手法を複数示す．各手法により n-gram の誤りスコアを計算し，優先的に分析する n-gram を選択する．

Ja	京 ちゃん （ 市 バス ）
En(Ref)	kyo-chan -lrb- city bus -rrb-
En(MT)	kyoto chan | -lrb- kyoto city | bus -rrb-
Rules	SYMP (x0:SYM SYMP (NP (NN (”市”) NN (”バス”)) x1:SYM)) → x0 ”kyoto” ”city” ”bus” x1
Eval	挿入誤り

図 1　評価シートの例．上 4 行が自動的に与えられる情報．5 行目が評価結果．四角で囲まれた単語列は選択された n-gram.

表 3　コーパス中の n-gram の分布

	システム出力	正解文	周辺確率
候補 (x)	$p(x, c = \mathrm{e})$	$p(x, c = \mathrm{c})$	$p(x)$
候補以外	$p(\bar{x}, c = \mathrm{e})$	$p(\bar{x}, c = \mathrm{c})$	$p(\bar{x})$
周辺確率	$p(c = \mathrm{e})$	$p(c = \mathrm{c})$	1.0

本研究では，正解文として機械翻訳の参照文 R_k とオラクル出力 E_k^* のいずれかを利用する．オラクル出力は，n-best 中の各候補に対し，参照文 R_k によって与えられる評価値 EV(例えば BLEU+1[5]) を計算した際，EV が最大となる文とする．訳出の精度を測る場合，参照文が比較対象として用いられることが多い．しかし，参照文には機械翻訳機が出力できないような翻訳候補が含まれていることがあり，システムの修正には新たな翻訳ルールの追加が必要になる場合が多くなると考えられる．一方オラクル出力を利用した場合，選択された n-gram の生成を抑制するだけでシステムの性能が高まると考えられる．

以下の節に，システム出力の 1-best と正解文による誤り候補選択手法を示す．

3.1 誤り頻度

誤り頻度では，ある n-gram がシステム出力に含まれながら正解文には含まれない回数を記録し，最も数が多いものを優先的に分析する．

機械翻訳の入力文 $F_k \in \mathcal{F}$ に対し，システム出力 E_k と正解文 C_k を得る．各文対に対し，C_k に含まれない n-gram x が E_k に含まれている個数 $\#_k(x)$ を $e_k(x)$ とする．

$$e_k(x) = \begin{cases} \#_k(x) & x \notin C_k \\ 0 & (otherwise) \end{cases}$$

各文対に対する e_k の総和 $\phi_e(x) = \sum_k e_k(x)$ を計算し，ϕ_e が大きい n-gram を分析の対象とする．

3.2 自己相互情報量

自己相互情報量では，ある n-gram のシステム出力との関連度を調べ，関連度の高いものを優先的に分析する．

n-gram x が選ばれた際，その n-gram が持つシステム

出力についての自己相互情報量（PMI）をスコアとする．PMI は下式のように定義する [1]．

$$PMI(x) = \log \frac{p(x, c = \mathrm{e})}{p(x) \cdot p(c = \mathrm{e})}$$

ここで関数 p は表 3 に従う．$p(x)$ は n-gram x がコーパスを占める割合，$p(c = \mathrm{e})$ は正解文中の n-gram 全体がコーパスを占める割合，$p(x, c = \mathrm{e})$ は正解文に含まれる x がコーパスを占める割合を表す．更に，コーパス全体における確率と PMI を組み合わせて

$$S(x) = p(x, c = \mathrm{e}) \log \frac{p(x, c = \mathrm{e})}{p(x) \cdot p(c = \mathrm{e})} \quad (1)$$

となる．より多く含まれる n-gram に着目するため，評価関数を式 (1) の通り定義する．$S(x)$ が大きいものを分析の対象とする．

3.3 平滑化された条件付き確率

平滑化された条件付き確率では，ある n-gram に着目し，その n-gram が含まれる文がシステム出力である割合をスコアとする．このスコアが高いものを優先的に分析する．

機械翻訳の入力文 $F_k \in \mathcal{F}$ に対するシステム出力 E_k と正解文 C_k について，含まれる n-gram の個数を比較し，E_k の余剰分を $e'_k(x)$，C_k の余剰分を $c'_k(x)$ とする．各方向について総和 $\phi'_e(x) = \sum_k e'_k(x)$，$\phi'_c = \sum_k c'_k(x)$ を計算する．

ある n-gram を選択した際，その n-gram がシステム出力に含まれる条件付き確率は

$$p(c = \mathrm{e} | w = x) = \frac{\phi'_e(x)}{\phi'_e(x) + \phi'_c(x)}$$

しかし，単純に確率を最尤推定で計算すると，コーパス全体に 1 回しか出現しないような稀な n-gram の確率が 1 となり，頻繁に選択されてしまう．しかし平滑化を行うことで，上述の問題点を解決する．MacKay らは文献 [6] で平滑化の手法としてディリクレ分布を利用しており，本手法もこれに習う．

n-gram x についての評価関数は式 (2) の通りであり，$S(x)$ が大きいものを代表的な n-gram とする．

$$S(x) = \frac{\phi'_e(x) + \alpha P_e}{\phi'_e(x) + \phi'_c(x) + \alpha} \quad (2)$$

ただし，

$$P_e = \frac{\sum_x \phi'_e(x)}{\sum_x \phi'_e(x) + \sum_x \phi'_c(x)}$$

この手法を用いた際，平滑化係数 α を決定する必要がある．n-gram を利用して正解文もしくはシステム出力文を選択する際，選択される文の種類がディリクレ過程 [13] に従うと仮定すると，コーパス全体に対する尤度は式 (3) で表される．

$$P = \prod_x \frac{\prod_{c \in \{\mathrm{e},\mathrm{c}\}} \prod_{k=0}^{\#(w=x,c)} (k + \alpha P(c))}{\prod_{k=0}^{\#(w=x)} (k + \alpha)} \quad (3)$$

式 (3) の P が最大化されるような α をパラメーターとする。P は全区間で微分可能であり，唯一の極があるとき，その点で最大値となる．よって α は P の微分からニュートン法により計算できる．

4. 識別モデルを用いた誤り n-gram 選択法

本節では，先行研究 [16] で提案されている識別言語モデルに基づく誤り n-gram 選択法を説明する．

分析対象として，入力文の集合 $\mathcal{F} = \{F_1, \ldots, F_K\}$ と機械翻訳の n-best 出力 $\hat{\mathcal{E}} = \{\hat{\boldsymbol{E}}_1, \ldots, \hat{\boldsymbol{E}}_K\}$，参照文 $\mathcal{R} = \{R_1, \ldots, R_K\}$ が与えられたとする．このとき，識別言語モデルでは $\hat{\boldsymbol{E}}_k = \{\hat{E}_{k,1}, \hat{E}_{k,2}, \ldots, \hat{E}_{k,I}\}$ の各候補に対して素性関数 $\boldsymbol{\phi}(\hat{E}_{k,i})$ を定義し，重みベクトルと素性関数の内積 $\boldsymbol{w} \cdot \boldsymbol{\phi}(\hat{E}_{k,i})$ をスコアとして定義する．

重みベクトル $\boldsymbol{w}$ は，n-best 中のオラクル E^*_k のスコアが高くなるように学習し，$\boldsymbol{w}$ の中で負に大きい重みが付いた素性を優先的に分析する．

4.1 構造化パーセプトロン

$\boldsymbol{w}$ の学習には構造化パーセプトロン [2] を利用する．構造化パーセプトロンでは，オラクル E^*_k と現在の仮説 $\hat{E}_k$ から計算された素性関数の差分を用いてモデルの更新を行う．各反復において，$\boldsymbol{\phi}(\hat{E}_k)$ と $\boldsymbol{\phi}(E^*_k)$ の差分で $\boldsymbol{w}$ を更新し，E^*_k と $\hat{E}_k$ が等しいときは差分が $\mathbf{0}$ となり更新を行わない．この操作を $\mathcal{F}$ 全体に対して逐次的に行い，さらに重みが収束するか決められた回数 N に達するまで反復する．

4.2 疎な識別言語モデルの学習

構造化パーセプトロンの学習時に L1 正則化 [14] を行うことで，学習の効果が小さい素性をモデルから削除することができる [12]．L1 正則化では，学習時に $\boldsymbol{w}$ に対して L1 ノルム $||\boldsymbol{w}||_1 = \sum_i |w_i|$ に比例するペナルティーを付与し，多くの素性を 0 にする．

L1 正則化は前向き後ろ向き分割 (FOBOS) アルゴリズム [3] により行う．FOBOS を利用することで，学習の処理と正則化の処理を分け，学習の高速化を行える．

4.3 識別言語モデルの素性

識別言語モデルの学習では，以下の素性を用いる:

(1) システムのスコア ϕ_s: この素性により，システムが出力した n-best の順序を修正するように学習する．

(2) n-gram の頻度 ϕ_n: n-gram が仮説中に含まれる個数を素性とする．これにより n-gram に焦点を当てた分析をする．

(3) 仮説の単語数 ϕ_l: 評価尺度が単語数に対する罰則を行

表 3　KFTT のデータサイズ

	文数	単語数	
		英語	日本語
Train	330k	5.91M	6.09M
Dev	1166	24.3k	26.8k
Test	1160	26.7k	28.5k

う場合，この素性によって単語数を調整する．

4.4　参照文を使った識別言語モデル

n-best とは別に，システム出力の 1-best を負例，参照文を正例とした識別言語モデルも学習できる．これを行うには，システム出力の 1-best $\hat{E}_k$ と参照文 R_k の組 $\{\hat{E}_k, R_k\}$ を n-best の代わりとして用いる．システムスコア素性 ϕ_s は正例と負例の識別に利用し，システム出力の 1-best に対して 0，参照文に対して 1 を与える．ただし，システムスコアに対する重みの更新は行わない．この際，小さいコーパスに対して適切な学習を行えるように，自動評価尺度のスコアに比例するマージンを設定する．マージンを設定することにより，自動評価が良い文の素性に対しては負の重みを小さく与え，自動評価が悪い文の素性に対しては負の重みを大きく与えることが可能となる．

5.　実験

各手法の有効性を検証するために，機械翻訳の訳出を利用した評価実験を行った．

各手法について，2 章で示した枠組により各 n-gram x の誤り箇所選択率を調査する．n-gram の誤り箇所選択率は，式 (4) によって定義される．

$$W(x) = \frac{\#(x, state = \text{error})}{\#(x)} \tag{4}$$

ここで $\#(x, state = \text{error})$ は n-gram が誤り箇所を選択する個数，$\#(x)$ は n-gram が評価シート中に出現する個数である．分析の候補として選択された n-gram が誤り箇所を選択する割合が高いとき，その手法は誤り分析において有用な手法である．

5.1　実験設定

すべての実験で京都フリー翻訳タスク (KFTT)[7] の日英翻訳を利用した．コーパスの大きさを表 3 に示す．評価対象とした翻訳システムは Travatar ツールキット [8] に基づく Forest-to-String システムである．

翻訳システムを構築する上で，単語間アラインメントに Nile[*1] を利用し，構文木の生成には Egret[*2] を利用した．チューニングには MERT [9] を利用し，評価尺度を BLEU[10] とした．

評価に用いた手法は，3 章と 4 章で説明した以下のシステムである．

(1) ランダム
(2) 誤り頻度
(3) 自己相互情報量
(4) 平滑化された条件付き確率
(5) 識別言語モデルの重み

なお，誤り頻度，自己相互情報量，平滑化された条件付き確率では，利用する正解文がオラクル出力と参照文の 2 通りの場合で実験を行った．

n-best による識別言語モデルの学習は，反復回数を 100 回とし，参照文を利用する識別言語モデルの学習では反復回数を 10000 回とした．これは，n-best による手法は計算に時間がかかる一方，参照文による手法は計算が高速であるためである。両手法で 4.2 節で説明した FOBOS により L1 正則化を行った．正則化係数は 10^{-7}-10^{-2} の中から選び，KFTT のテストデータに対して高い精度を示す値を利用した．参照文を使った識別言語モデルの学習では，マージンを BLEU+1 のスコア差の 100 倍に設定した．各手法とも，n-gram として 1-gram から 3-gram までを利用した．

n-best による識別言語モデルの学習では，オラクル文の選択に用いる評価尺度を BLEU+1 と RIBES[4] の 2 通りで行い，選択される n-gram の誤り傾向を分析した．

誤り頻度，平滑化された条件付き確率，自己相互情報量の計算では，頻繁に共起する n-gram に同一の高いスコアが与えられることがある．この n-gram の組はそれぞれ 1 つの誤りと見なせるが，何も処理せずに評価した場合，評価全体が 1 つの誤りが持つ複数の n-gram に支配される問題が起こる．本実験ではこの問題を回避するため，共起する n-gram をグループ化し，1 つの n-gram として扱う．グループ化の例を図 2 に示す．コーパス中で “bakufu -lrb- japanese” と “japanese feudal government” はすべての箇所で共起する．これを別々に見た場合，1 つの誤りを重複して見ることになるため，2 つの n-gram を同一視し，“bakufu -lrb- japanese feudal government” を 1 つの n-gram と見なしている．

Ja	室町 幕府 の 初代 征夷 大将軍 。
En(Ref)	he was the 1st seii taishogun of the muromachi shogunate .
En(MT)	he was the first seii taishogun of the muromachi bakufu -lrb- japanese feudal government headed by a shogun -rrb- .
Eval	挿入誤り

図 2　n-gram グループ化の例

5.2　各手法の誤り傾向

各手法により n-gram を選択した際の誤り箇所選択率を，2 章に示した手順で調査した．選択する誤りは各手法とも上位 30 個とし，上位 30 個の n-gram の誤り箇所選択率の

[*1] 入手先 ⟨http://code.google.com/p/nile/⟩
[*2] 入手先 ⟨http://code.google.com/p/egret-parser/⟩

表 4　各手法により選択された n-gram の内訳．誤り箇所選択率がランダムよりも高い 3 つの手法を太字で示した．また，3 つの手法の中で各種類の誤りについて多く検出されたものを太字とした．

種類	ランダム	誤り頻度		自己相互情報量		条件付き確率		識別言語モデル	
比較対象		ORACLE	REF	ORACLE	REF	ORACLE	REF	N-BEST	REF
誤り箇所選択率	0.483	0.290	0.323	0.427	0.410	**0.607**	**0.713**	**0.598**	0.387
文脈依存置換	0.124	0.460	0.258	0.500	0.520	0.218	**0.449**	0.332	0.259
文脈非依存置換	0.166	0.023	0.052	0.086	0.049	0.016	**0.140**	0.067	0.078
挿入	0.111	0.195	0.278	0.078	0.260	**0.537**	0.364	0.164	0.276
削除	0.062	0.103	0.093	0.016	0.041	0.042	0.023	**0.245**	0.052
並べ替え	0.319	0.218	0.299	0.305	0.106	0.176	0.023	**0.192**	0.155
活用	0.093	0	0.021	0.016	0.024	0.011	0	0	0.181

加算平均を計算した．その結果を表 4 に示す．

表から，誤り箇所選択率が高い手法は，平滑化された条件付き確率に基づく手法と，n-best による識別言語モデルに基づく手法であることが分かる．その他の手法はランダム選択を下回っており，誤り箇所選択率が高いとは言えない．上位 3 つの手法を比較すると，参照文を利用した条件付き確率に基づく手法では，置換誤りや挿入誤りを多く捉えているが，削除誤りをほとんど捉えていないことが分かる．一方 n-best による識別言語モデルに基づく手法では，他の手法に比べて 2 倍以上の削除誤りを捉えていることが分かる．

n-best による識別言語モデルに基づく手法以外で削除誤りの検出率が悪い原因として，削除誤りを検出する際には削除された単語列ではなく，その前後の文脈を見る必要があることが挙げられる．削除誤りが発生する前後の文脈は原文によって大きく変わるため，n-gram の発生頻度が小さく候補から外れやすくなる．しかし n-best を使った識別言語モデルに基づく手法の場合，同じ文脈における削除誤りが n-best 中の複数の候補に発生するため，削除誤りが修正されるまで n-gram の重みを大きくしようとする．結果として，n-best を使った識別言語モデルに基づく手法では，削除誤りも多く捉えることができる．

識別言語モデルに基づく手法では，学習データによって誤り箇所選択率が大きく変化することが分かる．識別言語モデルを利用する場合，単純に 1-best と参照文を利用するよりも n-best を利用した方が，誤り箇所選択率の向上に有効であることが分かる．これは，単純に 1-best と参照文で学習した場合は学習データが少なく過学習を起こす可能性が高いが，n-best を使う場合，重要でない誤りが正則化によって削除され，多くの候補で誤りとして含まれる n-gram に大きい重みが与えられるためと考えられる．

5.3　選択された n-gram のシステム出力内出現回数

各手法によって選択された n-gram がシステム出力に出現する回数を表 5 に示す．表を見ると，誤り頻度に基づく手法，相互情報量に基づく手法，1-best と参照文による識別言語モデルに基づく手法では，選択された n-gram が多くの翻訳に出現することが分かる．しかし表 4 の誤り箇所選択率は低いため，誤り以外の n-gram も多く選択していることが分かる．一方条件付き確率に基づく手法と n-best による識別言語モデルに基づく手法では，他の手法に比べて選択された n-gram の出現回数が少ないが，誤り箇所選択率が大きく，誤り以外の部分を捉えにくい手法であることが分かる．

表 5　選択された n-gram がシステム出力に出現する回数の平均

手法		dev	test
ランダム		1.1	0.2
誤り頻度	ORACLE	381.0	402.8
	REF	368.0	378.6
自己相互情報量	ORACLE	135.4	151.9
	REF	225.1	225.1
条件付き確率	ORACLE	21.4	11.2
	REF	19.7	7.7
識別言語モデル	N-BEST	6.2	4.0
	REF	175.9	183.1

n-best による識別言語モデルに基づく手法は，選択された n-gram の出現率が特に低いが，これは n-gram への重み付けが 1-best 以外の候補によっても行われ，1-best に含まれる n-gram が相対的に少なくなるからである．

条件付き確率に基づく手法は，識別言語モデルによる手法に比べて n-gram の出現回数，誤り箇所選択率の両方で優位な結果を示していることが分かる．

5.4　評価尺度の変更と誤り傾向

n-best による識別言語モデルの学習の際，オラクル文の選択に用いる評価尺度を変更することで，分析対象とする n-gram の種類を変更することができる．本研究では評価尺度を BLEU と RIBES の 2 通りで行い，比較を行った．その結果選択された n-gram の内訳を表 6 に示す．表を見ると，BLEU で最適化された識別言語モデルでは，RIBES に比べて削除誤りをより多く検出していることが分かる．これは，BLEU では短い翻訳に対して大きい罰則を与えていることから自然な結果と言える．一方の RIBES で最適

表 6 異なる尺度で最適化した場合に見つかった誤り．太字は高いスコア．

種類	+BLEU	+RIBES
誤り箇所選択率	0.598	0.529
文脈依存置換	**0.332**	0.216
文脈非依存置換	**0.067**	0
挿入	0.164	**0.285**
削除	**0.245**	0.125
並べ替え	0.192	**0.289**
活用	0	**0.086**

化された識別言語モデルでは，並べ換え誤りをより多く捉えていることが分かるが，これは RIBES が BLEU に比べて並べ換え誤りに敏感な評価尺度であることが理由である．

本実験から，識別言語モデルの最適化に使用する評価尺度を変更することで，異なる種類の誤りを検出できることが分かる．

6. まとめ

本研究では，n-gram の出現頻度に基づく誤り選択手法を示し，識別言語モデルに基づく手法との比較を行った．その結果，条件付き確率に基づく手法が，n-best による識別言語モデルに基づく手法に比べ，誤り箇所選択率と選択された n-gram の出現回数で優位な結果を示すが，削除誤りについては，n-best による識別言語モデルに基づく手法が検出能力で優位に立つことを示した．

今後の課題として，構文情報や品詞情報を利用した誤り分析への応用，誤り再現率（recall）の評価，誤り箇所選択を利用した実際の誤り修正での利用価値を示すことが挙げられる．また，n-best による識別言語モデルに基づく手法では，オラクル文の選択に異なる評価尺度を利用することで異なる種類の誤りが検出されることが分かったが，出現頻度に基づく 3 つの手法でも同様の現象が起こると考えられるため，今後調査する．

参考文献

[1] Church, K. W. and Hank, P.: Word association norms, mutual information, and lexicography, *Computational Linguistics*, Vol. 10, pp. 22–29 (1990).
[2] Collins, M.: Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms, *Proc. EMNLP*, pp. 1–8 (2002).
[3] Duchi, J. and Singer, Y.: Efficient Online and Batch Learning using Forward Backward Splitting, *Journal of Machine Learning Research*, Vol. 10 (2009).
[4] Isozaki, H., Hirao, T., Duh, K., Sudoh, K. and Tsukada, H.: Automatic Evaluation of Translation Quality for Distant Language Pairs, *Proc. EMNLP*, pp. 944–952 (2010).
[5] Lin, C.-Y. and Och, F. J.: Orange: a method for evaluating automatic evaluation metrics for machine translation, *Proc. COLING*, pp. 501–507 (2004).
[6] Mackay, D. J. and Petoy, L. C. B.: A Hierarchical Dirichlet Language Model, *Natural Language Engineering*, Vol. 1 (1995).
[7] Neubig, G.: The Kyoto Free Translation Task, 入手先 ⟨http://www.phontron.com/kftt⟩ (2011).
[8] Neubig, G.: Travatar: A Forest-to-String Machine Translation Engine based on Tree Transducers, *Proc. ACL* (2013).
[9] Och, F. J.: Minimum Error Rate Training in Statistical Machine Translation, *Proc. ACL* (2003).
[10] Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J.: BLEU: a method for automatic evaluation of machine translation, *Proc. ACL*, pp. 311–318 (2002).
[11] Popovic, M. and Ney, H.: Towards automatic error analysis of machine translation output, *Computational Linguistics*, pp. 657–688 (2011).
[12] Roark, B., Saraclar, M. and Collins, M.: Discriminative n-gram language modeling, *Computer Speech & Language*, Vol. 21, No. 2, pp. 373–392 (2007).
[13] Teh, Y. W., Jordan, M. I., Beal, M. J. and Blei, D. M.: Hierarchical Dirichlet processes, *Journal of the American Statistical Association*, Vol. 101, No. 476 (2006).
[14] Tsuruoka, Y., Tsujii, J. and Ananiadou, S.: Stochastic Gradient Descent Training for L1-regularized Log-linear Models with Cumulative Penalty, *Proc. ACL*, pp. 477–485 (2009).
[15] Vilar, D., Xu, J., D'Haro, L. F. and Ney, H.: Error analysis of statistical machine translation output, *Proc. LREC*, pp. 697–702 (2006).
[16] 赤部晃一，Neubig, G.，Sakti, S.，戸田智基，中村　哲：識別言語モデルによる機械翻訳システムの誤り分析とシステム間比較，言語処理学会第 20 回年次大会 (NLP2014)，北海道，pp. 959—-962 (2014).