Vol.2012-NL-207 No.9
2012/7/27

大規模語彙的知識に基づく受身形と能動形の表層格の対応付け

笹野 遼平[1]　河原 大輔[2]　黒橋 禎夫[2]　奥村 学[1]

概要：本稿では，同一の用言の対応する受身形と能動形の格の用例や分布の類似性に着目し，Web から自動獲得した大規模格フレームと，少数の受身形と能動形の格の変換規則を用いることで，受身形と能動形の表層格の対応付けに関する知識を自動獲得する手法を提案する．さらに，自動獲得した知識を受身文の能動文への変換における格変換タスクに適用することにより，その有用性を示す．

Case Alternations between the Passive and Active Voices on the Basis of Large Lexical Knowledge

Ryohei Sasano[1]　Daisuke Kawahara[2]　Sadao Kurohashi[2]　Manabu Okumura[1]

Abstract: We propose a method for automatically acquiring knowledge about case alternations between the passive and active voices. Our method leverages large lexical case frames obtained from large Web corpus, and several alternation patterns. We then use the acquired knowledge to a case alternation task and show the usefulness of the acquired knowledge.

1. はじめに

テキスト中に出現した述語の格構造を認識する処理は述語項構造解析や格解析などと呼ばれ，計算機によるテキスト理解のための重要な１ステップであると言える．しかし，一口に “格” と言っても，出現形に対する表層格や能動形に対する表層格，さらには深層格など複数の表現レベルが存在し，どの表現レベルが適しているかは使用するコーパス[*1]やタスクにより異なっている．

格構造を表層格構造で扱う利点としては，表層格はテキスト中に格助詞として明示的に出現することから，“格” を定義する必要がないこと，述語ごとに取り得る格をコーパスから自動獲得することが可能なことなどが挙げられる．さらに，出現形に対する表層格構造で扱う利点としては，能動形格構造に現れない使役文におけるガ格や一部の受身文のガ格を自然に扱えること，先行する述語のガ格の格要素が，後続する述語でもガ格の格要素となりやすいなどといった談話的な情報が自然に利用できることなどが挙げられる．特に後者はゼロ照応解析において重要な手掛りになることが知られており [1], [2]，省略された項の解析も含む高精度な述語項構造解析の実現のためには，格構造を出現形の表層格で扱うのが望ましいと考えられる．

一方，テキストの意味を考える上では，出現形に対する表層格解析では不十分な場合がある．たとえば (1)，(2) のような文を考えると，出現形の表層格としては (1) の「男」と (2) の「海」は同じニ格となっているが，前者は能動形ではガ格 (能動主体) となるのに対し，後者は能動形においてもニ格であり，その意味役割は異なっている．

(1)　女性が男に突き落とされた．

(2)　女性が海に突き落とされた．

(3)　男が女性を海に突き落とした．

また，例文 (3) は (1)，(2) の 2 文が表わす内容を含意していると考えられるが，出現形に対する表層格解析だけで

[1] 東京工業大学, Tokyo Institute of Technology
[2] 京都大学, Kyoto Uniersity

[*1] 京都大学コーパス [3] では出現形の表層格情報，NAIST テキストコーパス [4] では能動形の表層格情報が付与されている．

はこれらの含意関係を認識することはできない．このため，含意関係認識や情報検索などのタスクでは，能動形の格構造や深層格構造などといった，より深い格構造を扱うことが望ましいと言える．

そこで，まず出現形における表層格解析を行い，その結果をより深い格構造に変換することを考える．このような手順を用いることで，談話的な情報を自然に取り入れながら，含意関係認識や情報検索などのタスクにも有用な能動形格構造を扱うことができると考えられる．本研究ではこのうち出現形から能動形への格構造変換，特に受身形から能動形への格構造変換に焦点を当てる．

受身形と能動形の格の変換を扱った研究としては，近藤らの研究 [5]，村田らの研究 [6] がある．近藤ら [5] は単文の言い換えの 1 タイプとして，受身形と能動形の格の変換を扱っており，動詞のタイプや格パターンなどをもとに作成した 7 種類の変換パターンを用いて格の変換を行っている．動詞のタイプとしては，「比較動詞」，「授受動詞」，「対称動詞」，「一般動詞」の 4 種類を定義しており，IPAL 基本動詞辞書 [7] をもとに 1,564 エントリからなる動詞辞書を作成し，使用している．以下では，この辞書のことを VDIC 辞書と呼ぶ．

村田ら [6] は京都大学コーパス [3] の社説を除く約 2 万文において，受身形で出現した述語に係る格助詞を対象に，述語を能動形に変換した場合の格を付与した学習データを作成し，SVM[8] を用いた機械学習により受身形と能動形の格を変換する手法を提案している．学習に使用する素性には，関係する動詞や体言，格助詞の出現形や品詞情報などといった情報に加え，IPAL 基本動詞辞書や VDIC 辞書から得られる情報を使用している．

このように，既存の研究は人手で整備された大規模な語彙的リソースや，人手で作成した大規模な学習データを利用している．しかしながら，例文 (1) と (2) のように述語と表層格が一致していても原形に戻した場合の格が異なる場合があることからも分かるように，格の対応は用言ごと，用法ごとに異なっており，網羅的な対応付けに関する知識を人手で記述することは現実的ではないと言える．

一方，用言・用法ごとに異なるこれらの知識を自動獲得することを考えると，大量の生テキストから変換パターンの手掛りを得ることが可能だと考えられる．たとえば例文 (3) のような文を収集することができれば，例文 (1) におけるニ格が能動形ではガ格となるのに対し，例文 (2) ではニ格のままであることが，格要素の類似性を考慮することにより自動的に認識できると考えられる．

そこで，本研究では，同一の用言の対応する受身形と能動形の格の用例や分布の類似性に着目し，Web から自動獲得した大規模格フレームと，少数の受身形と能動形の格の変換規則を用いることで，受身形と能動形の表層格の対応付けに関する知識の自動獲得を行う．

2. 受身形と能動形の格の対応パターン

受身文は，何を主語として表現するかによって，直接受身文，間接受身文，持ち主の受身文の 3 つのタイプに分けられる [9]．

直接受身文とは，対応する能動文でヲ格やニ格で表される人や物を主語として表現する受身文であり，以下の例文 (4)〜(7) のように能動主体は基本的に「に」，「によって」，「から」，「で」のいずれかにより表される．また，直接受身文でガ格として表される名詞は，以下の例文 (4)，(6) のように基本的に能動文のヲ格，または，ニ格のいずれかに対応している．

(4) [**受身形**]: 女性 が 男性 に 突き落とされた．
[**能動形**]: 男性 が 女性 を 突き落とした．

(5) [**受身形**]: 原因 が 研究 によって 解明された．
[**能動形**]: 研究 が 原因 を 解明した．

(6) [**受身形**]: 私 が 彼 から 頼まれた．
[**能動形**]: 彼 が 私 に 頼んだ．

(7) [**受身形**]: 大半 が 推進派 で 占められた．
[**能動形**]: 推進派 が 大半 を 占めた．

間接受身文とは，対応する能動文の表す事態には直接的に関わっていない人物を主語とし，その人物が事態から何らかの影響を被っていることを表現する受身文であり，迷惑の受身文とも呼ばれる．例文 (8) のように間接受身文の能動主体は基本的に「に」によって表わされ，間接受身文でガ格として表される名詞は能動文では出現しない．

(8) [**受身形**]: 彼 が 雨 に 降られたのは · · ·
[**能動形**]: 雨 が 降った．

持ち主の受身文とは，ヲ格名詞やニ格名詞などで表されていた物の持ち主を主語とし，能動文で主語として表されていた名詞を主語でない項として表現する受身文である．例文 (9) のように持ち主の受身文の能動主体は基本的に「に」によって表わされ，持ち主の受身文でガ格として表される名詞は能動文ではヲ格名詞やニ格名詞にノ格で係る名詞句として出現する場合がある．

(9) [**受身形**]: 友人 が カード を 泥棒 に 盗まれた．
[**能動形**]: 泥棒 が 友人 の カード を 盗んだ．

以上のように，受身文には 3 つのタイプがあるものの，いずれの場合も格変換の対象となるのは，能動文で主体として表される要素と，受身文においてガ格で表現される要素の 2 つのみであり，また，前者の能動文における対応先は「に」，「によって」，「から」，「で」，後者の受身文におけ

る対応先は「を」,「に」,「の」，または，対応先なしのいずれかであり，これらの組み合わせからなる格の対応パターンを考えれば十分であると言える．

3. Webから自動獲得した大規模語彙的知識

本研究では用言ごとの格構造に関する大規模語彙的知識として，河原らの手法 [10] を用いて Web テキスト 69 億文から自動構築した格フレームを使用する．この Web テキストは，約 10 億 Web ページから日本語文を抽出し，重複を除いた結果得られたものである．

この格フレームは用言ごと，用法ごと，出現形ごとに構築されており，それぞれ取り得る格とその用例，および，各用例の出現回数がまとめられている．たとえば「突き落とされる」という用言に対しては 59 個の格フレームが構築されており，その中には以下の例のように[*2]，ニ格が場所を表す格フレームと，能動主体を表す格フレームが含まれている．

(10) **「突き落とされる」の格フレーム 4:**
{ 女性:5, 僕:2, 彼:2, ⋯ } **が**
{ 海:229, 川:115, 池:51, ⋯ } **に** 突き落とされる

(11) **「突き落とされる」の格フレーム 5:**
{ 京子:3, 女性:1, 監督:1, ⋯ } **が**
{ 誰:143, 何者:85, 男:23, ⋯ } **に** 突き落とされる

また，格として収集する対象としては格助詞を伴なって直接用言に係る要素に加えて，「によって」などの一部の複合辞や，持ち主の受身文のガ格になりうることから用言の直前格要素にノ格で係る要素も収集の対象としている．このため以下の例のように，ニヨッテ格やノ格を含む格フレームも生成される．

(12) **「解明される」の格フレーム 1:**
{ 謎:1998, メカニズム:804, 原因:734, ⋯ } **が**
{ 研究:29, 生物学:27, 進歩:15, ⋯ } **によって**
解明される

(13) **「盗む」の格フレーム 3:**
{ 子供:23, 誰:16, 泥棒:8, ⋯ } **が**
{ 親:165, 人:98, 家:58, ⋯ } **の**
{ 金:4163, 現金:951, 金品:681, ⋯ } **を** 盗む

4. 受身形と能動形格フレームの対応付け

本研究では，ある受身形の格フレームが与えられた場合に，対応する能動形用言の格フレームと適切に対応付けることを目的とする．1 つの用言に対し，複数の格フレームが構築されることから，複数ある能動形格フレームから最適な格フレームを選択した上で，それぞれの格同士の最適な対応付けを行う必要がある．

4.1 対応付けの手掛かり

本研究では対応付けの手掛かりとして，対応する格の用例集合間の意味的な類似度 sim_{SEM} と，対応する格の出現頻度の分布の類似度 sim_{DIST} の 2 つを利用し，対応付け A のスコアをこれらの積により定義する．

まず，格の用例集合 C_1，C_2 間[*3]の意味的な類似度を，ある単語と共起する単語の分布の類似度から計算された単語間の分布類似度 $sim(w_1, w_2)$[11] をもとに，以下の式により定義する．

$$sim_{SEM}(\mathrm{C}_1, \mathrm{C}_2) = \frac{1}{2}(sim_{DRCT}(\mathrm{C}_1, \mathrm{C}_2) + sim_{DRCT}(\mathrm{C}_2, \mathrm{C}_1))$$

ただし，

$$sim_{DRCT}(\mathrm{C}_1, \mathrm{C}_2) = \frac{1}{|\mathrm{C}_1|} \sum_{w_1 \in \mathrm{C}_1} \max_{w_2 \in \mathrm{C}_2} (sim(w_1, w_2))$$

さらに，対応元格フレームにおける n 番目の格の用例集合 $C_{1,n}$ が対応付けられた格の用例集合を $C_{2,align(n)}$ と表した場合，格フレームの対応付け全体 $A = \{C_{1,1} \rightarrow C_{2,align(1)}$ $C_{1,2} \rightarrow C_{2,align(2)}, \cdots, C_{1,N} \rightarrow C_{2,align(N)}\}$ に対する意味的な類似度 (sim_{SEM}) を以下の式により定義する．

$$sim_{SEM}(A) = \frac{1}{N} \sum_{i=1}^{N} sim_{SEM}(C_{1,i}, C_{2,align(i)})$$

一方，類似度 sim_{DIST} は対応する格同士の出現頻度の分布は似ているという仮定に基づく指標であり，ベクトル $(|C_{1,1}|, |C_{1,2}|, \ldots, |C_{1,N}|)$ とベクトル $(|C_{2,align(1)}|, |C_{2,align(2)}|, \ldots, |C_{2,align(N)}|)$ のコサイン類似度として定義する．

(14) **「選ばれる」の格フレーム 1:**
{ 選手:1119, 作品:983, 私:232, ⋯ } **が**
{ 代表:18295, 選手:9661, 百選:7024, ⋯ } **に**
{ 作品:5, 市長:3, 選手:2, ⋯ } **を** 選ばれる
それぞれの用例数の合計は順に $(17722, 122273, 96)$

(15) **「選ぶ」の格フレーム 13:**
{ 私:22, 先生:18, 誰:14, ⋯ } **が**
{ 優秀賞:42, シングル:17, 自由曲:17, ⋯ } **に**
{ 曲:16666, 作品:9967, 漫画:3820, ⋯ } **を**選ぶ
それぞれの用例数の合計は順に $(382, 800, 33338)$

たとえば (14) に示す「選ばれる」の格フレーム 1 を (15) に示す「選ぶ」の格フレーム 13 に対応付ける場合を考える

[*2] 本稿では，格フレームを示す場合，主要な格のみを抜粋して示す．また，用例の後の数字はその用例の出現回数を表す．

[*3] ここで，用例集合 C は単語ごとではなく単語の出現ごとの用例集合とする．したがって類似度 sim_{SEM} は単語単位で考えると出現頻度で重み付けされていることになる．

表 1 能動形格フレームへの対応付けアルゴリズムの疑似コード

Input: ある受身形格フレームと，それに対応する能動形格フレームの集合
Output: 対応する能動形格フレームと，格の対応情報 ($max\_pattern$)

$max\_score = 0$, $max\_pattern = ()$
foreach $cf \in$ (対応する能動形格フレームの集合)
  foreach $ga\_to \in$ 受身形ガ格の能動形における対応先の候補: { を, に, の, NIL}
    foreach $to\_ga \in$ 能動形におけるガ格の対応先候補: { に, によって, から, で, NIL}
      if (illegal(ga_to, to_ga)) **continue**　(※不適切な格対応の組み合わせをフィルタリング)
      $score = sim_{SEM}(A) \times sim_{DIST}(A)$　(※ A は cf, ga_to, to_ga により生成される格の対応付け)
      if ($score > max\_score$) **then**
        $(max\_score, max\_pattern) \leftarrow (score, pattern(A))$

と，対応する格の用例集合間の意味的な類似度のみを考慮した場合は { ニ格 → ガ格, ガ格 → ニ格, ヲ格 → ヲ格 } という対応付けの方が，{ ニ格 → ニ格, ガ格 → ヲ格, NIL→ ガ格 } という対応付けよりも高いスコアとなるが，対応する格の出現頻度の分布の類似度を計算すると，前者は $(122273, 17722, 96)$ と $(382, 800, 33338)$ のコサイン類似度，後者は $(122273, 17722, 0)$ と $(800, 33338, 382)$ のコサイン類似度となり，後者の方がはるかに大きな値となることから，最終的に後者の対応付けの方が優先して選択される．

4.2 対応付けアルゴリズム

ある受身形格フレームが与えられた場合の，能動形格フレームへの対応付けアルゴリズムの疑似コードを表 1 に示す．

基本的に考えうるすべての格フレームと格の対応付けパターンに対して，対応する格の意味的な類似度 sim_{SEM} と対応する格の出現頻度の分布の類似度 sim_{DIST} の積を対応付けのスコアとして計算し，最大となる格の変換パターンを出力する．ただし，開発データがある実験設定では，格の対応付けパターンごとの出現しやすさを考慮し，スコアに補正を加える．

能動主体への対応付け候補としては「に」,「によって」,「から」,「で」に加え，例文 (16) における「簡素化」のように，多くの場合，能動主体が省略される述語も存在することから「NIL(対応付けなし)」も加えた 5 つを考える．

(16) 免許の取得方法が簡素化される．

また，不適切な格対応の組み合わせはスコア計算前にフィルタリングする．具体的にフィルタリングするのは，格変換の結果，同一の格が重複してしまう場合と，受身形格フレームにヲ格が存在しない場合に受身形ガ格の変換先としてノ格が選択された場合の 2 通りである．

計算時間を短縮するため，能動形の格フレームは頻度順にソートし全体の 80% をカバーしたところで計算を打ち切っている．同様に，対応する格の意味的な類似度 sim_{SEM} を計算する際も，それぞれの格の用例のうち頻度上位 10 用例のみから類似度を計算している．

5. 自動獲得した受身形・能動形格フレームの対応付け知識に基づく格の変換実験

自動獲得した対応付け知識の定量的な評価を行うため自動獲得した対応付け知識を用いた受身文から能動文への格の変換実験を行なった．

5.1 実験設定

実験データには，村田ら [6] が使用しているデータを利用した．ただし，村田らの実験設定では持ち主の受身文においてガ格で表わされる名詞の能動文における格としてノ格を認めておらず，カラ格やヲ格となっているため，持ち主の受身文のガ格と考えられる 5 事例の変換後の格をノ格に変更した．また，それ以外にも誤っている考えられる 16 事例に修正を加えた上で使用した．

村田らのデータは，受身文に出現した格助詞をそれぞれ 1 つの事例とし，全部で 3,576 事例からなっている．村田らはこのデータを 1,788 個ずつに分け，それぞれクローズドデータ，オープンデータと呼んでいる．本研究でも村田らと同様に分割して使用し，クローズドデータを用いて適切な対応付けのスコアの補正方法を求め[*4]，補正したスコアを用いて格フレームの対応付けを行なった．このため，提案手法においてクローズドデータを使った実験はオープンな条件ではない．

また，村田らのデータには一部，複数の格が正解として付与されている場合があるが，本研究では正解として付与されている格のうち 1 つを出力できれば正解とみなすという評価基準[*5]を採用した．

[*4] 具体的には，受身形のガ格が能動形で「を」に対応付けられた場合のスコアを 2.0 倍,「NIL」に対応付けられた場合のスコアを 0.5 倍，能動形におけるガ格の対応先が「から」となっている場合のスコアを 1.5 倍,「で」となっている場合のスコアを 0.5 倍するという補正を行った．
[*5] 村田らの論文 [6] における評価 B

5.2 格の変換アルゴリズム

受身形における格から能動形における格への変換は次の手順により行なった．以下ではこの手法を**提案手法 (教師なし)** と呼ぶ．

(i) 格フレームに基づく構文・格解析器 KNP [*6] を用いて入力文の出現形格解析を行う．
(ii) 格解析の結果，選択された受身形格フレームを，能動形格フレームに対応付ける．
(iii) 格フレームの対応付け情報を利用し，受身文の格を能動文における格に変換する．この際，出現格がニ格であった場合でも，格解析の結果，時間格や修飾格に対応付けられている場合は，格変換を行わない．

ただし，村田らのデータに 32 事例含まれる以下のような使役受身文については，能動形の文型に付け加えられた使役者を表すニ格を除き，格が変化しないことから，出現格をそのまま出力する．

(17) 人間が窮地に立たされれば、···

また，比較対象として以下の 3 つの手法を用いた．1 つ目は村田らの論文 [6] においてベースラインとして用いられている手法の 1 つで，格ごとにクローズドデータにおいて最も頻度の高い格に変換するという手法である．本稿ではこの手法を**最頻変換モデル**と呼ぶ．

2 つ目の比較対象として，対応する格の出現頻度の分布の類似度 sim_{DIST} の有効性を確認するため，対応する格の意味的な類似度 sim_{SEM} のみを用いて対応付けのスコアを計算する手法を用いる．以下では，この手法を**分布類似度不使用モデル**と呼ぶ．

最後に，3 つの目の比較対象として機械学習に基づく**村田らの手法**を用いる．クローズドデータを評価対象とする場合はオープンデータを，オープンデータを評価対象とする場合はクローズドデータを学習データとして使用した．本研究では村田らの提案手法のうちオープンデータに対し最も高い精度を実現している手法，すなわち，学習データを出現形の格ごとに分割し，素性選択は行わないモデルを使用した．村田らの手法ではクローズドデータは素性選択にのみ使用しているため，素性選択を用いない本稿の実験設定では村田らの手法はクローズドデータに対してもオープンな条件となっている．また，本実験では SVM の実装として YamCha[*7]を 2 値分類器として使用した．

さらに，学習ベースの手法における自動獲得した受身形・能動形格フレームの対応付け知識の有用性を確認するため，提案手法 (教師なし) の出力結果を村田の手法に素性として加えた手法での実験も行なった．以下では，この手法を**提案手法 (教師あり)** と呼ぶ．

[*6] http://nlp.ist.i.kyoto-u.ac.jp/index.php?KNP
[*7] http://chasen.org/~taku/software/yamcha/

表 2 受身形から能動形格変換実験の結果

手法	クローズドデータ	オープンデータ
機械学習に基づかない手法		
最頻変換	0.8842 (1581/1788)	0.8826 (1578/1788)
分布類似度不使用	0.9217 (1648/1788)	0.9234 (1651/1788)
提案手法 (教師なし)	**0.9290** (1661/1788)	**0.9279** (1659/1788)
機械学習に基づく手法		
村田モデル	0.9446 (1689/1788)	0.9441 (1688/1788)
提案手法 (教師あり)	**0.9581** (1713/1788)	**0.9530** (1704/1788)

5.3 実験結果と考察

実験結果を表 2 に示す[*8]．提案手法 (教師なし) が最頻変換モデルより高い精度での変換精度を実現していることから，大規模語彙知識に基づき受身形と能動形の格を対応づける手法の有効性が確認できる．また，分布類似度不使用モデルよりも提案手法 (教師なし) の方が高い精度となっており，対応する格の出現頻度の分布の類似度 sim_{DIST} は，格の対応付けを行う上で一定の手掛かりとなっている考えられる．ただし，その差は僅かであり効果は限定的であると言える．

機械学習に基づく手法の精度を比較すると，提案手法 (教師なし) の出力結果を，村田の手法に素性として加えることで，解析精度が向上していることが確認できる．このことから自動獲得した対応付け知識は IPAL 基本動詞辞書や VDIC 辞書からは得られない有用な情報を含んでいると考えられる．

村田らは，素性 F25(近藤法 [5] で変換の際に用いた格変換規則) を使用しなかった場合に，村田モデルの精度が大きく低下することを報告している [6]．この素性は VDIC 辞書に含まれている用言に対象は限定されるものの，用言ごとに人手で記述した変換パターンに関する情報とみなすことができ，本研究で自動獲得した対応付け知識と重複する素性であることが予想される．そこで，この素性を使用しない場合における，自動獲得した対応付け知識の効果を確認するため，素性 F25 を使用しなかった場合の精度を調査した．結果を表 3 に示す．

村田モデル，提案手法 (教師あり) ともに表 2 に示した全ての素性を用いた場合と比べ精度が大きく低下しているが，その下がり幅は提案手法の方が小さい．このことから，本研究で自動獲得した対応付け知識は，素性 F25 に近い情報を含んでいる可能性があると考えられる．

[*8] 最頻変換モデル，村田モデルともに村田ら [6] が報告している数値より僅かに高い数値となっているが，これはデータに修正を加えたためである．

表 3 村田モデルから素性 F25 から除いた場合の精度

手法	クローズドデータ	オープンデータ
村田モデル	0.9262 (1656/1788)	0.9234 (1651/1788)
提案手法 (教師あり)	0.9402 (1681/1788)	0.9368 (1675/1788)

表 4 持ち主の受身文の解析例

入力テキスト:
… 松樹さん が 金属バットで頭を殴られ、…

KNP による解析で選択された受身形格フレーム:
「殴られる」の格フレーム 2:
{ 何者:2, 部員:1, リサ:1, …} **によって**
{ 女性:5, 女児:4, 小沢:4, …} **が**
{ 頭:3944, 顔:1186, 頭部:840, …} **を**
{ 鈍器:84, バット:45, 拳:43, …} **で** 殴られる

対応付けられた能動形格フレーム:
「殴る」の格フレーム 2:
{ 男:51, 拳:30, 誰:23, …} **が**
{ 自分:360, 私:223, 相手:192, …} **の**
{ 頭:5424, 顔:3215, 顔面:1529, …} **を**
{ 拳:316, 平手:157, 拳骨:126, …} **で** 殴る

格の対応関係:
{ によって → が, が → の, を → を, で → で }

続いて，持ち主の受身文のガ格の変換先であるノ格の変換精度に注目する．このノ格は村田らのデータに新たに追加した格であり，全体で 5 つ出現する．提案手法 (教師なし) の解析では，このうち 4 つが正しくノ格と解析できており，また，全体でノ格に変換した事例は 5 つであったことから適合率も 4/5 と非常に高い精度で解析できていた．表 4 に例として正しく変換できた持ち主の受身文の例を，使用された格フレーム，および，その対応関係とともに示す．

一方，主な解析誤りの要因，既知の問題点としては以下の 4 つが挙げられる．

格を 1 対 1 で対応付けている 「採用される」のニ格，ニヨッテ格，カラ格のように受身形における複数の格が，能動主体を表わす場合も，いずれか 1 つしか能動主体と対応付けられないため，対応付けられなかった格の変換が正しくできない

受身と尊敬の「れる/られる」を区別していない 格フレーム構築時に「れる/られる」の用法を考慮していないため，「使われる」などのように本来，ヲ格を持たない受身形用言にヲ格が生成されてしまい，格フレームが適切に対応付けられない

KNP による格フレームの選択精度の問題 格フレーム同士は正しく対応付けられていても，KNP が適切な格フレームを選択しないと誤った格を出力してしまう

複数のニ格を取る受身形格構造に対応していない 「彼 に 家 に 帰られる」などのように，能動主体を表すニ格と，能動形ニ格の 2 つのニ格を取る場合，これらの格を区別せず，まとめてしまっている

6. おわりに

本稿では，Web から自動獲得した大規模格フレームと，少数の受身形と能動形の格の変換規則を組み合わせることで，受身形と能動形の表層格の対応付けに関する知識の自動獲得を行う手法を提案した．また，獲得した知識を受身文の能動文への変換における格変換タスクに適用することにより，その有用性を示した．今後の課題としては，使役形と能動形の対応付けや，授受動詞間の対応付けが挙げられる．本稿で提案した手法は基本的に学習データを必要としないことから，考えうる格の変換パターンさえ記述すれば，自動的に対応を取ることが可能である．また，本稿では受身文において格助詞が明示された項のみを格変換の対象としているが，今後は提題助詞の使用や，被連体修飾要素としての出現，ゼロ代名詞化などにより格が明示されていない場合も解析の対象としていく予定である．

参考文献

[1] Iida, R., Inui, K. and Matsumoto, Y.: Zero-Anaphora Resolution by Learning Rich Syntactic Pattern Features, *ACM Transactions on Asian Language Information Processing (TALIP)*, Vol. 6, p. Article 12 (2007).
[2] 笹野遼平，黒橋禎夫：大規模格フレームを用いた識別モデルに基づく日本語ゼロ照応解析，情報処理学会論文誌，Vol. 52, No. 12, pp. 3328–3337 (2011).
[3] 黒橋禎夫，長尾　眞：京都大学テキストコーパス・プロジェクト，言語処理学会 第 3 回年次大会発表論文集，pp. 115–118 (1997).
[4] Iida, R., Komachi, M., Inui, K. and Matsumoto, Y.: Annotating a Japanese Text Corpus with Predicate-Argument and Coreference Relations, *Proc. of ACL'07 Workshop: Linguistic Annotation Workshop*, pp. 132–139 (2007).
[5] 近藤恵子，佐藤理史，奥村　学：格変換による単文の言い換え，情報処理学会論文誌，Vol. 42, No. 3, pp. 465–477 (2001).
[6] 村田真樹，金丸敏幸，白土　保，井佐原均：入力文の格助詞ごとに学習データを分割した機械学習による受身文の能動文への変換における格助詞の変換，システム制御情報学会論文誌，Vol. 21, No. 6, pp. 165–175 (2008).
[7] 情報処理振興事業協会技術センター：計算機用日本語基本動詞辞書 IPAL (1996).
[8] Vapnik, V.: *The Nature of Statistical Learning Theory*, Springer (1995).
[9] 日本語記述文法研究会（編）：現代日本語文法 2, 第 4 部 ヴォイス，くろしお出版 (2009).
[10] 河原大輔，黒橋禎夫：格フレーム辞書の漸次的自動構築，自然言語処理，Vol. 12, No. 2, pp. 109–131 (2005).
[11] 柴田知秀，黒橋禎夫：超大規模ウェブコーパスを用いた分布類似度計算，言語処理学会第 15 回年次大会，pp. 705–708 (2009).