Vol.2011-NL-203 No.2
2011/9/16

述語項構造の共起情報と格フレームを用いた事態間知識の自動獲得

柴田知秀[†1]　黒橋禎夫[†1]

本論文では述語項構造の共起情報と格フレームを用いることにより，大規模コーパスから事態間知識を獲得する手法について述べる．述語項構造の共起情報はアソシエーション分析を用いて効率的に計算し，述語に対する項の必須性の判断を行なう．そして，格フレームを用いて項のアライメントをとる．16 億文からなる Web コーパスを用いて実験を行なったところ，事態ペアの獲得精度が 96%，項のアライメント精度が 79.1%であり，獲得された事態ペアの数は約 2 万となった．

Acquiring Strongly-related Events using Predicate-argument Co-occurring Statistics and Caseframe

Tomohide Shibata [†1] and Sadao Kurohashi[†1]

This paper proposes a method for automatically acquiring strongly-related events from a large corpus using predicate-argument co-occurring statistics and caseframe. The co-occurrence measure is calculated using an association rule mining method, and the importance of an argument for each predicate-argument is judged. Then, the argument alignment in the pair of predicate-arguments is performed by using a caseframe. We conducted experiments using a Web corpus consisting of 1.6G sentences. The accuracy for the extracted event pairs was 96%, and the accuracy of the argument alignment was 79.1%. The number of acquired event pairs was about 20 thousands.

†1 京都大学
Kyoto University

1. はじめに

自然言語理解のためには様々な言語知識が必要となる．そのような知識の一つに述語と項の関係がある．これは格フレームという形でコーパスから自動獲得され，構文解析などで有効性が示されている[1)]．さらに述語項構造の間の知識が重要となる (以降，事態間知識と呼ぶ)．事態間知識は共参照解析[2)] や照応解析[3)] などの基礎解析や対話などのアプリケーションで有用である．

本論文では，関連の強い事態ペアを以下のような形で獲得する．

PA_1		PA_2
A_1:{ 人, 男, ...} ガ A_2:{ 財布, ...} ヲ　拾う	⇒	A_1:{ 人, 男, ...} ガ A_2:{ 財布, ...} ヲ　届ける A_3:{ 警察 } ニ

この例では，項 A_1 と A_2 は述語項構造 PA_1，PA_2 ともに出現している一方で，項 A_3 は PA_2 にしか出現しておらず，PA_2 の述語「届ける」の意味を特定する役割を果たす．事態間知識を獲得する手法として Chambers ら[4),5)] の手法があるが，この手法では共参照関係にある語 (アンカー) を手がかりとしており，一方の事態にのみ出現する項は抽出することができない．

上記のような事態ペアをテキストから獲得するために事態ペアの共起情報を利用する．上記の事態ペアを表す文は以下のような形で出現する．

(1) a. 人が 財布を 拾って 警察に 届ける
　　b. 財布を 拾って 警察に 届ける

日本語では省略が頻繁に用いられるため，文 (1-a) において「人」と「財布」が PA_2 では省略され，また，エージェントはより頻繁に省略されるため，文 (1-b) においては PA_1 のガ格も省略されている．Chambers らの手法のようにアンカーを手がかりにするとこれらの文の出現からは上記のような事態間知識を獲得するのは難しい．

そこで提案手法では 2 段階で事態間知識を獲得する．PA_1 の「A_2: { 財布, ...} ヲ」や PA_2 の「A_3: { 警察 } ニ」のような述語の意味を特定するような項は少なくとも一方の事態では出現することから，まず，述語項構造の共起情報に基づき関連の強い事態ペアを獲得

する．上記の例では，この段階で「A_2:{ 財布, ...} ヲ 拾う」 ⇒ 「A_3:{ 警察 } ニ 届ける」が獲得される．

次に，格フレームを用いて，項のアライメントをとる．格フレームは用言の意味ごとにとりうる格要素が記述されており，格フレームにおいて格要素の分布の類似性をみることにより項のアライメントをとることができる．上記の例では PA_1 の「A_2:{ 財布, ...} ヲ」は PA_2 のヲ格に対応し，PA_1 の「A_1:{ 人, 男, ...} ガ」と PA_2 のガ格が対応することがわかる．

本論文の構成は以下のとおりである．2 節で関連研究について述べ，3 節で提案手法の概要を示す．4 節で述語項構造ペアの抽出，5 節でアソシエーション分析と述語項構造ペアの共起度計算について述べ，6 節で格フレームに基づく項のアライメントについて述べる．7 節で実験結果を述べる．

2. 関 連 研 究

まず，人手により構築された事態間関係に関するリソースについて述べ，次に，コーパスからの事態間関係の自動獲得手法について述べる．

2.1 人手により構築されたリソース

WordNet は人手で構築された語彙に関するリソースである[6]．WordNet に記述されている関係は同義語・反義語・上位語・下位語だけでなく，因果関係や含意なども含まれる．

LifeNet は人間の日常行動に関する常識を人手により構築したデータベースである[7]．このデータベースは 8 万ノード，41 万リンクからなる．また，EventNet では，Openmind Commonsense Knowledge Base から得た事態間関係よりネットワーク構造を構築している[8]．

近年，Regneri らは Amazon Mechanical Turk を利用して，ある場面での典型的な事態列を記述した知識であるスクリプトを収集し，複数人により記述されたスクリプトを基にグラフ構造を構築している[9]．彼らは例えば「レストランで食事をする」のような 22 個のシナリオに対して 493 個の事態列を構築している．

2.2 コーパスからの事態間知識の自動獲得

様々な事態間関係知識の自動獲得手法が提案されている．一つには推論知識の獲得があげられる．Lin らは依存構造木での二つのパスの分布仮説を考えることにより，推論知識の獲得を行なっている[10]．例えば，“X is the author of Y” と “X wrote Y” のように，X,Y ともに語の出現分布が似ている場合にそれらを推論知識を獲得している．

別のタイプの事態間関係としてはスクリプト知識がある．Chambers らは生コーパスから事態列を獲得している[4),5)]．例えば,「accused X」,「X claimed」,「X argued」,「dismissed X」のような事態列である．この手法ではまず共参照関係にある語を共有して構文的関係を持つ二つの事態を獲得し，相互情報量の高い事態ペアを獲得する．そして，時間の順序関係の推定などを行ない，スクリプト知識の獲得を行なっている．この手法は共参照解析結果に依存しており，省略/照応が頻繁に生じる日本語のような言語には適用しづらいという問題がある．

藤木らはスクリプト知識を日本語新聞テキストから獲得している[12]．テキスト集合から事態列を取り出し，頻度などの情報から典型的な事態列を得ている．

鳥澤は並列構造と動詞-名詞の共起情報を用いて，推論知識を獲得している[13]．阿部らはパターンベースの手法とアンカーベースの手法を組み合わせることにより，事態間知識を獲得している[14]．まず，得たい事態間関係を表すパターンを用いてブートストラップで事態間関係を表す事態ペアの候補を得る．そして，アンカーとなる名詞の出現をチェックすることにより，事態間関係知識を獲得している．鳥澤の手法や阿部らの手法ではアンカーを手がかりとしており，事態の一方のみに出現する項を獲得することができない．

3. 提案手法の概要

図 1 に提案手法の概要を示す．まず，Web コーパスから係り受け関係にある述語項構造ペアを抽出する．そして，相互情報量の高い述語項構造ペアを関連の強い事態ペアとして得る．ここで，例えば,「拾う」を含む述語項構造 PA_1 と「届ける」を含む述語項構造 PA_2 はそれほど関連は強くないが,「財布 ヲ 拾う」を含む述語項構造 PA_1 と「警察 ニ 届ける」を含む述語項構造 PA_2 は強く関連しているといえる．このように述語項構造をどのような単位として扱うかという問題がある．この問題に対して，本研究ではアソシエーション分析を用いる[15]．アソシエーション分析を用いることにより，相互情報量の高い述語項構造ペアを効率的に見つけることができる．

次に，上記で獲得された事態ペアにおいて項のアライメントをとる．述語「拾う」に対して項「財布 ヲ」をとる場合に他の格にどのような用例が出現するかは格フレーム[1)] という形で集められている．図 1 のように,「拾う」の 10 番の格フレームではヲ格に「財布」があり，ガ格には「男」,「女の子」などの用例が集まっている．同様に，述語「届ける」に対して項「警察 ニ」をとる場合，ガ格には「男」,「人」，ヲ格には「財布」,「金」などの用例が集まっている．格フレームの格要素の分布の類似性をみることにより項のアライメントをと

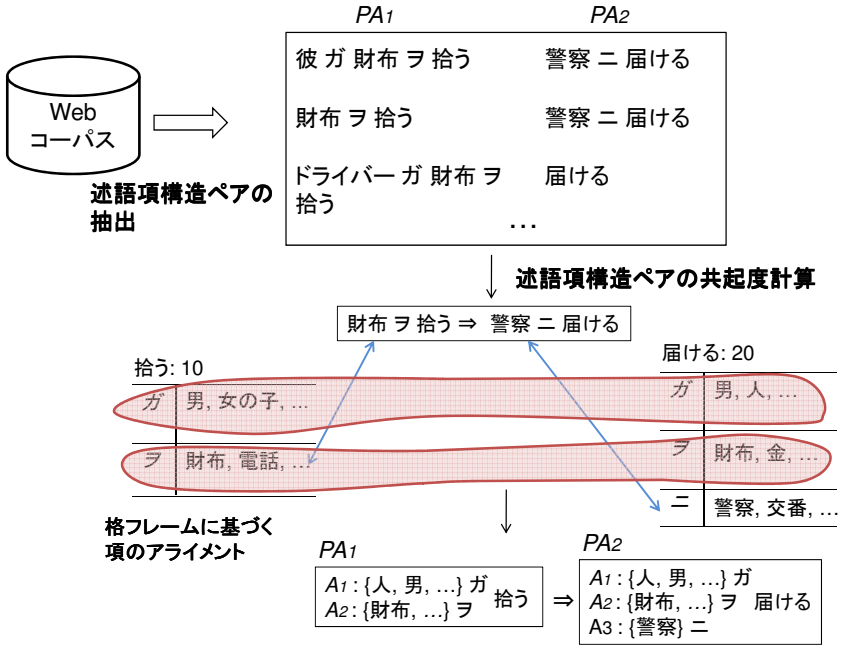

図 1 提案手法の概要

ることができ，この場合，PA_1 のガ格と PA_2 のガ格，PA_1 のヲ格と PA_2 のヲ格が対応することがわかる．

4. 述語項構造ペアの抽出

事態間ペアはテキストにおいて様々な節間関係とともに出現する．例えば,「財布を拾う」と「警察に届ける」という二つの事態は以下のような節間関係で出現する．

(2) a. 財布を拾って 警察に届けた．

b. 財布を拾った ので 警察に届けた．

表 1 節間関係と述語項構造の抽出例

節間関係	手がかり表現を含む文の例	PA_1	PA_2
順接	蜂に刺され**て** 腫れた	蜂ニ 刺される	腫れる
理由	蜂に刺された**ので** 腫れた	蜂ニ 刺される	腫れる
条件	蜂に 刺される**と** 腫れる	蜂ニ 刺される	腫れる
目的	水分を飛ばす**ために** 加熱する	加熱する	水分ヲ 飛ばす
逆接	蜂に刺された**けれど** 腫れなかった	蜂ニ 刺される	腫れる
同時	シャワーを浴び**ながら** 歯を磨く	シャワーヲ 浴びる	歯ヲ 磨く

表 2 単語クラスとそれに属する名詞の例

クラス	名詞
77	蜂, 蚊, ···
105	ドレス, 衣裳, スーツ, ···
502	アドレス, 番号, ID, ···
956	銃撃, 襲撃, ···
1829	研修, インターン, ···
1901	道路, 国道, ···

本研究では，係り受け関係にある大量の述語項構造ペアから事態間知識を獲得する．まず，構文解析結果から係り受け関係にある述語項構造ペアを抽出する．獲得する格要素はガ，ヲ，ニ格とし，また，述語に否定，使役，受身などの素性があれば，述語にフラグとして付与する．また，表 1 に用いた節間関係と述語項構造の抽出例を示す．順接で出現する述語項構造ペアを基準として考え (PA_1 て PA_2)，その他の節間関係での出現はこの形に正規化する．節間関係が「理由」「条件」「同時」の場合はそのまま PA_1，PA_2 とするが，節間関係が「目的」の場合は，PA_2 と PA_1 を逆にし (表中の例では PA_1 を「加熱する」，PA_2 を「水分ヲ 飛ばす」とする)，また，節間関係が「逆接」の場合は PA_2 の否定フラグを反転させる (表中の例では PA_2 を「腫れる」とする)．

項の汎化

データスパースネスを軽減するために，項を単語クラスに汎化する．単語クラスとして風間らの大規模類似語リストを用いる[16]．この単語クラスは動詞-名詞の係り受け関係をクラスタリングして構築されたものであり，単語クラスは 2,000 である．表 2 に単語クラスの例とそれに属する名詞を示す．

単語クラスへの汎化は以下のようにして行なう．抽出された述語項構造ペアにおいて，名詞 n を，最も帰属確率 ($P(c|n)$) の高い単語クラス $\langle c \rangle$ に置換する．例えば，名詞「蚊」はクラス 77 への帰属確率が最も高いため,「PA_1: 蚊に 刺される, PA_2: 腫れる」は「PA_1:

⟨77⟩ に 刺される, PA_2: 腫れる」となり，同様に，名詞「蜂」もクラス 77 への帰属確率が最も高いため,「PA_1: 蜂に 刺される, PA_2: 腫れる」も「PA_1: ⟨77⟩ に 刺される, PA_2: 腫れる」となり，これらの述語項構造ペアを同一視することができる．

5. 述語項構造ペアの共起度計算

4 節で抽出された大量の述語項構造ペアから，任意の述語項構造ペアの共起度を計算し，共起度の高い述語項構造ペアを関連の強い事態間知識として抽出する．任意の述語項構造ペアの組み合わせは膨大となるため，いかにして共起度の高い述語項構造ペアを見つけるかが問題となる．この問題を解決するために，述語項構造の共起度計算にアソシエーション分析[15] を適用する．

5.1 アソシエーション分析

アソシエーション分析は大量のデータから有用なルールを発見する手法である[15]．この手法はトランザクションデータから例えば「おむつを買う客はビールも買う傾向にある」というルールを発見するために提案されたものである．

アイテム $I = I_1, I_2, \cdots, I_m$ をバイナリの属性，トランザクション t をアイテムの集合からなると定義する ($t \subseteq I$)．また，トランザクションデータベース T を トランザクションの集合と定義する ($T = t_1, t_2, \cdots, t_n$)．

ルールを $X \Rightarrow Y$ ($X, Y \subseteq I$, $X \cap Y = \phi$) という形で定義し，これは「X が生じれば Y も生じやすい」ことを意味する．ここで，X を antecedent (left-hand side, lhs)，Y を consequent (right-hand side, rhs) と呼ぶ．ルールそれぞれについて，以下の 3 つの尺度 support 値，confidence 値，lift 値を定義する．

$$support(X \Rightarrow Y) = \frac{C(X \cup Y)}{|T|} \tag{1}$$

$$confidence(X \Rightarrow Y) = \frac{C(X \cup Y)}{C(X)} = \frac{support(X \Rightarrow Y)}{support(X)} \tag{2}$$

$$lift(X \Rightarrow Y) = \frac{confidence(X \Rightarrow Y)}{support(Y)} \tag{3}$$

ここで，$C(X)$ は X を含むトランザクションの数を表す．

support 値は X，Y が同時に出現する確率である．confidence 値は X が出現した際に Y が出現する条件付き確率である．lift 値は上記のように定義され，X と Y の相互情報量と等しくなる．

表 3 トランザクションデータの例 (一行が一トランザクションを表す)

PA_1		PA_2	
項	述語	項	述語
財布-ヲ	拾う	警察-ニ	届ける
彼-ガ, 財布-ヲ	拾う	警察-ニ	届ける
財布-ヲ	拾う		届ける
	拾う	警察-ニ	届ける
	...		
財布-ヲ	拾う		手渡す
財布-ヲ	拾う	彼-ニ	手渡す
男-ガ, 財布-ヲ	拾う		手渡す
	...		

Apriori アルゴリズム[17] はアソシエーション分析の実装のうちの一つである．このアルゴリズムは，アイテム群 abc の同時出現回数を t_1 回，アイテム群 abcd の同時出現回数を t_2 とすると必ず $t_1 \geq t_2$ となる性質を利用し，指定した条件を満たすルールを高速に見つける．Apriori アルゴリズムへの入力は，トランザクションデータ，support 値の最小値，confidence 値の最小値である．

5.2 Apriori アルゴリズムの述語項構造の共起度計算への適用

Apriori アルゴリズムを述語項構造の共起度計算に適用し，共起度の高い述語項構造ペアを得る．前節で定義したアイテムは述語または項に対応し，トランザクションは 4 節で抽出した係り受け関係にある述語項構造ペアに対応する．トランザクションデータの例を表 3 に示す．

本研究で抽出したいルールは以下の条件を満たすものである．

- X は PA_1 の述語と，PA_1 中の 0 個以上の項からなる
- Y は PA_2 の述語と，PA_2 中の 0 個以上の項からなる

したがって，上記の条件を満たさないルールは棄却する．残ったルールのうち，lift 値が *lift-min* 以上 *lift-max* 以下のものを採用する．*lift-max* 以上のルールは捨てるのは，相互情報量は頻度の低いものに対して過度に高い値をとるからである．

Apriori アルゴリズムによって，適切な述語項構造の単位が決定され，結果としてどの項が必須であるかを判断することができる．例えば，表 3 に示したトランザクションデータからは以下のルールが獲得される．

(1) 財布-ヲ 拾う ⇒ 警察-ニ 届ける
(2) 財布-ヲ 拾う ⇒ 手渡す

表 4 自動構築された格フレームの例 (用例の後の数字は頻度を表す)

用言	格	用例
拾う:1	ガ	女性 (2), 人 (2), ···
	ヲ	タクシー (3513), 車 (80), ···
	···	
拾う:10	ガ	男 (4), 女の子 (2), ···
	ヲ	財布 (580), 電話 (136), ···
	···	
届ける:1	ガ	スタッフ (164), 職員 (144), ···
	ヲ	情報 (103400), ニュース (4797), ···
	···	
届ける:20	ガ	男 (11), 人 (8), ···
	ヲ	財布 (8), 金 (6), ···
	ニ	警察 (2587), ···
	···	

最初のルールは,「拾う」と「届ける」の述語ペアに対して PA_1 の項「財布- ヲ」と PA_2 の項「警察-ニ」が必須であり，同様に，二つ目のルールは「拾う」と「手渡す」の述語ペアに対して PA_1 の項「財布-ヲ」が必須であることを意味する．

6. 格フレームに基づく項のアライメント

抽出した述語項構造ペアにおいて格要素がしばしば省略されるため，5 節で獲得されたルールにおいて格要素が欠如することが多い．例えば，以下のルールでは，PA_1 のヲ格は PA_2 でもヲ格であるが欠如しており，また,「男」「人」などが PA_1，PA_2 ともにガ格であるが欠如している．

- 財布-ヲ 拾う ⇒ 警察-ニ 届ける

獲得されたルールで欠如している項のアライメントを格フレームを用いて行なう．本研究では Web から自動獲得した格フレーム[1] を用いる．自動構築された格フレームの例を表 4 に示す．

PA_1 に対応付けられた格フレーム cf_1 のある項と，PA_2 に対応付けられた格フレーム cf_2 のある項が同じ ような格要素の分布を持つ時に，それらの項のアライメントをとる．

PA_1 と PA_2 での格フレームの選択ならびに格の対応付けの最善なものを以下のように決定する．

(1) ルールに項がある場合，それに基づき候補となる格フレームを絞り込み，そうでなければ全ての格フレームを候補とする．上記の例において，PA_1 では,「財布」をヲ格にとる格フレームに絞り込み，また，PA_2 では,「警察」をニ格にとる格フレームに絞り込む．5 節の最後にあげたルール (2) の PA_2 の場合，項がなく，述語「手渡す」だけのため,「手渡す」の格フレーム全てが候補となる．

(2) 以下のスコアを最大とする格フレームペアを選択し，その時の項アライメントを採用する．

$$\underset{cf_1,cf_2}{\operatorname{argmax}} \max_{\mathbf{a}} \sum_{a\in\mathbf{a}} sim(arg_1, a(arg_1)) \tag{4}$$

ここで，$\mathbf{a}$ は PA_1 と PA_2 の間の格のアライメント，arg_1 は PA_1 のうちのある格，$a(arg_1)$ は arg_1 とアライメントされた PA_2 の格，a は $arg1$ と $a(arg_1)$ のアライメント，sim は arg_1 と $a(arg_1)$ の格要素の分布の cosine 類似度を示す．例えば，格フレーム「拾う:10」のガ格と格フレーム「届ける:20」のガ格の sim は以下の 2 つのベクトルの cosine 類似度をとる．

	男	人	女の子	···
「拾う:10」のガ格	(4,	2,	2,	···)
「届ける:20」のガ格	(11,	8,	0,	···)

PA_1，PA_2 の格フレーム候補について上記のスコアを計算すると，この例では PA_1 に対応付けられた格フレーム 10 番と，PA_2 に対応付けられた格フレーム 20 番が選択され，その時のアライメントであるガ格とガ格，ヲ格とヲ格が対応付けられる．その際，PA_1，PA_2 の両方で格要素となっている名詞を用例として獲得する．上記の「拾う」のガ格と「届ける」のガ格の場合，用例として「男」,「人」, ··· が獲得される．

7. 実　　験

7.1 実験設定

日本語約 1 億ページからなるコーパスを利用して実験を行なった．これは約 60 億文からなる．ウェブにはミラーページなどの重複ページが多数存在することから，約 60 億文から重複を除いた約 16 億文を実験に利用した．

表 5 抽出されたルールと項アライメントの精度

抽出されたルール	○		×
	96(96.0%)		4(4.0%)
項アライメント	○	×	
	76(79.1%)	20(20.8%)	–

表 6 アソシエーション分析により獲得されたルールの例 (5 節)

	PA_1			PA_2		評価
	項	述語		項	述語	
(1)	定員-ニ	達する	⇒		締め切る	○
(2)	大学-ヲ	卒業する	⇒	会社-ニ	就職する	○
(3)		転倒する	⇒		骨折する	○
(4)		ノミネートされる	⇒		受賞する	○
(5)		訪ねる	⇒	話 ヲ	伺う	○
(6)		プレゼントする	⇒		喜ばれる	○
(7)		結婚する	⇒	子供-ガ	いる	○
(8)		利用-ニ あたる	⇒		登録-ガ 必要だ	×

まず，形態素解析器 JUMAN[⋆1]で形態素解析を行ない，構文解析器 KNP[⋆2]で構文解析を行なった．そして，構文解析結果から述語項構造ペアを抽出した．抽出された述語項構造ペアの数は約 4 億であった．

5.2 節で述べた Apriori アルゴリズムの適用において，support 値の最小値を 1.0×10^{-7}，confidence 値の最小値を 1.0×10^{-3} とし，また，*lift-min*, *lift-max* をそれぞれ 10，10,000 とした．

格フレームは上記の 16 億文から河原らの手法[1)] で自動構築した．約 30,000 用言において格フレームが構築され，1 用言あたりの平均格フレーム数は 25，1 格フレームあたりの格スロットの平均数は 4.7 であった．

7.2 実験結果と考察

7.2.1 抽出されたルールの評価

5 節で述べたアソシエーション分析によって約 2 万ルールが得られ，その中からランダムに 100 ルールを選び，それらが妥当であるかどうかを評価した．

⋆1 http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman.html
⋆2 http://nlp.kuee.kyoto-u.ac.jp/nl-resource/knp.html

表 7 獲得された事態ペアの例 (表の左の数字は表 6 の数字と対応する．また，下線をひいた項はルール獲得の段階で得られた項を示す．)

	PA_1			PA_2		評価
	項	述語		項	述語	
(1)	A_1:{ 募集, 申し込み, ...} ガ A_2:{ 定員 } ニ	達する	⇒	A_1:{ 募集, 申し込み, ...} ヲ	締め切る	○
(2)	A_1:{ 私, 子供, 娘, ...} ガ A_2:{ 大学 } ヲ	卒業する	⇒	A_1:{ 私, 子供, 娘, ...} ガ A_3:{ 会社 } ニ	就職する	○
(3)	A_1:{ 息子, 子供, 娘, ...} ガ	転倒する	⇒	A_1:{ 息子, 子供, 娘, ...} ガ	骨折する	○
(4)	A_1:{ 作品, ...} ガ A_2:{ 賞, 優秀賞, ...} ニ	ノミネートされる	⇒	A_1:{ 作品, ...} ガ A_2:{ 賞, 優秀賞, ...} ヲ	受賞する	○
(5)	A_1:{ 私, 人, ...} ガ A_2:{ 先生, 社長, ...} ヲ	訪ねる	⇒	A_1:{ 私, 人, ...} ガ A_2:{ 先生, 社長, ...} ニ A_3:{ 話 } ヲ	伺う	○
(6)	A_1:{ 人, 女性, ...} ガ A_2:{ 商品, 花, ...} ヲ	プレゼントする	⇒	A_2:{ 商品, 花, ...} ガ A_1:{ 人, 女性, ...} ニ	喜ばれる	×
(7)	A_1:{ 子供 } ガ	結婚する	⇒	A_1:{ 子供 } ガ	いる	×

表 5 の上部に精度を示す．精度は 96%であり，高い精度で関連の強い事態ペアを得ることができた．抽出されたルールとその評価を表 6 に示す．誤り原因としては複合辞の解析誤り (表 6 の (8)) や構文解析誤りがある．

7.2.2 項アライメントの評価

前節で正しいと評価された 96 ルールについて項アライメントの評価を行なった．表 5 の下部に精度を示す．精度は 79.1%であった．表 7 に獲得された項アライメントを含めた事態ペアの例を示す．

誤り例としては格フレームの複数の格の格要素の分布が非常に似ている場合に誤って対応をとるものがある．例えば，表 7 の (6) は誤って PA_1 の「A_1 ガ」と PA_2 の「A_1 ニ」が対応付いているが，正しい項のアライメントは以下のようになり，A_1 と A_3 は違う〈人〉である．

$$\begin{array}{l} A_1\text{:\{ 私, 人, ...\} ガ} \\ A_2\text{:\{ 商品, 花, ...\} ヲ プレゼントする} \\ A_3\text{:\{ 彼女, 親, ...\} ニ} \end{array} \Rightarrow \begin{array}{l} A_2\text{:\{ 商品, 花, ...\} ガ} \\ A_3\text{:\{ 彼女, 親, ...\} ニ} \end{array} \text{喜ばれる}$$

この問題に対処するためには,「喜ばれる」を能動態にし,「プレゼントする ⇒ 喜ぶ」での項のアライメントなどを総合的に考えることにより対処する予定である.

また，格フレームの格要素に用例があまり集まっておらず対応が誤る場合がある．例えば，表 7 の (7) の例では，正しい項のアライメントは以下のようになるが,「子供ガ いる」と対応付いた格フレームのニ格に〈人〉を表す用例があまり集まっていないため，PA_2 のニ格と PA_1 のガ格との対応がとれず，誤って，PA_1 のガ格と PA_2 のガ格を対応付けてしまっている．

$$\begin{array}{l} A_2\text{:\{ 私, 人, 女性, ...\} ガ 結婚する} \\ \underline{A_1\text{:\{ 子供 \} ガ}} \end{array} \Rightarrow \begin{array}{l} A_2\text{:\{ 私, 人, 女性, ...\} ニ} \\ \text{いる} \end{array}$$

この問題に対しては格フレームを構築するコーパスサイズを大きくすることが考えられる．

7.2.3 アンカーベースの手法との比較

提案手法をアンカーベースの手法[4] と比較した．共参照解析の精度がそれほど高くないことから (笹野らは新聞ドメインにおいて F 値で 0.75 と報告している[18])，ある Web ページで名詞が 2 度出現し，述語 w と述語 v に対して構文的関係を持てば，アンカーとみなすという単純な手法をとった．$e(w,d)$，$e(v,g)$ をそれぞれ述語 w と項 d の係り受け関係，述語 v と項 g の係り受け関係とし，項 d と項 g が共参照関係にある場合に，$e(w,d)$ と $e(v,g)$ の相互情報量は以下のように計算される．

$$pmi(e(w,d),e(v,g)) = \log \frac{P(e(w,d),e(v,g))}{P(e(w,d))P(e(v,g))} \quad (5)$$

提案手法で獲得されたルールにおいて，アライメントがとれた項における頻度上位 k 個の名詞を対象に，それらがアンカーベースの手法で獲得されるかどうかを調べた (k は 5 に設定した)．結果を表 8 に示す．カバー率は PA_1 と PA_2 の格に応じて分類している．表より，提案手法で獲得された名詞はアンカーベースの手法ではあまり獲得されないことがわかり，特に PA_1，PA_2 ともにガ格であるもののカバー率は相対的に低い．これは通常はエージェントに相当し，しばしば省略されることから，アンカーベースの手法では獲得されにくく，一方，提案手法では格フレームを用いたアライメントによって獲得することができている．

7.2.4 事態間ネットワーク

提案手法によって獲得された事態ペアを連結することによって，事態間ネットワークを構築することができる．図 2 に「入院」に関連する事態間ネットワーク，図 3 に「開発」に関連する事態間ネットワークを示す.「入院」や「開発」の前後にどのような事態が生じるかが

表 8 提案手法とアンカーベースの手法の比較 (カバー率は提案手法で獲得された項の対応付けがアンカーベースの手法でどれくらい獲得されたかを表す)

PA_1 の格	PA_2 の格	カバー率	
ガ	ガ	0.163	(3,768 / 23,180)
ガ	ヲ	0.282	(549 / 1,944)
ガ	ニ	0.176	(474 / 2,689)
ヲ	ガ	0.272	(753 / 2,764)
ヲ	ヲ	0.483	(7,106 / 14,713)
ヲ	ニ	0.321	(1,054 / 3,284)
ニ	ガ	0.163	(344 / 2,113)
ニ	ヲ	0.338	(1,042 / 3,086)
ニ	ニ	0.282	(549 / 1,944)

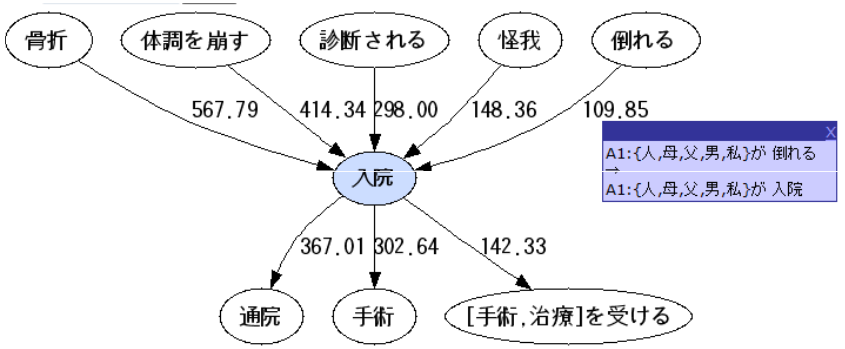

図 2 「入院」に関する事態間ネットワーク (「倒れる ⇒ 入院」の項の対応付けを表示している．図中の数字は lift 値を示す．)

獲得されていることがわかる．また，アンカーに基づく Chamber らの手法では図 2 中における「体調を 崩す」の「体調を」のような 1 つのノードにしか現れない項は獲得することができず，本研究ではこのような述語「崩す」の意味を特定するような項も獲得することができている．

8. おわりに

本論文では，述語項構造の共起情報と格フレームを用いて，大規模コーパスから事態間知識を自動獲得する手法について述べた．述語項構造の共起情報はアソシエーション分析を用いて効率的に計算し，項のアライメントは格フレームを用いて行なった．

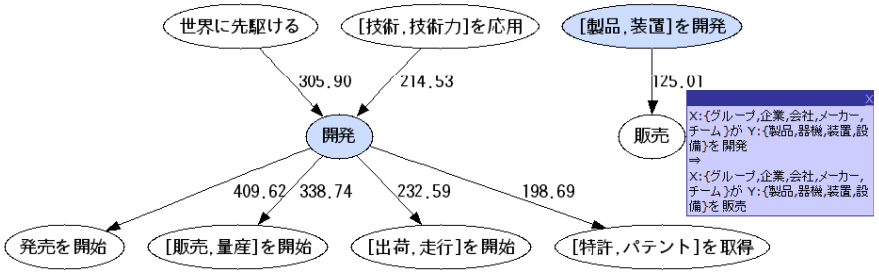

図 3 「開発」に関する事態間ネットワーク (「[製品, 装置] を開発 ⇒ 販売」の項の対応付けを表示している．)

今後の課題としては，時間経過，因果関係，手段などの事態間関係に分類することや，獲得された事態間知識を省略解析などの基礎解析や RTE(Recognizing Textual Entailment) や質問応答などのアプリケーションで利用し有用性を実証することなどがあげられる．

参 考 文 献

1) Kawahara, D. and Kurohashi, S.: A Fully-Lexicalized Probabilistic Model for Japanese Syntactic and Case Structure Analysis, *Proceedings of the HLT-NAACL2006*, pp.176–183 (2006).
2) Bean, D. and Riloff, E.: Unsupervised Learning of Contextual Role Knowledge for Coreference Resolution, *HLT-NAACL 2004: Main Proceedings*, pp.297–304 (2004).
3) Gerber, M. and Chai, J.: Beyond NomBank: A Study of Implicit Arguments for Nominal Predicates, *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp.1583–1592 (2010).
4) Chambers, N. and Jurafsky, D.: Unsupervised Learning of Narrative Event Chains, *Proceedings of ACL-08: HLT*, pp.789–797 (2008).
5) Chambers, N. and Jurafsky, D.: Unsupervised Learning of Narrative Schemas and their Participants, *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp.602–610 (2009).
6) Miller, G. A.: Wordnet: A lexical detabase for English, *Communications of the ACM* (1995).
7) Singh, P. and Williams, W.: LifeNet: A Propositional Model of Ordinary Human Activity, *Proceedings of Workshop on Distributed and Collaborative Knowledge Capture* (2003).
8) Espinosa, J. and Lieberman, H.: EventNet: Inferring Temporal Relations Between Commonsense Events, *Proceedings of the 4th Mexican International Conference on Artificial Intelligence*, pp.61–69 (2005).
9) Regneri, M., Koller, A. and Pinkal, M.: Learning Script Knowledge with Web Experiments, *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp.979–988 (2010).
10) Lin, D. and Pantel, P.: Discovery of Inference Rules for Question Answering, *Natural Language Engineering*, Vol.7, No.4, pp.343–360 (2001).
11) Szpektor, I. and Dagan, I.: Learning Entailment Rules for Unary Templates, *Proceedings of the 22nd International Conference on Computational Linguistics (COLING)*, pp.849–856 (2008).
12) Fujiki, T., Nanba, H. and Okumura, M.: Automatic Acquisition of Script Knowledge from a Text Collection, *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2003)*, pp.91–94 (2003).
13) Torisawa, K.: Acquiring Inference Rules with Temporal Constraints by using Japanese Coordinated Sentences and Noun-Verb Co-occurrences, *Proceedings of Human Language Technology Conference/North American chapter of the Association for Computational Linguistics annual meeting (HLT-NAACL06)*, pp.57–64 (2006).
14) Abe, S., Inui, K. and Matsumoto, Y.: Two-phased event relation acquisition: coupling the relation-oriented and argument-oriented approaches, *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pp.1–8 (2008).
15) Agrawal, R., Imielinski, T. and Swami, A.: Mining association rules between sets of items in large databases, *Proceedings of the ACM-SIGMOD 1993 International Conference on Management of Data (1993)*, pp.207–216 (1993).
16) Kazama, J. and Torisawa, K.: Inducing Gazetteers for Named Entity Recognition by Large-Scale Clustering of Dependency Relations, *Proceedings of ACL-08: HLT*, pp.407–415 (2008).
17) Borgelt, C. and Kruse, R.: Induction of Association Rules: Apriori Implementation, *Proceedings of 15th Conference on Computational Statistics*, pp.395–400 (2002).
18) Sasano, R., Kawahara, D. and Kurohashi, S.: Improving Coreference Resolution Using Bridging Reference Resolution and Automatically Acquired Synonyms, *Discourse Anaphora and Anaphor Resolution Colloquium*, pp.125–136 (2007).