評判分析のための評価視点木構築における汎用シソーラス利用法の検討

山下 和輝　乾 孝司　山本 幹雄
筑波大学大学院 システム情報工学研究科
yamashita@mibel.cs.tsukuba.ac.jp {inui,myama}@cs.tsukuba.ac.jp

1 はじめに

詳細な評判分析を実施する場合，入力文書から評価視点（属性とも呼ばれる）を抽出し，評価視点を単位として肯定／否定の評価極性を推定することがある．自動抽出される評価視点は一般に異なり数が多いため，これまでに評価視点を既定カテゴリへ分類するなど，評価始点の集合を構造化する研究が進められている（例えば，文献 [3]）．

このような背景のもと，我々も評価視点集合を構造化する研究に取り組んでいる．以前から評価視点は階層性を有することが指摘されている [6]．そこで我々は，評価視点集合を構造化するあたり，図 1 に示すような評価視点（の部分）をノードとする木構造を想定し，これを自動構築する手法を検討してきた [5]．以降，この木を評価視点木と呼び，文献 [5] の手法を基本法と呼ぶ．

本稿では，基本法がもつ問題点について，汎用シソーラスから獲得できる関係知識を利用することで解消する方法について議論する．以降，まず 2 節で基本法について説明する．3 節で基本法の問題点を述べ，それを解消する手法について論じる．その後，4 節で提案手法の有効性を評価実験を通じて検証する．

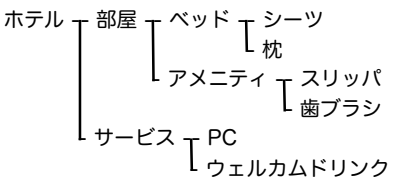

図 1: 宿泊サービスにおける評価視点木（部分）の例

2 基本法の概要

基本法では，ある評価対象に関するレビュー文書集合を入力とし，そこから階層関係が成立する評価視点対（図 1 の例で言えば,「部屋-ベッド」など）を自動抽出する．そして，評価視点をノード，評価視点対をエッジとみなすことでグラフ構造を得る．この段階で得られるグラフは，何らかの構造的制約を課されておらず，また，評価視点対の誤抽出の影響も入り込み，木構造となることは稀である．そこで，グラフから何らかの基準に従って最良な評価視点木を求める．文献 [5] ではグラフから評価視点木を求めるアルゴリズムを幾つか提案しているが，本稿では以降の議論に関連する 2 手法（貪欲法と MST 法）を説明する．

Algorithm 1 評価視点木構築：貪欲法

Input:
$G =< V, E, W >$
$V = \{v_i\}\ (1 \le i \le |V|)$
$E = \{e_{ij}\}\ (1 \le i, j \le |V|)$
$W = \{w_{ij}\}\ (1 \le i, j \le |V|)$ // $w(e_{ij})$ で参照
$M\ (\le |V|)$ // 最大ノード数
$v_r\ (\in V)$ // ルートノード

Output: $G_T =< V_T, E_T >$

```
1:  begin
2:    V_T = {v_r}, V = V − {v_r}, E_T = {}
3:    E_C = expand(G, v_r)
4:    while |V_T| < M
5:      e_kl = arg max_{e_ij ∈ E_C, v_i ∈ V_T, v_j ∉ V_T} w(e_ij)
6:      V_T = V_T ∪ {v_l}, V = V − {v_l}
7:      E_T = E_T ∪ {e_kl}, E_C = E_C − {e_kl}
8:      E_C = E_C ∪ expand(G, v_l)
9:    end
10: end
```

貪欲法のアルゴリズムを Algorithm 1 に示す．入力は重み付き有向グラフである．エッジの重みとして，文献 [5] ではコーパス中でのエッジ（評価視点対）の出現頻度を採用しており，本稿でもそれに従う．M は出力される木のサイズを指定するパラメータである．expand(G,v) は G のエッジ集合（E）のうち，v を始点とする部分エッジ集合を返す関数である．貪欲法では，指定された木のルートノード（v_r）から順に，木を形成する制約下において，重みが大きいエッジを局所的に選択することを繰り返すことで出力木を得る．

次に MST 法のアルゴリズムを Algorithm 2 に示す．基本的な手続きの流れは貪欲法と同様であるが，出力木の骨格となる形状を定めるべく，アルゴリズムの冒頭（2 行目）において最大全域木を求めている．これによって，不要なエッジを事前に排除している点が貪欲法と異なる．最大全域木の獲得には Prim 法 [4] を用いる．ただし，オリジナルの Prim 法は無向グラフを対象としたアルゴリズムである．そのため，まず入力グラフのエッジの向きを無視したグラフに対して Prim 法を適用することで最大全域木を求める．その後，全域木の各エッジに元の向き情報を復元する処理をおこ

Algorithm 2 評価視点木構築：MST 法

Input:
$G =< V, E, W >$
$V = \{v_i\}\ (1 \leq i \leq |V|)$
$E = \{e_{ij}\}\ (1 \leq i, j \leq |V|)$
$W = \{w_{ij}\}\ (1 \leq i, j \leq |V|)$
$M\ (\leq |V|)$, $v_r\ (\in V)$

Output: $G_T =< V_T, E_T >$

1: **begin**
2: G の最大全域木 $G_M =< V_M, E_M, W_M >$ を求める
3: $V_T = \{v_r\}$, $V_M = V_M - \{v_r\}$, $E_T = \{\}$
4: $E_C = \mathrm{expand}(G_M, v_r)$
5: **while** $|V_T| < M$
6: $e_{kl} = \underset{\substack{e_{ij} \in E_C \\ v_i \in V_T,\ v_j \notin V_T}}{\arg\max}\ w(e_{ij})$
7: $V_T = V_T \cup \{v_l\}$, $V_M = V_M - \{v_l\}$
8: $E_T = E_T \cup \{e_{kl}\}$, $E_C = E_C - \{e_{kl}\}$
9: $E_C = E_C \cup \mathrm{expand}(G_M, v_l)$
10: **end**
11: **end**

なっている．

3 提案手法

3.1 基本法の問題点と解決方針

基本法の出力誤りを分析したところ，入力グラフ G を作成する段階で，出力として望まれる評価視点木の構成要素となるエッジが一部欠落している，あるいはエッジの重みが小さくなることがわかった．例えば，出力として望む評価視点木に「食事-朝食」という構成要素があるとしよう．にも関わらず，「食事は，朝，昼，夕の時間帯ごとに下位分類でき，朝食はそのひとつである」というような常識的な知識をコーパスから抽出することが難しいため，基本法ではこのようなエッジを出力木の構成要素として含めることができていなかった．

この問題を解決するために，本研究では既存の汎用シソーラスを利用する．汎用シソーラスには上記のような常識的な知識が多く登録されていると仮定できる．そこで，汎用シソーラスから必要な評価視点間の関係知識を抽出し，その知識を考慮できるよう，Algorithm 1 および Algorithm 2 を拡張する．

3.2 拡張アルゴリズム

前処理として，シソーラスから「食事-朝食」のような上位- 下位関係と，「朝食-夕食」のような同位関係の 2 種類の 2 項関係の知識を抽出しておく．そして，Algorithm 1 あるいは Algorithm 2 内でエッジを伸ばす際に，関連する関係知識が存在していれば，優先的に出力木に取り込まれるようにアルゴリズムを拡張する．

シソーラスから抽出した関係集合を $R\ (\subseteq E)$ とする．R を考慮した貪欲法の拡張アルゴリズムを Algorithm 3 に示す．このアルゴリズムは上位-下位関係を考慮する

Algorithm 3 評価視点木構築：貪欲法 + シソーラス

Input:
$G =< V, E, W >$
$V = \{v_i\}\ (1 \leq i \leq |V|)$
$E = \{e_{ij}\}\ (1 \leq i, j \leq |V|)$
$W = \{w_{ij}\}\ (1 \leq i, j \leq |V|)$
$M\ (\leq |V|)$, $v_r\ (\in V)$
R // シソーラスから抽出した関係知識集合

Output: $G_T =< V_T, E_T >$

1: **begin**
2: $V_T = \{v_r\}$, $V = V - \{v_r\}$, $E_T = \{\}$
3: $E_C = \mathrm{expand}(G, v_r)$
4: **while** $|V_T| < M$
5: $e_{kl} = \underset{\substack{e_{ij} \in E_C \\ v_i \in V_T,\ v_j \notin V_T}}{\arg\max}\ \left(w(e_{ij}) + \sum_{e_{jm} \in \mathrm{isa}_e(R, v_j)} w(e_{jm})\right)$
6: $V_T = V_T \cup \{v_l\} \cup \mathrm{isa}_v(R, v_l)$
7: $V = V - \{v_l\} - \mathrm{isa}_v(R, v_l)$
8: $E_T = E_T \cup \{e_{kl}\} \cup \mathrm{isa}_e(R, v_l)$
9: $E_C = E_C - \{e_{kl}\} - \mathrm{isa}_e(R, v_l)$
10: $E_R = \bigcup_{v_m \in \mathrm{isa}_v(R, v_l)} \mathrm{expand}(G, v_m)$
11: $E_C = E_C \cup \mathrm{expand}(G, v_l) \cup E_R$
12: **end**
13: **end**

版である．評価視点木にエッジを追加する箇所およびその後続処理（5 行目から 11 行目）において， isa() 関数によって上位-下位関係の関係知識が取り込まれる点が元アルゴリズムと異なる．ここで，$\mathrm{isa}_e(R, v)$ は，エッジ集合 R の部分集合を返す関数であり，ノード v を始点とし，かつ，評価視点木として未採用なエッジの集合を返す．また，$\mathrm{isa}_v(R, v)$ は上記の各部分エッジの終点の集合を返す関数である．

アルゴリズムの差異を具体的に述べると以下の通りである．元アルゴリズム（Algorithm 1）ではエッジを追加する際に常に追加エッジ（e_{kl}）を 1 本ずつ選択していく．一方，拡張アルゴリズム（Algorithm 3）では同じく追加エッジが選択された時，それと同時に isa() 関数が返す上位-下位関係が成立するエッジの束があわせて処理され，これによって R に含まれるエッジは評価視点木内で親子関係を形成するよう制約されながら追加される．

詳細は割愛するが，Algorithm 3 において isa() 関数が関連する箇所を同位関係（すなわち，共通の親ノードをもつエッジ集合）を返す関数に置き換えることで，同位関係が考慮できる同様なアルゴリズムが実現できる．また，両方の関数を同時に適用することで，両方の関係知識を同時に考慮することができる．

次に，関係知識 R を考慮した MST 法の拡張アルゴリズムについて述べる．拡張の基本的な考え方は Algorithm 3 と同じである．すなわち，木の構築においてエッジを追加する際，関係知識から得られる追加エッジと関連する（ノードを共有する）エッジの束を

Algorithm 4
制約付き最大全域木獲得：Prim 法 + シソーラス

Input:
$G =< V, E, W >$
$V = \{v_i\}\ (1 \le i \le |V|)$
$E = \{e_{ij}\}\ (1 \le i, j \le |V|)$
$W = \{w_{ij}\}\ (1 \le i, j \le |V|)$
R // シソーラスから抽出した関係知識集合

Output: $G_M =< V_M, E_M >$

begin
  $V_M = \{v_r\}$, $V = V - \{v_r\}$, $E_M = \{\}$
  $E_C = \mathrm{expand}(G, v_r)$
  while $|V| > 0$
    $e_{kl} = \underset{\substack{e_{ij} \in E_C \\ v_i \in V_M,\ v_j \notin V_M}}{\arg\max} \left(w(e_{ij}) + \sum_{e_{jm} \in \mathrm{isa}_e(R, v_j)} w(e_{jm}) \right)$
    $V_M = V_M \cup \{v_l\} \cup \mathrm{isa}_v(R, v_l)$
    $V = V - \{v_l\} - \mathrm{isa}_v(R, v_l)$
    $E_M = E_M \cup \{e_{kl}\} \cup \mathrm{isa}_e(R, v_l)$
    $E_C = E_C - \{e_{kl}\} - \mathrm{isa}_e(R, v_l)$
    $E_R = \bigcup_{v_m \in \mathrm{isa}_v(R, v_l)} \mathrm{expand}(G, v_m)$
    $E_C = E_C \cup \mathrm{expand}(G, v_l) \cup E_R$
  end
end

あわせて処理すれば良い．ただし，MST 法では手続きの冒頭で入力グラフを最大全域木に変換する．そのため，関係知識は最大全域木を獲得する過程で取り込む．上記をまとめると，関係知識 R を考慮した MST 法の拡張アルゴリズムは，手続きとしては Algorithm 2 と同一のものとなる．ただし，Algorithm 2 では最大全域木の獲得に Prim 法を用いていたが，ここを関係知識を考慮できるよう，次の Algorithm 4 に置き換えることとする．Algorithm 4 内で使われている記号の意味は他のアルゴリズムと同様であるが，元の Prim 法が無向グラフを対象としていることから，入力となるエッジ集合 E および R は，元のエッジから向きを無視した無向エッジである．また，isa() 関数も始点と終点を区別しないものとする．最大全域木 G_M が獲得された後，全域木の各エッジへは元の向き情報（と重み情報）が復元されるとする．

貪欲法の場合と同様の議論で，isa() 関数の箇所を置き換えることで，同位関係を考慮する版も実現できる．

4 評価実験

4.1 実験設定

実験には，楽天トラベルの宿泊レビューデータ（約 400 万件）[1] を用いた．まず，山下らの手法 [5] によって，このデータから「朝食-パン」のような階層関係にある評価視点対を自動抽出した．その後，著者のうち２名で相談を持ちながら，抽出結果から手作業で正解となる評価視点木を作成した．具体的には，「ホテル」を正解木のルートノードとし，評価視点対をエッジとみなして「ホテル」から順次エッジを伸ばすことで正解木を作成した．作成した正解木はノードが 241 個で，木の深さの平均が 3.15 である．

次に，正解木に含まれるノードに対して上記の自動抽出された評価視点対の情報に基づいてエッジを生成することで入力グラフを作成した．入力グラフはノードが 241 個で，エッジ数が 4,168 本である．入力グラフのエッジは自動処理で獲得されるため，正解木を完全には被覆していない．正解木における入力グラフのエッジ単位の被覆率は 77.5% であった．

汎用シソーラスには日本語 WordNet（日本語 WN）[1] を利用した．日本語 WN を参照し，入力グラフに含まれる評価視点間に上位-下位関係（is-a 関係）が成立するか否かの情報を関係知識として抽出し，木の構築実験に用いた．また，日本語 WN において同じ上位概念を共有する下位ノード間には同位関係が成立するとし，その情報も実験で用いる．抽出された関係の例を以下に示す．

- is-a：食事-朝食，魚-マグロ
- 同位：朝食-夕食，石鹸-シャンプー

評価指標には，パス適合率（Path P）とパス再現率（Path R）を用いる．また，正解木内のノードの上位-下位関係を出力木がどの程度保っているかを示す指標として上位-下位再現率（is-a R）も計測した．ここで，出力木の全パス集合を P_{out}，正解木の全パス集合を P_{gold} としたとき，

- Path P = $|P_{out} \cap P_{gold}| / |P_{out}|$
- Path R = $|P_{out} \cap P_{gold}| / |P_{gold}|$

である．また，P_{out} の要素である各パスに対して両端点ノードの対を抽出し，それを要素とする集合を T_{out}，P_{gold} の要素に対して同様の処理を適用して得られる集合を E_{gold} としたとき，

- is-a R = $|T_{out} \cap T_{gold}| / |T_{gold}|$

である．

4.2 実験結果

実験結果を表 1 および表 2 に示す．表 1 は，各手法で出力された評価視点木の構造に関する要約情報として，ルートノードから葉ノードまでのパスの最大値と平均値をそれぞれ示している．表 2 は各手法における評価指標の評価値である．表中の（+ is-a）は，上位-下位関係の関係知識を考慮した場合，（+ is-a，同位）は，上位-下位関係と同位関係の両方を考慮した場合の結果である[2]．各表において，上２行が関係知識を考慮しないベースライン，下４行が関係知識を考慮する提案手法の結果である．

まず，表 2 の性能比較から，上位-下位関係の知識を考慮した MST 法で性能の向上が確認できる．特に，上位-下位再現率の改善が著しい．性能変化の詳細を把握するために定性分析を実施したところ，以下のことが

[1] http://rit.rakuten.co.jp/opendataj.html

[2] 実験条件として（+ 同位）という設定の実験も実施したが，結果の議論が（+ is-a，同位）と同様になるので割愛する．

表 1: 実験結果（出力木の深さ）

	最大値	平均値
貪欲法	7	4.3
MST 法	6	3.8
貪欲法（+ is-a）	6	4.0
MST 法（+ is-a）	4	3.5
貪欲法（+ is-a，同位）	6	3.7
MST 法（+ is-a，同位）	4	2.5

表 2: 実験結果（性能評価）

	Path P	Path R	is-a R
貪欲法	0.28	0.26	0.33
MST 法	0.27	0.27	0.29
貪欲法（+ is-a）	0.28	0.26	0.33
MST 法（+ is-a）	**0.31**	**0.30**	**0.66**
貪欲法（+ is-a，同位）	0.25	0.23	0.29
MST 法（+ is-a，同位）	0.18	0.17	0.11

わかった．図 2 の点線で囲われた例のように評価視点木の葉ノードに近く具体性の高いノードをもつ部分木はどの手法でもある程度良好な出力を得ていた．しかし，これら部分木が適切な親ノードへ正しく接続されるか否かが手法によって大きく異なっていた．図 2 の「食事-朝食」関係のような抽象性がやや高いノード間の関係はレビュー内で記述される事が稀である．そのため，このようなエッジの重みは入力グラフにおいて極端に低い．その結果として，関係知識を用いない場合は「食事-朝食」エッジを評価視点木に含めることが難しい．一方で，上位-下位関係の知識を考慮する場合，シソーラスから抽出した「食事-朝食」関係を使うことで上記の状態において適切なエッジ選択がなされる．

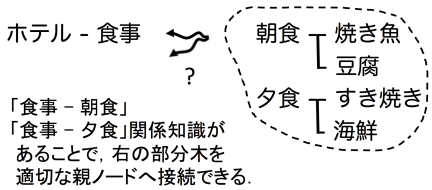

図 2: 抽象性の高い親ノードへの接続問題

上位-下位関係の知識を考慮した貪欲法も上記のような状態において出力の改善がなされるが，次の理由から，関係知識が適切に考慮される回数が少なくなったと考えられる．すなわち，関係知識が効果を発揮するためには，その条件として，Algorithm 3 あるいは Algorithm 4 の 5 行目において関係知識が反映されるエッジ（e_{kl}）が適切に選択される必要がある．しかしながら，貪欲法は，局所的にエッジ重みが評価されてしまうため，関係知識が反映されるエッジ（e_{kl}）を構成するノード（v_l）が，知識が反映されるよりも早く出力木に取り込まれてしまい，関係知識が効果を発揮する条件を満たしづらいようである．

同位関係の知識を考慮した場合は，どちらの手法でも芳しい結果を得られていない．この原因は，汎用シソーラスがもつ知識構造の違いとして説明できると考えている．あるノードに対して成立する同位関係は上位-下位関係よりも一般に多い[3]．これにより同位関係を考慮すると階層の浅い出力木が得られることは表 1 からも確認できる．今回の結果では，同位関係の関係知識が誤った位置で反映された際の副作用がその効果を上回っており，性能の劣化を招いていた．現在のところ，同位関係の扱いは上位-下位関係よりも難しく，さらなる検討が望まれる．

[3] つまり，知識の階層構造を仮定した場合，階層数よりも同階層での兄弟数の方が多い．

5 おわりに

本稿では，評価視点集合から評価視点木を自動構築する際に，既存の汎用シソーラスを利用する方法について論じた．評価実験の結果，シソーラスから得られる関係知識を利用することによって，特に，抽象性の高いノードが関連する部分木を評価視点木の適切な位置へ配置できるようになった．

関連研究として，SemEval[2] では，構造として厳密に木とする制限はないが，階層的知識を自動構築する Taxonomy Extraction の研究が進められおり，特に，概念対の自動抽出に対する知見が蓄積されている．今後，これらの知見も評価視点木の自動構築に取り込んでいきたい．また，オントロジーの観点から見れば，本研究は汎用オントロジーを使って，ドメイン特化オントロジーを自動補完する方法の検討とみなせる．今後，オントロジー研究との関連についても精査していきたい．

謝辞

評価実験にあたり，楽天データ公開において公開された楽天トラベル施設レビューデータを使用させて頂きました．関係者に深く感謝いたします．

参考文献

[1] Francis Bond, Hitoshi Isahara, Sanae Fujita, Kiyotaka Uchimoto, Takayuki Kuribayashi, and Kyoko Kanzaki. Enhancing the japanese wordnet. In *Proceedings of the 7th Workshop on Asian Language Resources*, pp. 1–8, 2009.

[2] Georgeta Bordea, Paul Buitelaar, Stefano Faralli, and Roberto Navigli. Semeval-2015 task 17: Taxonomy extraction evaluation (TExEval). In *Proceedings of the SemEval2015*, pp. 902–910, 2015.

[3] Giuseppe Carenini, Raymond T Ng, and Ed Zwart. Extracting knowledge from evaluative text. In *Proceedings of the 3rd international conference on Knowledge capture*, pp. 11–18, 2005.

[4] Robert Clay Prim. Shortest connection networks and some generalizations. *Bell system technical journal*, Vol. 36, No. 6, pp. 1389–1401, 1957.

[5] 山下和輝, 乾孝司, 山本幹雄. 表層的言語パタンを用いた階層的評価視点カタログの自動生成. 第 28 回人工知能学会全国大会, 2014.

[6] 小林のぞみ, 乾健太郎, 松本裕治. 意見情報の抽出/構造化のタスク仕様に関する考察. 情報処理学会自然言語処理研究会 (NL-171-18), 2006.