

大規模語彙知識を用いた 仮説推論による談話解析の 課題と対策

井之上 直也[†] 乾 健太郎[†]

Ekaterina Ovchinnikova[‡] Jerry R. Hobbs[‡]

[†]東北大学 [‡]USC/ISI

潜在する情報の顕在化

入力 (文章):

Steve will resign from Microsoft next week.

His products have attracted many people over the years.

work(Steve, Microsoft)?
create(Steve, products)?
sick(Steve)?
:

仮説選択

work → resign
create → of
sick → resign
:
:

論理ベースの推論

背景知識

出力 (文章に潜在する情報):

- Steve works for Microsoft
- His refers to Steve
- Steve creates products

潜在する情報の抽出

入力 (文章):

Steve will resign from
His products have attri-
butes. Steve has been work-
ing for Microsoft for 20 years.

推論の基盤技術のお話
(杉浦+ の推論の実現
のための基盤技術)

work(Steve, Microsoft)?
create(Steve, products)?
sick(Steve)?
:

仮説選択



論理ベースの推論

work → resign
create → of
sick → resign
:
:

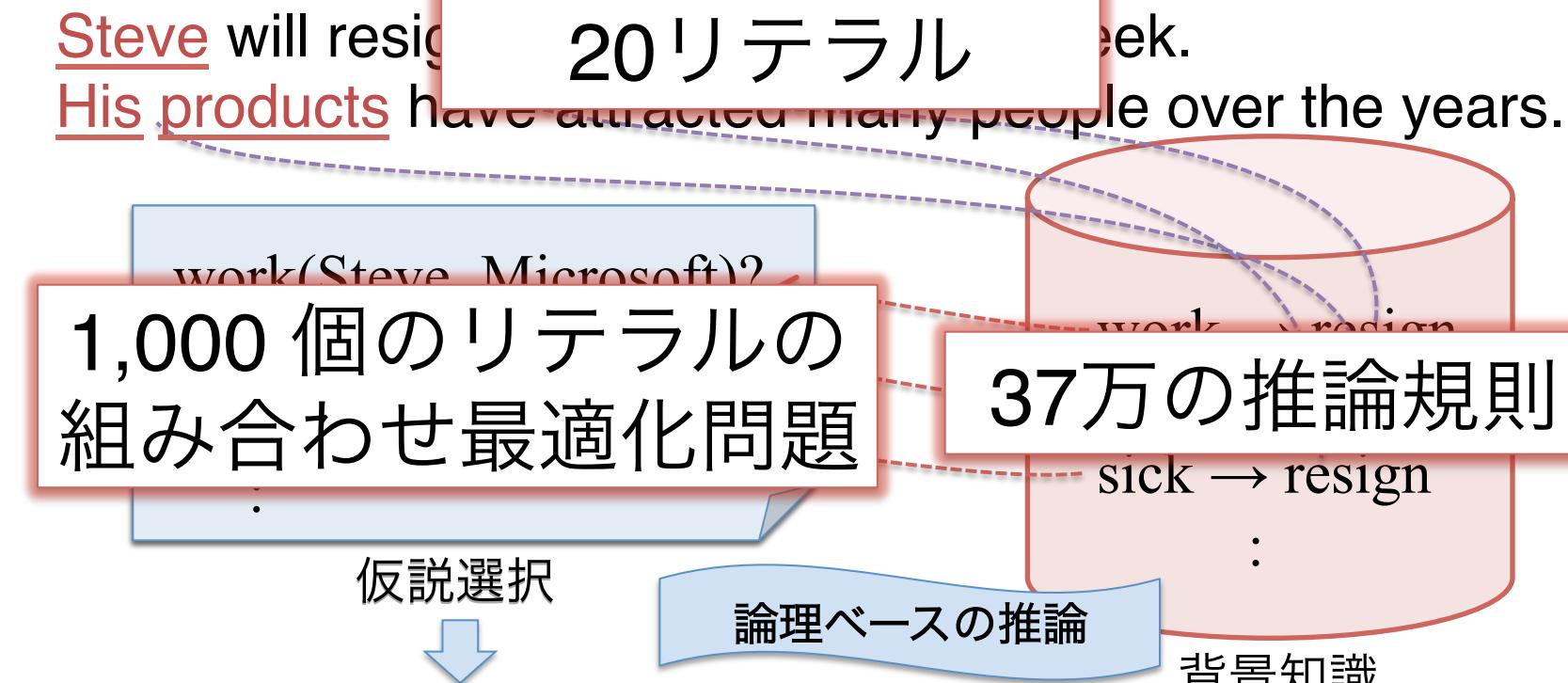
背景知識

出力 (文章に潜在する情報):

- Steve works for Microsoft
- His refers to Steve
- Steve creates products

本研究の成果: 推論効率の改善

入力 (文章):



出力

提案手法の推論時間: 約2.6秒

- Markov Logic Networks (Richardson & Domingos 05)による実装 (Blythe+ 11): 7分
- Mini-TACITUS (Mulkar-Mehta+ 07): 30分以上

- 仮説推論
- 仮説選択の高速化
- 仮説の無矛盾性制約の導入
- 評価実験

仮説推論

整数線形計画法 (ILP) による
定式化で高速化

1. **仮説候補生成:** 最良仮説の候補 H の集合 \mathcal{H} を生成する
 - 観測 O を含意せよ ($B \cup H \models O$)
 - 矛盾するな ($B \cup H \not\models \perp$)
2. **仮説選択:** 最良の仮説 H^* を選択する
 - 評価関数が最大となる仮説を選べ
$$(H^* = \arg \max_{H \in \mathcal{H}} score(H))$$

ヒューリスティクス論

1. 0-1 ILP 変数への
値割り当てにより表現

する仮説 H^* を,
する論理的推論

言語

1. **仮説候補生成**: 最良仮説の候補 H の集合

\mathcal{H} を生成する

- 観測 O を含意せよ ($B \cup H \models O$)
- 矛盾するな ($B \cup H \not\models \perp$)

3. ILP 変数間の
制約により表現

2. **仮説選択**: 最良の仮説 H^*

- 評価関数が最大となる仮説

$$(H^* = \arg \max_{H \in \mathcal{H}} score(H))$$

2. ILP 目的関数
により表現

入力

ILP 変数
ILP 目的関数
ILP 制約

背景知識 B :

$$\begin{cases} \text{sick}(x) \rightarrow \text{resign}(x, y) \\ \text{hate}(x, y) \rightarrow \text{resign}(x, y) \\ \text{old}(x) \rightarrow \text{resign}(x, y) \end{cases}$$

$$\begin{cases} \text{sick}(x) \rightarrow \text{go}(x, y) \wedge \text{hospital}(y) \\ \text{hospital}(\text{AbcHospital}). \\ \text{boring}(y) \rightarrow \text{hate}(x, y) \end{cases}$$

観測

$$O: \text{resign}(\text{Steve}, \text{Microsoft}) \wedge \text{go}(\text{Steve}, \text{AbcHospital})$$

ILP による定式化

$$\begin{array}{ccccccccc} & & \text{resign}(\text{Steve}, \text{Microsoft}) & & \text{go}(\text{Steve}, \text{AbcHospital}) & & & & \\ \text{sick}(\text{Steve}) & \rightarrow & \text{hate}(\text{Steve}, \text{Microsoft}) & \text{old}(\text{Steve}) & \text{sick}(\text{Steve}) & \text{hospital}(\text{AbcHospital}) & & & \\ & & & & \downarrow & & & & \\ & & & & \text{boring}(\text{Microsoft}) & \text{hospital}(\text{AbcHospital}) & & & \end{array}$$

$$score(H) =$$

出力

最良の仮説 H^* ($B \cup H^* \models O$):

入力

背景知識 B :

$$\text{sick}(x) \rightarrow \text{resign}(x, y)$$

$$\text{hate}(x, y) \rightarrow \text{resign}(x, y)$$

$$\text{old}(x) \rightarrow \text{resign}(x, y)$$

ILP 変数

ILP 目的関数

ILP 制約

$$\text{sick}(x) \rightarrow \text{go}(x, y) \wedge \text{hospital}(y)$$

$$\text{hospital}(\text{AbcHospital}).$$

$$\text{boring}(y) \rightarrow \text{hate}(x, y)$$

観測 O :

$$\text{resign}(\text{Steve}, \text{Microsoft}) \wedge \text{go}(\text{Steve}, \text{AbcHospital})$$

ILP による定式化

$$h_{\text{resign}(\text{Steve}, \text{Microsoft})} = 1, h_{\text{go}(\text{Steve}, \text{AbcHospital})} = 1$$
$$h_{\text{sick}(\text{Steve})} = 1, h_{\text{hate}(\text{Steve}, \text{Microsoft})} = 0, h_{\text{old}(\text{Steve})} = 0, h_{\text{sick}(\text{Steve})} = 0, h_{\text{hospital}(\text{AbcHospital})} = 0$$
$$h_{\text{boring}(\text{Microsoft})} = 0, h_{\text{hospital}(\text{AbcHospital})} = 0$$

$$score(H) =$$

出力

最良の仮説 $H^* (B \cup H^* \models O)$:

$$\text{sick}(\text{Steve}) \wedge \text{resign}(\text{Steve}, \text{Microsoft}) \wedge \text{go}(\text{Steve}, \text{AbcHospital})$$

入力

ILP 変数
ILP 目的関数
ILP 制約

背景知識 B :

$$\text{sick}(x) \rightarrow \text{resign}(x, v)$$

重み付き仮説推論では...

sick: 支払う

resign:

ILP による定義

$$h_{\text{sick}(\text{Steve})} = 1, h_{\text{hate}(\text{Steve}, \text{Microsoft})} = 0$$

$$\text{sick}(x) \rightarrow \text{go}(x, y) \wedge \text{hospital}(y)$$

ital(AbcHospital).

$$g(y) \rightarrow \text{hate}(x, y)$$

1. $r_p = 1$: コストを支払う
別のリテラルに説明されていないかつ
2. 単一化されていない

$$h_{\text{boring}(\text{Microsoft})} = 0, h_{\text{hospital}(\text{AbcHospital})} = 0$$

$$\text{score}(H) = \sum_{p \in \{p | p \in P, h_p = 1\}}$$

重み付き仮説推論
(Hobbs+ 93)

出力

最良の仮説 H^* ($B \cup H^* \models O$):

$$\text{sick}(\text{Steve}) \wedge \text{resign}(\text{Steve}, \text{Microsoft}) \wedge \text{go}(\text{Steve}, \text{AbcHospital})$$

入力

背景知識 B :

$$\text{sick}(x) \rightarrow \text{resign}(x, y)$$

$$\text{hate}(x, y) \rightarrow \text{resign}(x, y)$$

$$\text{old}(x) \rightarrow \text{resign}(x, y)$$

ILP
変数
ILP 目的
ILP 節

2. 仮説間の含意関係

e.g., $h_{\text{boring}(\text{Mi})} \leq h_{\text{hate}(\text{St}, \text{Mi})}$

$$\text{sick}(x) \rightarrow go(x, y) \wedge hospital(y)$$

$$hospital(\text{AbcHospital}).$$

$$\text{boring}(y) \rightarrow \text{hate}(x, y)$$

観測 O :

$$\text{resign}(\text{Steve}, \text{Microsoft}) \wedge go(\text{Steve}, \text{AbcHospital})$$

ILP による定式化

1. 仮説は常に観測を含意

$$\text{e.g., } h_{\text{resign}(\text{St}, \text{Mi})} = 1$$

四ノノ

最良の仮説 $H^* (B \cup H^* \models O)$:

$$\text{sick}(\text{Steve}) \wedge \text{resign}(\text{Steve}, \text{Microsoft}) \wedge go(\text{Steve}, \text{AbcHospital})$$

3. コスト支払い条件

e.g., $1 - h_{\text{boring}(\text{Mi})} \leq r_{\text{hate}(\text{St}, \text{Mi})}$

$cost(p)$

$, h_p = 1, r_p = 1 \}$

仮説推論

仮説選択の高速化

仮説の無矛盾性制約の導入

評価実験

仮説候補の矛盾問題

- 矛盾した仮説候補とは?
 - 同一の実体に対して、意味的互換性のない性質が同時に仮説されている仮説候補
 - e.g., $H = (\exists x) \text{cat}(x) \wedge \text{dog}(x)$
- 単一化による矛盾の発生が特に問題
仮説リテラル間の排他性制約を導入
 - 単一化(2)を支える評価関数
 - 評価関数の最大化による問題

e.g., $\text{cat}(x), \text{dog}(x)$ は同時に仮説できない
(ILP 制約による表現については予稿を参照)

→ 単一化すべきでないものを单一化してしまう
(オーバーマージ問題)

 - 単一化前: $H_1 = (\exists x, y) \text{cat}(x) \wedge \text{run}(x) \wedge \text{dog}(y) \wedge \text{run}(y)$
 - 単一化後: $H_2 = (\exists x) \text{cat}(x) \wedge \text{run}(x) \wedge \text{dog}(x)$

評価実験

- 実験の目的
 1. ILP による定式化の推論効率を評価する
 2. 推論により得られた仮説の質を評価する
- データセット
 - 背景知識
 - WordNet 3.0 (Fellbaum 98), FrameNet 1.5 (Ruppenhofer+ 10)
 - 約37万の推論規則
 - IS-A 関係以外を排他的概念とした
 - 観測 (入力)
 - RTE-2 開発セット
 - 1,600 文章, 平均 20.3 リテラル
- ツール
 - 論理式への変換: Boxer (Bos 08)
 - ILP ソルバー: Gurobi Optimizer 4.6

推論効率の評価実験

- 後ろ向き推論の深さを変えたとき、推論時間はどのように変化するか

深さ	潜在仮説集合 の平均サイズ	2分以内に最適解 を得られた問題	平均推論時間
1	89	99.8%	0.33 秒
2	479	98.4%	1.07 秒
3	1,171	95.8%	2.63 秒

– 比較参考値

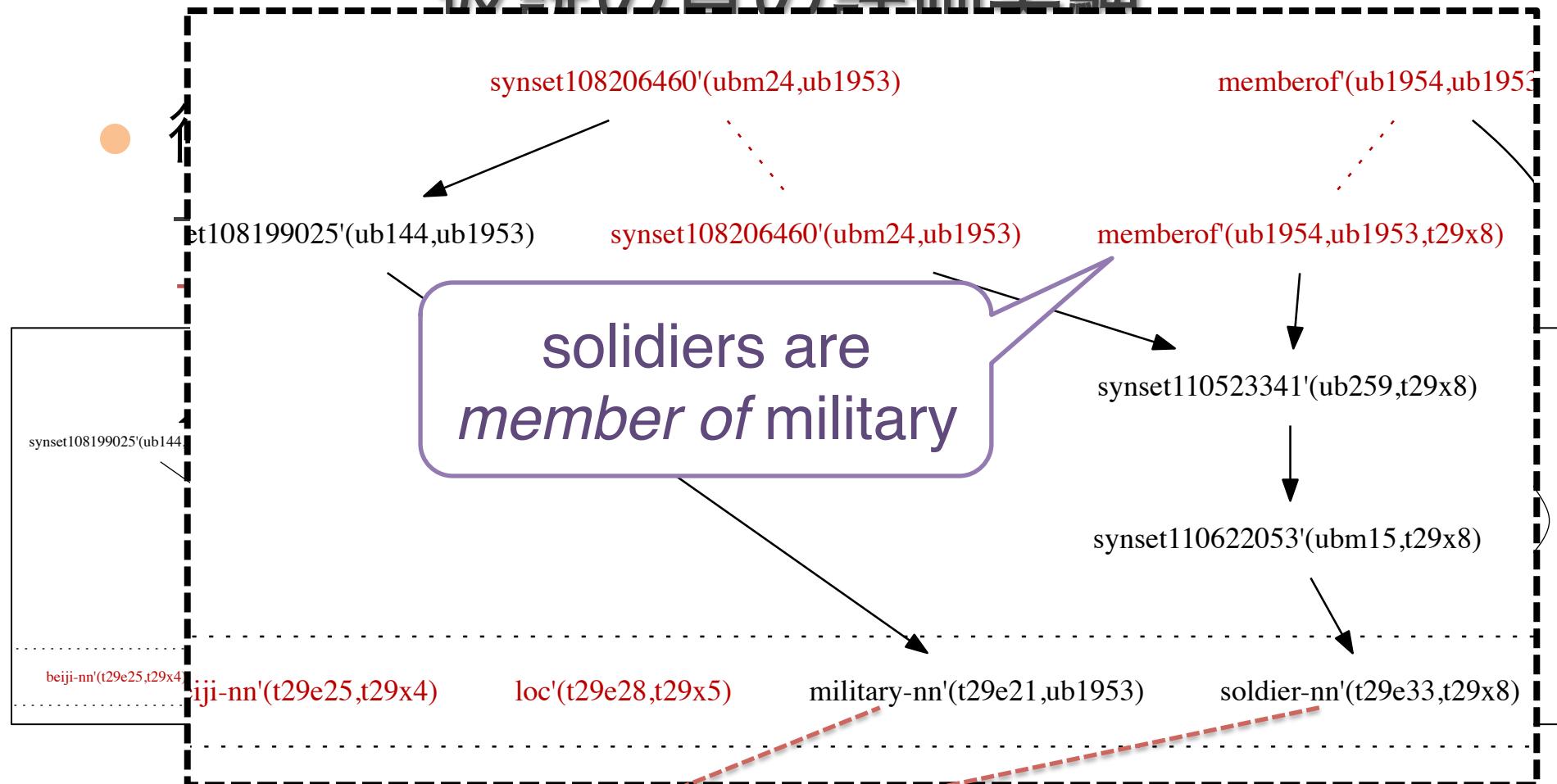
- Mini-TACITUS (Mulkar-Mehta+ 07): 30分以上 (深さ 3)
- Markov Logic Networks (Richardson & Domingos 05)による実装 (Blythe+ 11): 7分

→ 既存の実装よりも大幅に効率的だとわかった

仮説の質の評価実験

- 得られる仮説のサイズはどれぐらいか
 - 平均 1.9 リテラル (深さ1), 4.9 (深2), 7.4 (深3)
→ 新たな情報を顕在化できていることがわかった
- 顕在化した情報と談話の内容に矛盾はないか
 - 30 個の仮説をサンプリング, 人手チェック

仮説の質の評価実験



観測
(入力)

Also Friday, five Iraqi **soldiers** were killed and nine wounded in a bombing, targeting their convoy near Beiji, 150 miles north of Baghdad, the Iraqi **military** said.

仮説の質の評価実験

synset108913434'(ubm19,ub1953)

synset108913434'

913434'(ubm19,ub1953)

synset10819902

'ub1,t29x8)

synset109714694'(ub2,t29x0)

(x) Baghdad is
part of military

memberof(ub296,ub1953,t29x0)

partof(ub614,ub1953)

)
た

か

synset108914193'(ub9

14,ub1953,t29x5)

14193'(ub97,t29x5)

観測
(入力)

Also Friday, five Iraqi soldiers were killed and nine wounded in a bombing, targeting their convoy near Beiji, 150 miles north of Baghdad, the Iraqi military said.

仮説の質の評価実験

- 得られる仮説のサイズはどれぐらいか
 - 平均 1.9 リテラル (深さ1), 4.9 (深2), 7.4 (深3)
→ 新たな情報を顕在化できていることがわかった
 - 顕在化した構造の内容に矛盾はないか
 - 30 個の *part-of* クエリ
→ 正しい
- 解決の方針: 単一化の可能性 = 意味的互換性 + a (統語的な特徴など)
(共参照解析モデルのアナロジーで、学習可能な单一化スコア関数を設計中)
- 誤った仮説 (7事例):
- e.g., By 1979, the literacy rate in Cuba was higher than 90%, comparable to the rates in the United States and other developed countries
- 誤った单一化
(cuba(x) ∧ country(y) に対して $x=y$ を仮説) 19

まとめ

- 仮説推論の基盤技術の開発における二つの問題に取り組んだ
 - 1. 計算効率の問題
 - ILP による定式化を提案
 - 既存の実装より大幅に効率的であることを確認した
 - 2. オーバーマージ問題
 - 概念間の排他性制約を ILP 制約として導入
 - 意味的互換性以外の特徴も考慮することが必要
- 今後の課題
 - 仮説の評価関数の学習 (現在進行中!)
 - 要素仮説のコスト関数
 - 単一化のコスト関数
 - 談話解析の評価方法の検討