

第2回ナレッジグラフ推論チャレンジ2019 応募シート

「ナレッジグラフ埋め込みに基づく犯人探し」

2019年12月8日
株式会社KDDI総合研究所
黒川 茂莉

氏名
黒川 茂莉

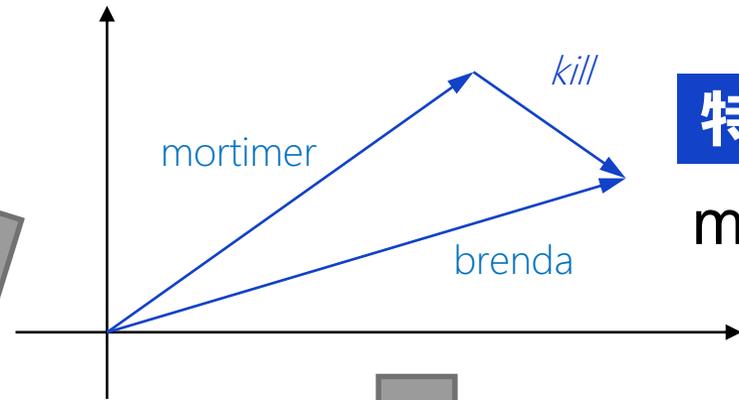
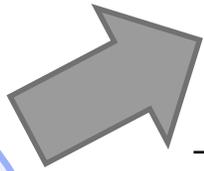
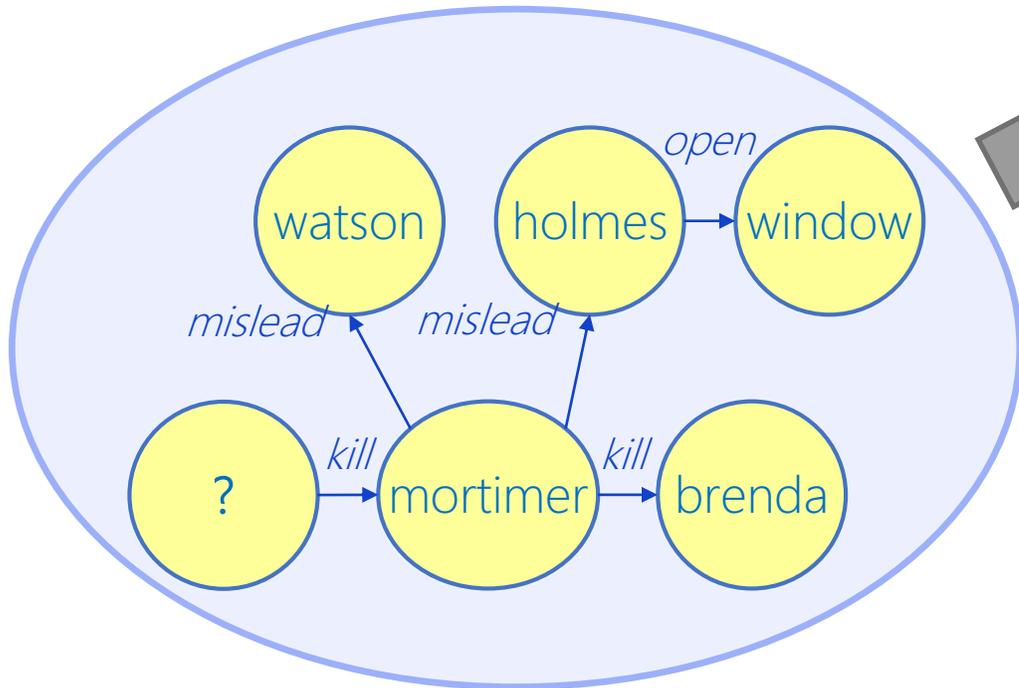
所属
株式会社KDDI総合研究所

メールアドレス
mo-kurokawa@kddi-research.jp

- 推論方法の概要
- 推論方法のポイント
- 推論過程の説明
- 利用データ、パラメータ
- 推論結果
- 推論結果の説明
 - (1)特徴空間の可視化
 - (2)理由説明の試行
- その他
 - (1)利用したツール、実行方法、実行環境
 - (2)実行時間、使用メモリ量
- まとめ
- 謝辞
- 参考文献
- 備考

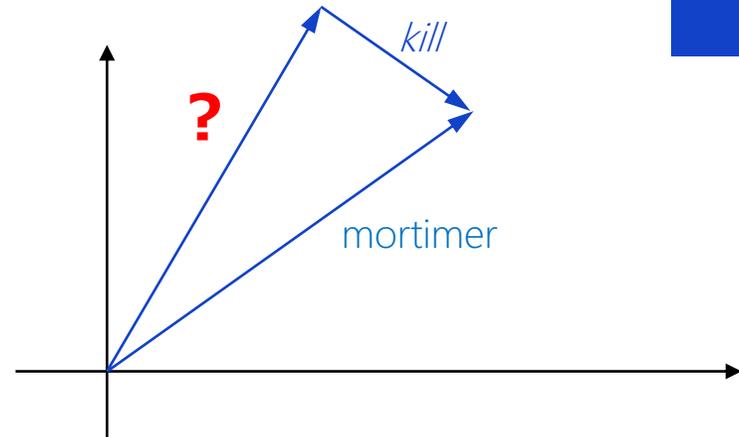
- ナレッジグラフの要素（ホームズ、ワトソン、被害者、凶器、犯行場所等）を特徴空間に埋め込み、特徴空間上のベクトル演算に基づいて犯人を推論する

▼ 悪魔の足ナレッジグラフ



特徴空間に埋め込み

$$\text{mortimer} + \text{kill} \sim \text{brenda}$$

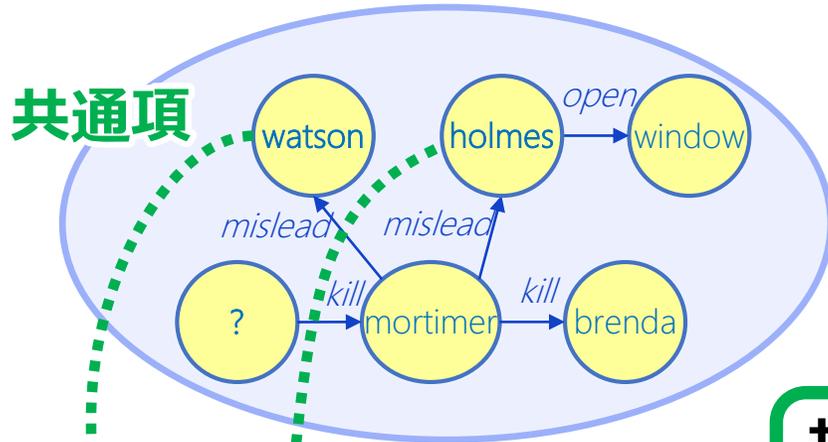


犯人の推論

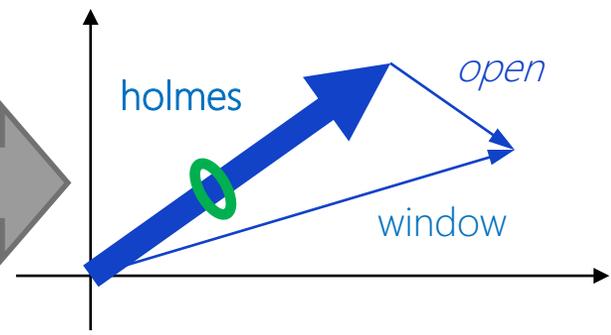
$$? + \text{kill} \sim \text{mortimer}$$

■ 複数ナレッジグラフに拡張

▼ 悪魔の足ナレッジグラフ



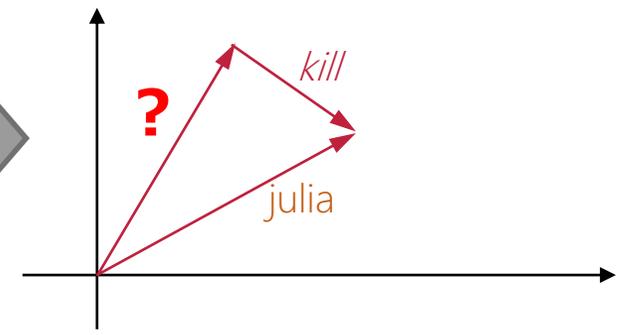
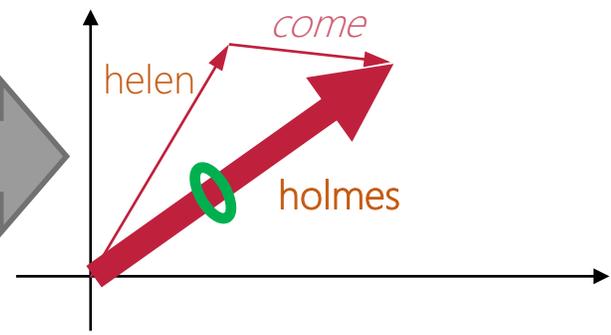
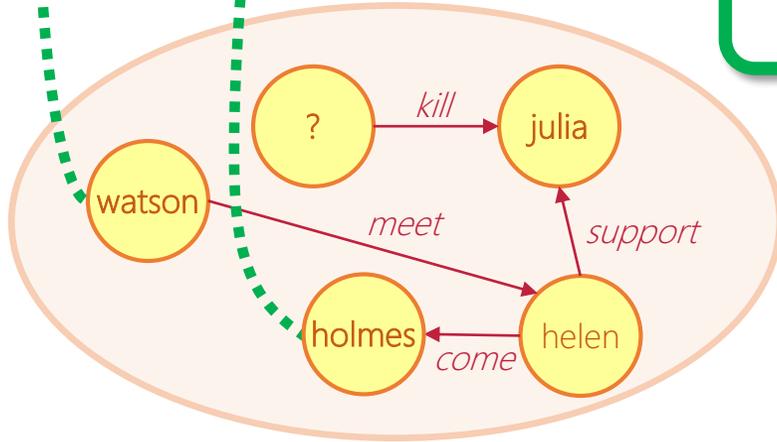
特徴空間に埋め込み



共通項(holmes, watson)のベクトル
が同じベクトルになるように制約

犯人の推論

?+kill~julia



▲ まだらの紐ナレッジグラフ

ポイント1: 被害者の近傍探索 (ナレッジグラフの要素を埋め込んだ特徴空間上で)

- ナレッジグラフの要素を特徴空間に埋め込むことで、特徴空間上で被害者の近傍探索を行うことにより関係する人物を特徴空間から探ることが可能になる

ポイント2: *kill* 関係を考慮した被害者の近傍探索

- 悪魔の足ナレッジグラフにおいて「mortimer *kill* brenda」や「brenda *kill*(ed) case1」の関係が存在するため、特徴空間上で*kill*を表現でき、**?**+*kill*~mortimerという推論 (= mortimer-*kill*の近傍探索) が可能になる

ポイント3: 複数ナレッジグラフの援用

- 複数ナレッジグラフの共通項であるholmes, watsonを用いて特徴空間を関連付けることで、他のナレッジグラフからの知識を流入させ、推論の精度を向上させることが可能になる

■ 処理フロー

① ナレッジグラフからの三つ組抽出と整形



② 三つ組の各要素の特徴空間への埋め込み

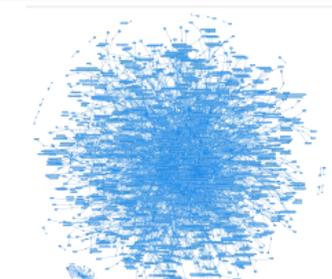


③ 特徴ベクトル演算に基づく犯人の推論

- 各ナレッジグラフにSPARQLで問い合わせを行い「ヘッド, リレーション, テイル」の三つ組を抽出
 - 「第1回ナレッジグラフ推論チャレンジ」における野村総合研究所チームの手法と同様 * 次ページ
- ヘッド, リレーション, テイルの名称を全て小文字に変換 (例: Holmes→holmes)
- ヘッドとテイルに含まれるエンティティの名称の先頭に小説タイトルを追記 (例: standale→DevilsFoot_standale)
 - エンティティは、同じ名称でも小説毎に異なる意味を持つ想定
ただし、holmes、watsonは小説間で共有
- リレーションの名称はレンマ化をして動詞の語形変化を除去 (例: guessed→guess)
 - レンマ化にはNatural Language Toolkit (<https://www.nltk.org/>) のWordNetLemmatizerを利用
 - リレーションは、同じ名称であれば小説が異なっても同じ意味を持つと想定し、小説間で共有

①オントロジーデータから、SPARQLで、主語述語目的語のRDFデータを抽出

- 手順の目的
 - 関係性解析をテンソル形式で行うため、ナレッジグラフを変換する。
- 手順の概要
 - 以下のSPARQLファイルで、主語述語目的語、場所FROM、場所TO、時間、理由 データを抽出



ナレッジグラフ全体
<http://knowledge-graph.jp/visualization/>

+

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX kgc: <http://kgc.knowledge-graph.jp/ontology/kgc.owl#>
PREFIX kd: <http://kgc.knowledge-graph.jp/data/SpeckledBand/>
SELECT ?id ?s ?p ?o ?o1 ?o2 ?o3 ?o4 ?o5
WHERE {
  ?id kgc:subject ?s.
  OPTIONAL (?id kgc:hasProperty ?o .).
  OPTIONAL (?id kgc:hasPredicate ?o .).
  OPTIONAL (?id kgc:what ?o1 .).
  OPTIONAL (?id kgc:whom ?o1 .).
  OPTIONAL (?id kgc:to ?o1 .).
  OPTIONAL (?id kgc:where ?o1 .).
  OPTIONAL (?id kgc:from ?o2 .).
  OPTIONAL (?id kgc:when ?o3 .).
  OPTIONAL (?id kgc:time ?o4 .).
  OPTIONAL (?id kgc:why ?o5 .).
}
order by (?id)
                
```

抽出順	述語	主語	対象	場所	時間	理由	抽出順	何	理由	どうして
1	come	Helen	NA	NA	NA	house of Holmes	NA	NA	NA	NA
2	beScared	Helen	NA	NA	NA	NA	NA	NA	NA	NA
3	beUpset	Helen	NA	NA	NA	NA	NA	NA	NA	NA
4	notHave	Helen	NA	NA	NA	NA	NA	money	NA	NA
5	getMarried	Helen	NA	NA	NA	NA	within 2 months	NA	NA	NA
6	obtain	Helen	NA	NA	NA	NA	within 2 months	money	NA	NA
7	pay	Helen	Holmes	NA	NA	NA	NA	money	reward of request	NA
8	live	Helen	NA	mansion of Roylott	NA	NA	NA	NA	NA	NA

NRI Copyright (C) Nomura Research Institute, Ltd. All rights reserved.

4

引用元 : https://challenge.knowledge-graph.jp/submissions/2018/tamura/submission_tamura.pdf

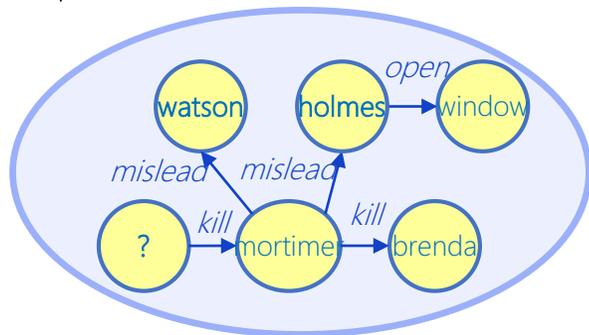
■ 後処理

- ヘッド(h)=主語、リレーション(r)=述語は上表より抽出
- 「対象」～「どうして」まで全てをテイル(t)とみなす
- $hrt_1t_2t_3\dots$ は、 hrt_1 、 hrt_2 、 $hrt_3\dots$ に分解

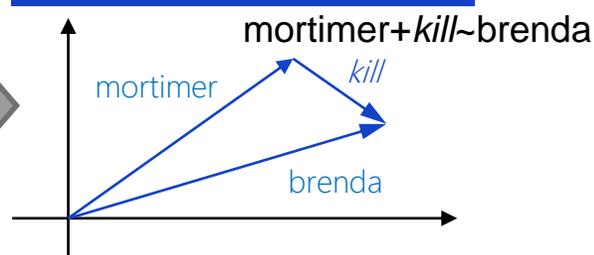
■ 単一ナレッジグラフの場合

- TransE[1] : $\vec{h} + \vec{r} \sim \vec{t}$ の関係ができる限り成り立つように埋め込みベクトルを学習

▼悪魔の足ナレッジグラフ



特徴空間に埋め込み



正例の $\|\vec{h} + \vec{r} - \vec{t}\|$ を小さく、負例の $\|\vec{h}' + \vec{r}' - \vec{t}'\|$ を大きく

$$\min O = \sum_{(h,r,t) \in T_1} \sum_{(h',r',t') \in T_1^-} \max(0, \|\vec{h} + \vec{r} - \vec{t}\| - \|\vec{h}' + \vec{r}' - \vec{t}'\| + \gamma)$$

s. t. $\forall \vec{h} \in T_1, \|\vec{h}\|=1, \forall \vec{r} \in T_1, \|\vec{r}\|=1, \forall \vec{t} \in T_1, \|\vec{t}\|=1$

数式	意味
T_1	対象のナレッジグラフの「ヘッド, リレーション, テイル」の集合
$\vec{h}, \vec{r}, \vec{t}$	T_1 の要素である「ヘッド, リレーション, テイル」の埋め込みベクトル
T_1^-	T_1 の各要素について「ヘッド, リレーション, テイル」のうちいずれか一つを別の要素で置き換えた「ヘッド', リレーション', テイル'」の集合 (負例サンプリング用の集合)
γ	マージン (損失関数のパラメータ)
$\ \ $	2乗ノルム

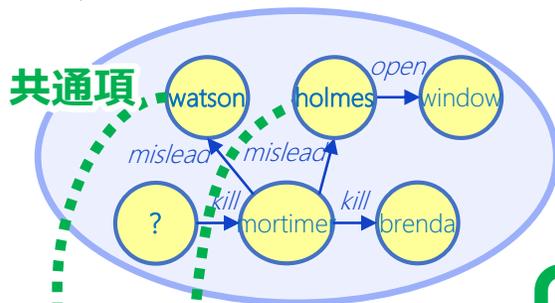
- リレーションのパス $\vec{p} = \vec{r}_1 \circ \vec{r}_2 \circ \dots$ を考慮した PTransE[2] も評価

■ 複数ナレッジグラフの場合

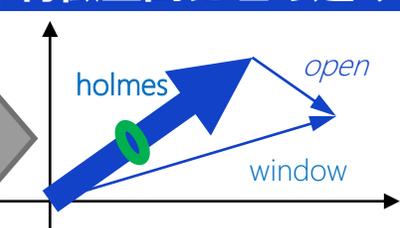
- ITransE/IPTransE[3] : TransE/PTransEに加え、**各ナレッジグラフの共通項を用いて特徴空間が関連づけられるように制約**して埋め込みベクトルを学習

※擬似的に共通項を増やすソフトアラインメントは実施しない（備考で実施した場合を比較）

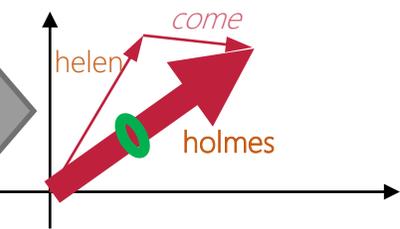
▼ 悪魔の足ナレッジグラフ



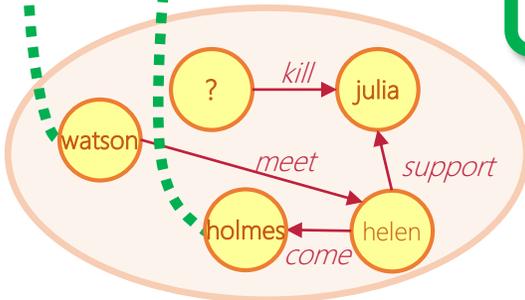
特徴空間に埋め込み



共通項のベクトルが
同じベクトルになるように制約



▲ まだらの紐ナレッジグラフ



$$\min O = \sum_{(h,r,t) \in T} \sum_{(h',r',t') \in T^-} \max(0, \|\vec{h} + \vec{r} - \vec{t}\| - \|\vec{h}' + \vec{r}' - \vec{t}'\| + \gamma)$$

$$\text{s.t. } \forall h \in T, \|\vec{h}\|=1, \forall r \in T, \|\vec{r}\|=1, \forall t \in T, \|\vec{t}\|=1$$

$$\forall (e_1, e_2, e_3, e_4, e_5) \in (E_1, E_2, E_3, E_4, E_5), \vec{e}_1 = \vec{e}_2 = \vec{e}_3 = \vec{e}_4 = \vec{e}_5$$

数式	意味
$T_i (i = 1, 2, \dots, 5)$	各ナレッジグラフの「ヘッド, リレーション, テイル」の集合
$T_i^- (i = 1, 2, \dots, 5)$	T_i の各要素について「ヘッド, リレーション, テイル」のうちいずれか一つを別の要素で置き換えた「ヘッド', リレーション', テイル」の集合（負例サンプリング用の集合）
T, T^-	$T_1 \cup T_2 \cup T_3 \cup T_4 \cup T_5, T_1^- \cup T_2^- \cup T_3^- \cup T_4^- \cup T_5^-$
$\vec{h}, \vec{r}, \vec{t}$	T の要素である「ヘッド, リレーション, テイル」の埋め込みベクトル
$E_i (i = 1, 2, \dots, 5)$	ナレッジグラフ間の共通項のエンティティの集合
$\vec{e}_i (i = 1, 2, \dots, 5)$	E_i の要素の埋め込みベクトル（例：各ナレッジグラフのholmesの埋め込みベクトル）

■ 悪魔の足 (case2)

🔍 犯人当て
(?, kill, mortimer)



$\overrightarrow{mortimer} - \overrightarrow{kill}$ の近傍のエンティティをランキング*
正解：standaleを当てられるか？

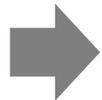
🔍 犯人を仮定したときの被害者当て
(standale, kill, ?)



$\overrightarrow{standale} + \overrightarrow{kill}$ の近傍のエンティティをランキング*
正解：mortimerを当てられるか？

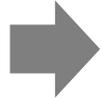
■ まだらの紐

🔍 犯人当て
(?, kill, julia)



$\overrightarrow{julia} - \overrightarrow{kill}$ の近傍のエンティティをランキング*
正解：roylottを当てられるか？

🔍 犯人を仮定したときの被害者当て
(roylott, kill, ?)



$\overrightarrow{roylott} + \overrightarrow{kill}$ の近傍のエンティティをランキング*
正解：juliaを当てられるか？

*いずれのランキングもユークリッド距離の近い順でソート

■ 事後フィルタ

- ランキングで人間でない要素が出てきた場合は、除外する
- ランキングでholmes、watsonが出てきた場合は、除外する

■ 利用データ

5KG...5つのナレッジグラフ全て使用の場合

2KG...複数ナレッジグラフ(悪魔の足+まだらの紐)使用の場合

1KG...単一ナレッジグラフ(悪魔の足)使用の場合

	悪魔の足	まだらの紐	同一事件	背中の曲 がった男	踊る人形
使用ID (フル)	1-489	1-401	1-580	1-124	1-231
使用ID (-10%)	1-440	1-360	1-522	1-111	1-207
使用ID (-25%)	1-366	1-300	1-435	1-93	1-173
エンティティ数	579	266	648	121	226
リレーション数	208	110	163	61	71

297

469

利用データ、パラメータ

- 「②三つ組の各要素の特徴空間へ埋め込み」ステップで用いるパラメータ
 - マージン $\gamma=1$
 - 負例は正例1個に対し1個
 - 勾配降下法で学習（学習率=0.001）

■ 利用するナレッジグラフ の数を増やすと (1KG→2KG→5KG) 犯人の順位が上昇

🔍 犯人当て

<悪魔の足(case2)>

- ITransEもIPTransEも、
**5KG全てを用いた場合、
1位で犯人を当てている**

<まだらの紐>

- **5KG全てを用いた場合、
ITransEは2位、
IPTransEは3位
(上位については次ページ)**

		TransE	ITransE	
		1KG	2KG	5KG
悪魔の足(case2)	犯人当て	6 (348)	1 (2)	1 (2)
	犯人→被害者当て	3 (13)	1 (4)	1 (2)
まだらの紐	犯人当て	-	2 (2)	2 (2)
	犯人→被害者当て	-	2 (2)	1 (1)
平均順位		4.5	1.5	1.3
平均逆順位		0.25	0.75	0.88

		PTransE	IPTransE	
		1KG	2KG	5KG
悪魔の足(case2)	犯人当て	3 (5)	6 (151)	1 (1)
	犯人→被害者当て	1 (2)	2 (3)	1 (2)
まだらの紐	犯人当て	--	2 (2)	3 (5)
	犯人→被害者当て	--	2 (3)	1 (1)
平均順位		2.0	3.0	1.5
平均逆順位		0.67	0.42	0.83

※括弧内は事後フィルタ前の順位

■ ITransE(5KG)の順位詳細

	1位	2位
犯人当て	holmes	DevilsFoot_standale
犯人→被害者当て	holmes	DevilsFoot_mortimer
犯人当て	SpeckledBand_helen	SpeckledBand_roylott
犯人→被害者当て	SpeckledBand_julia	

⇒holmesは事後フィルタされるため、**1位**

⇒holmesは事後フィルタされるため、**1位**

⇒helenに次いで**2位**

⇒**1位**

■ ITransE(5KG)の順位詳細

	1位	2位	3位	4位	5位
犯人当て	DevilsFoot_standale		⇒ 1位		
犯人→被害者当て	DevilsFoot_404	DevilsFoot_mortimer	⇒404(人間でない)は事後フィルタされるため、 1位		
犯人当て	SpeckledBand_helen	SpeckledBand_roma	SpeckledBand_window	SpeckledBand_safe	SpeckledBand_roylott
犯人→被害者当て	SpeckledBand_julia				

⇒**1位**

⇒window、safe(人間でない)は事後フィルタされるため、**helen、romaに次いで3位**

推論結果(3)

■ フル

		ITransE		IPTransE	
		2KG	5KG	2KG	5KG
悪魔の足(case2)	犯人当て	1 (2)	1 (2)	6 (151)	1 (1)
	犯人→被害者当て	1 (4)	1 (2)	2 (3)	1 (2)
まだらの紐	犯人当て	2 (2)	2 (2)	2 (2)	3 (5)
	犯人→被害者当て	2 (2)	1 (1)	2 (3)	1 (1)
平均順位		1.5	1.3	3.0	1.5
平均逆順位		0.75	0.88	0.42	0.83

■ -10%

		ITransE		IPTransE	
		2KG	5KG	2KG	5KG
悪魔の足(case2)	犯人当て	6 (16)	6 (6)	10 (479)	6 (7)
	犯人→被害者当て	1 (4)	1 (1)	2 (5)	2 (2)
まだらの紐	犯人当て	1 (1)	2 (2)	1 (1)	2 (3)
	犯人→被害者当て	1 (1)	2 (2)	2 (2)	1 (1)
平均順位		2.3	2.8	3.8	2.8
平均逆順位		0.79	0.54	0.53	0.54

■ -25%

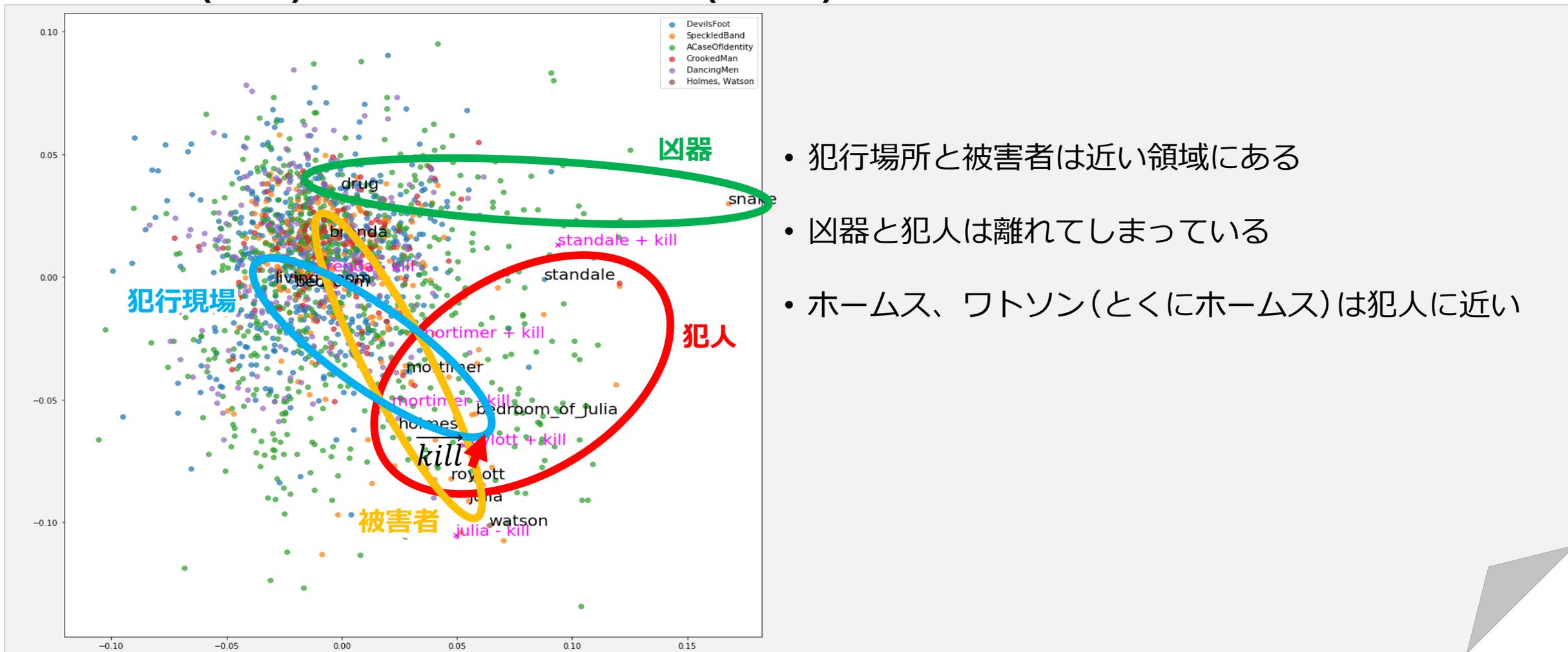
		ITransE		IPTransE	
		2KG	5KG	2KG	5KG
悪魔の足(case2)	犯人当て	7 (13)	10 (87)	5 (9)	11 (322)
	犯人→被害者当て	2 (7)	2 (2)	5 (9)	2 (2)
まだらの紐	犯人当て	2 (3)	2 (2)	4 (8)	2 (4)
	犯人→被害者当て	3 (7)	1 (1)	2 (3)	1 (1)
平均順位		3.5	3.8	4.0	4.0
平均逆順位		0.37	0.53	0.29	0.52

・利用するナレッジ（使用ID）を減らすと、平均順位が低下 = 精度が悪化
(ITransEの場合1.3→2.8→3.8)
(IPTransEの場合1.5→2.8→4.0)

※括弧内は事後フィルタ前の順位

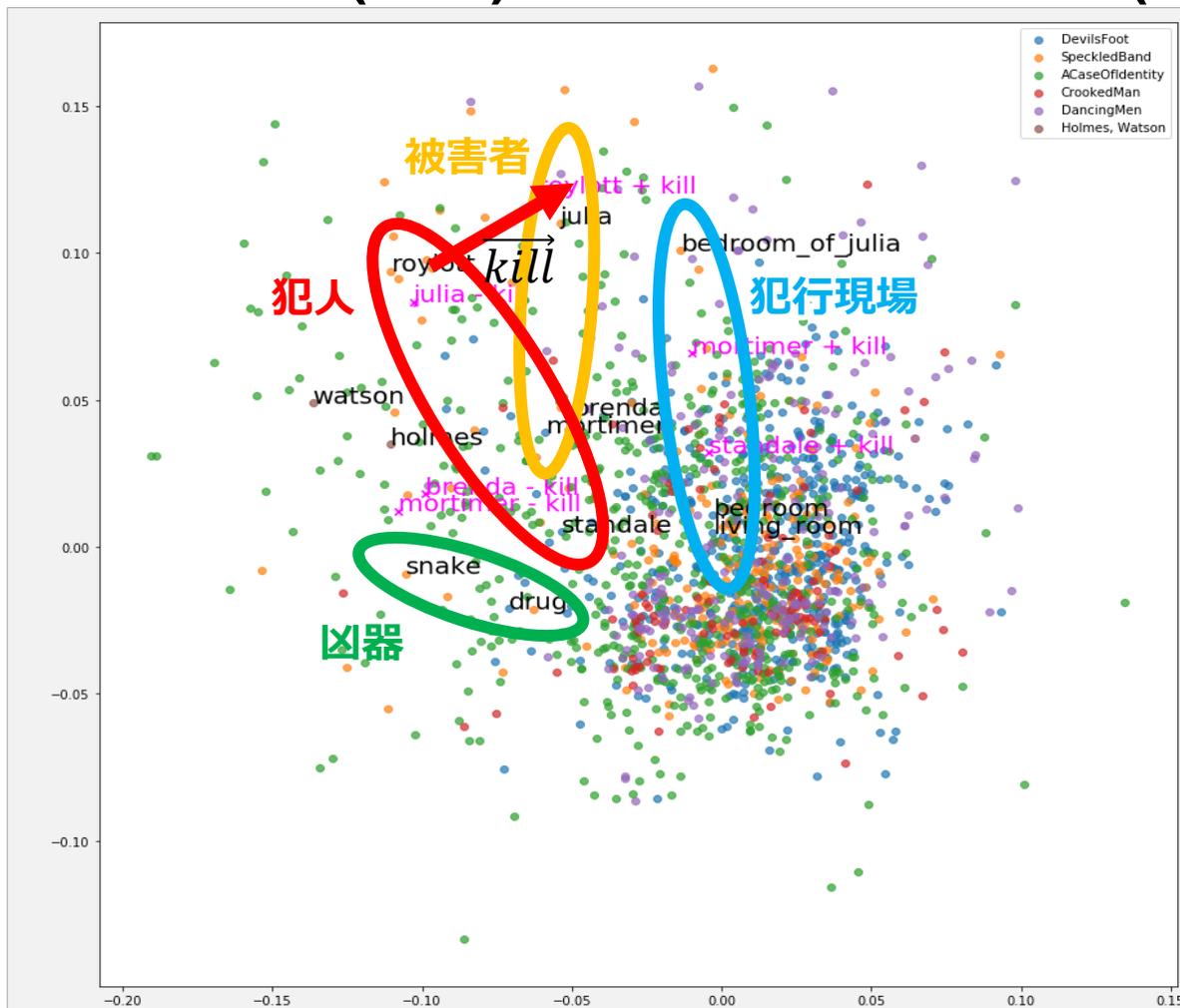
<特徴空間は事件の構造を捉えられているか？>

■ ITransE(5KG)の特徴空間の2D可視化(t-SNE)



<特徴空間は事件の構造を捉えられているか？>

■ IPTransE(5KG)の特徴空間の2D可視化(t-SNE)

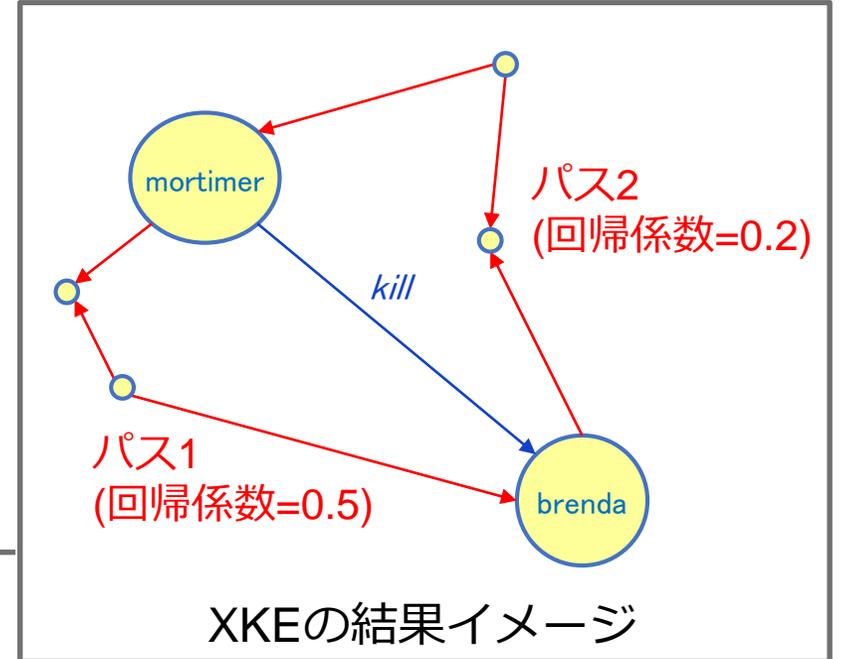


- 犯行場所は被害者に近い領域にある
- 凶器は犯人に近い領域にある
- ホームス、ワトソン(とくにホームス)は犯人に近い
- \vec{kill} は犯人と被害者をつなぐノルム、方向を持っているように見える

〈なぜ悪魔の足で犯人を当てられたか?〉

■ ナレッジグラフ埋め込みの説明法 (XKE[4]) により \vec{kill} を説明する重要パス探索を試行

- 正例作成：killを含む三つ組から $\|\vec{h} + \vec{r} - \vec{t}\|$ が一定値未満の点を抽出し、特徴量=ヘッドとテイルを結ぶパスの有無を表すマルチホットベクトル、ラベル=1の正例サンプルを作成
- 負例作成：killを含む三つ組の一部に対し、テイルを変えたラベル=-1の負例サンプルを作成
- 以上の正例、負例集合をもとに線形回帰を実行し、回帰係数を導出



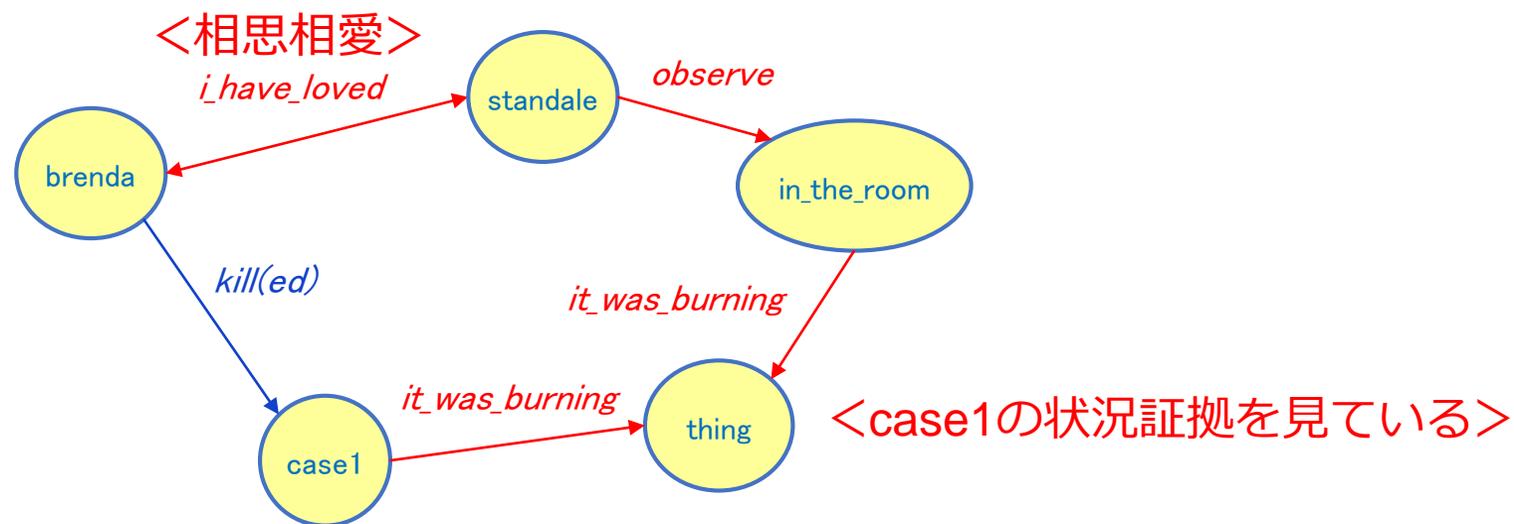
■ IPTransE(5KG)の重要パス1位、2位が大きな回帰係数

パス	回帰係数
1位 -i_have_loved-observe-it_was_burning-it_was_burning-	0.968
2位 -i_have_loved-observe-it_was_burning-it_was_burning-	0.968
-follow-shoot-shoot-_beshot-	0.122
-follow-shoot-sit-_dead-	0.122
...	

■ 重要パス1位、2位

	パス	回帰係数
1位	-i_have_loved-observe-it_was_burning-it_was_burning-	0.968
2位	-i_have_loved-observe-it_was_burning-it_was_burning-	0.968

■ 重要パス1位の意味 (=2位も同様)



standaleがmortimerを殺した動機に関連付けられている

ただし、case1かつ受け身=kill(ed)の説明であり、また、学習用サンプルが少ないため、解釈の妥当性については引き続き精査を行う

■ 利用したツール

- PTransE: <https://github.com/thunlp/KB2E/tree/master/PTransE>
- ITransE: <https://github.com/thunlp/IEAJKE>
- XKE: <https://github.com/arthurcgusmao/XKE>

■ 各手法の実行方法

- ITransE...上記ツールをそのまま実行（ソフトアラインメントを外して実行）
- TransE... ITransEに対し、共通項なしの場合に相当（共通項を与えずに実行）
- IPTransE... 実装が公開されていないため、ITransEにPTransEを独自に組み込んで実行（ソフトアラインメントを外して実行）
- PTransE... IPTransEに対し、共通項なしの場合に相当（共通項を与えずに実行）
- XKE... 埋め込みベクトルを直接入力できるように改変し、実行

■ 計算機のスペック

- MacBook Pro 2016（CPU : Intel Core i5 2.9GHz、メモリ : 16GB RAM）

その他(2) : 実行時間、使用メモリ量

■ 実行時間

- 9.0秒 (ITransE、5つのナレッジグラフ全て使用の場合)
- 112.5秒 (IPTransE、5つのナレッジグラフ全て使用の場合)

■ 使用メモリ量

- 約1.2MB (ITransE、5つのナレッジグラフ全て使用の場合)
- 約4.3MB (IPTransE、5つのナレッジグラフ全て使用の場合)

■ 犯人の推論

- ナレッジグラフの埋め込みに基づき推論した結果、悪魔の足では犯人を当てることに成功
* ITransEで5つのナレッジグラフ全てを用いた場合

- 悪魔の足では犯人 (standale) を1位で当てることに成功
- まだらの紐では犯人 (roylott) は2位 (1位はhelen)

- 利用するナレッジグラフの数を増やすと (1KG→2KG→5KG)、精度 (ランキング) が上昇

⇒他のナレッジグラフからの知識流入により、犯人推論の精度向上が可能

- 利用するナレッジ (使用ID) を減らすと (フル→-10%→-25%)、精度が低下

■ 犯人の説明

- ナレッジグラフ埋め込みの可視化により、事件の構造をおおまかに捉えていることを発見
- ナレッジグラフ埋め込みの説明法を用い、動機に相当するリレーションパスを発見
* 解釈の妥当性については精査中

■ 本研究は、JST、CREST、J181401085の支援を受けたものである。

- [1] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O., Translating embeddings for modeling multi-relational data, In *Advances in neural information processing systems*, pp. 2787-2795, 2013.
- [2] Lin, Y., Liu, Z., Luan, H., Sun, M., Rao, S., & Liu, S., Modeling Relation Paths for Representation Learning of Knowledge Bases, In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 705-714, 2015.
- [3] Zhu, H., Xie, R., Liu, Z., & Sun, M., Iterative Entity Alignment via Joint Knowledge Embeddings, In *IJCAI*, pp. 4258-4264, 2017.
- [4] Gusmão, A. C., Correia, A. H. C., De Bona, G., & Cozman, F. G., Interpreting Embedding Models of Knowledge Bases: A Pedagogical Approach, *ICML Workshop on Human Interpretability in Machine Learning (WHI 2018)*, 2018.

■ ITransE/IPTransEで擬似的に共通項を生成するソフトアラインメントを用いた場合 (使用ID=フルの場合)

		ITransE		IPTransE	
		2KG	5KG	2KG	5KG
悪魔の足(case2)	犯人当て	3 (7)	3 (16)	6 (109)	1 (1)
	犯人→被害者当て	7 (293)	7 (169)	8 (345)	8 (152)
まだらの紐	犯人当て	2 (2)	2 (4)	2 (6)	2 (2)
	犯人→被害者当て	4 (169)	7 (143)	3 (49)	2 (32)
平均順位		4.0	4.8	4.8	3.3
平均逆順位		0.31	0.28	0.28	0.53

※括弧内は事後フィルタ前の順位

■ cf. ソフトアラインメントを用いない場合

		ITransE		IPTransE	
		2KG	5KG	2KG	5KG
悪魔の足(case2)	犯人当て	1 (2)	1 (2)	6 (151)	1 (1)
	犯人→被害者当て	1 (4)	1 (2)	2 (3)	1 (2)
まだらの紐	犯人当て	2 (2)	2 (2)	2 (2)	3 (5)
	犯人→被害者当て	2 (2)	1 (1)	2 (3)	1 (1)
平均順位		1.5	1.3	3.0	1.5
平均逆順位		0.75	0.88	0.42	0.83

※括弧内は事後フィルタ前の順位

・平均順位で、ソフトアラインメントを用いない場合のほうがよい。理由としては、確かな共通項は holmes、watsonのみであり、さらに擬似的に共通項を生成することが難しかったと考えられる。

■ 応募フォーム

● 公開の可否：

公開してよい

非公開とする

● 公開形式：

ナレッジグラフ推論チャレンジのサイトで公開

独自のサイトで公開してリンクを希望

■ 応募したプログラム, データ等

● 公開の可否：

公開してよい

非公開とする

