



共起を考慮した主成分回帰分析による 推理小説の犯人推定

東京都市大学大学院総合理工学研究科 情報専攻

勝島修平 穴田一

目次

アプローチの特徴と結論

既存研究

提案手法

実験結果

分析・考察

まとめと今後の課題

アプローチの特徴と結論

アプローチの特徴

- 犯人を**穴埋め問題**として推定
- 単語の**同時性**を考慮：CBoW
- 説明モデルの構築：**主成分回帰分析**

結論

- 小説「まだらの紐」の犯人を2位で推定
- 共起関係から学習過程を考察
- 追加知識によって犯人推定できることを示唆

既存研究

TransEによる埋め込み手法

- データを主語・述語・目的語のトリプルに加工
- ベクトルで表現したトリプルの関係によって犯人をランキング



犯人”Royslott”を2位で予想

既存研究の問題点

- トリプル間の時間の流れが欠けている

提案手法

犯人推定手法

課題

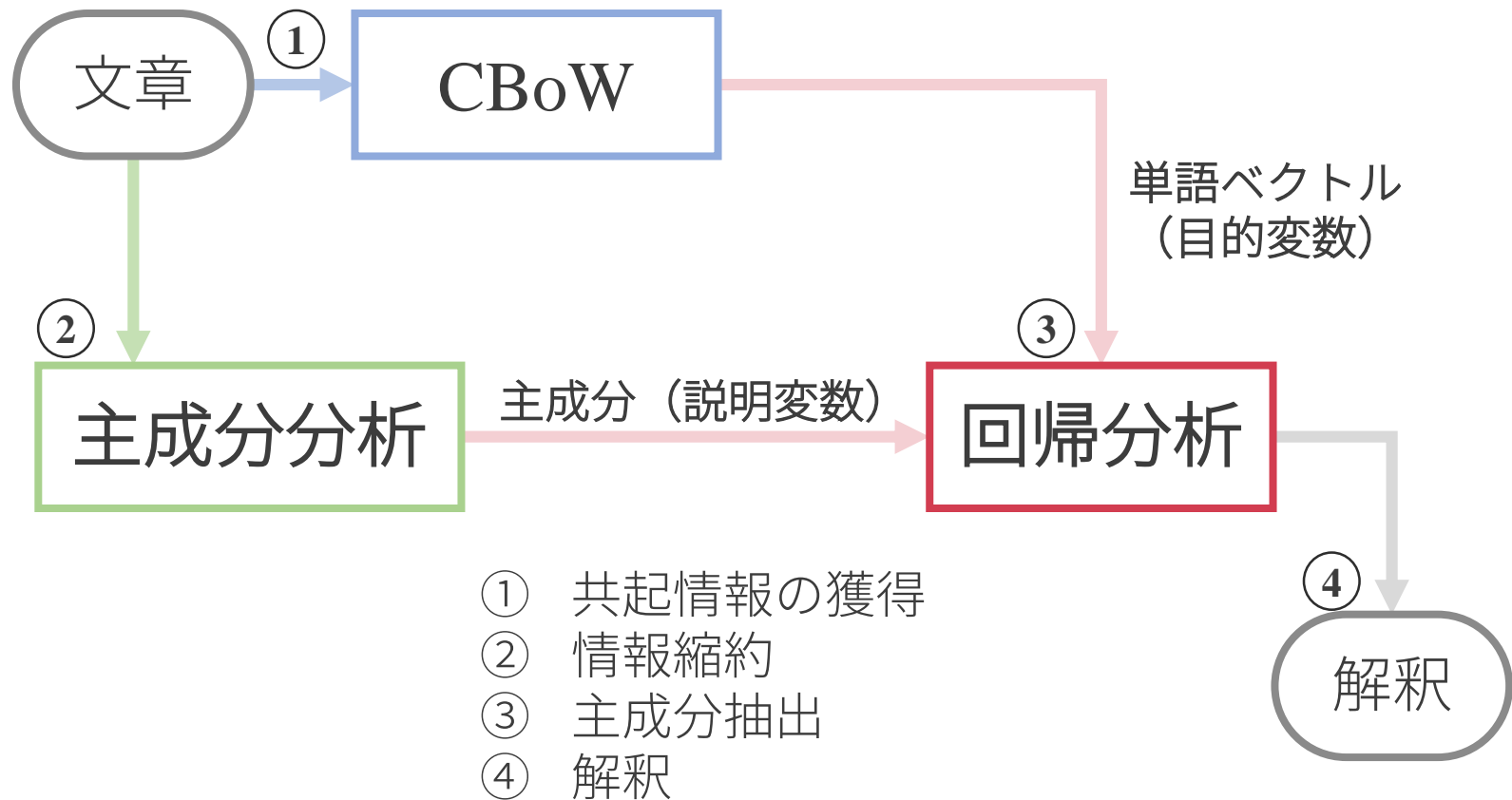
事象の同時性

提案手法

単語の共起の学習 + 主成分回帰分析

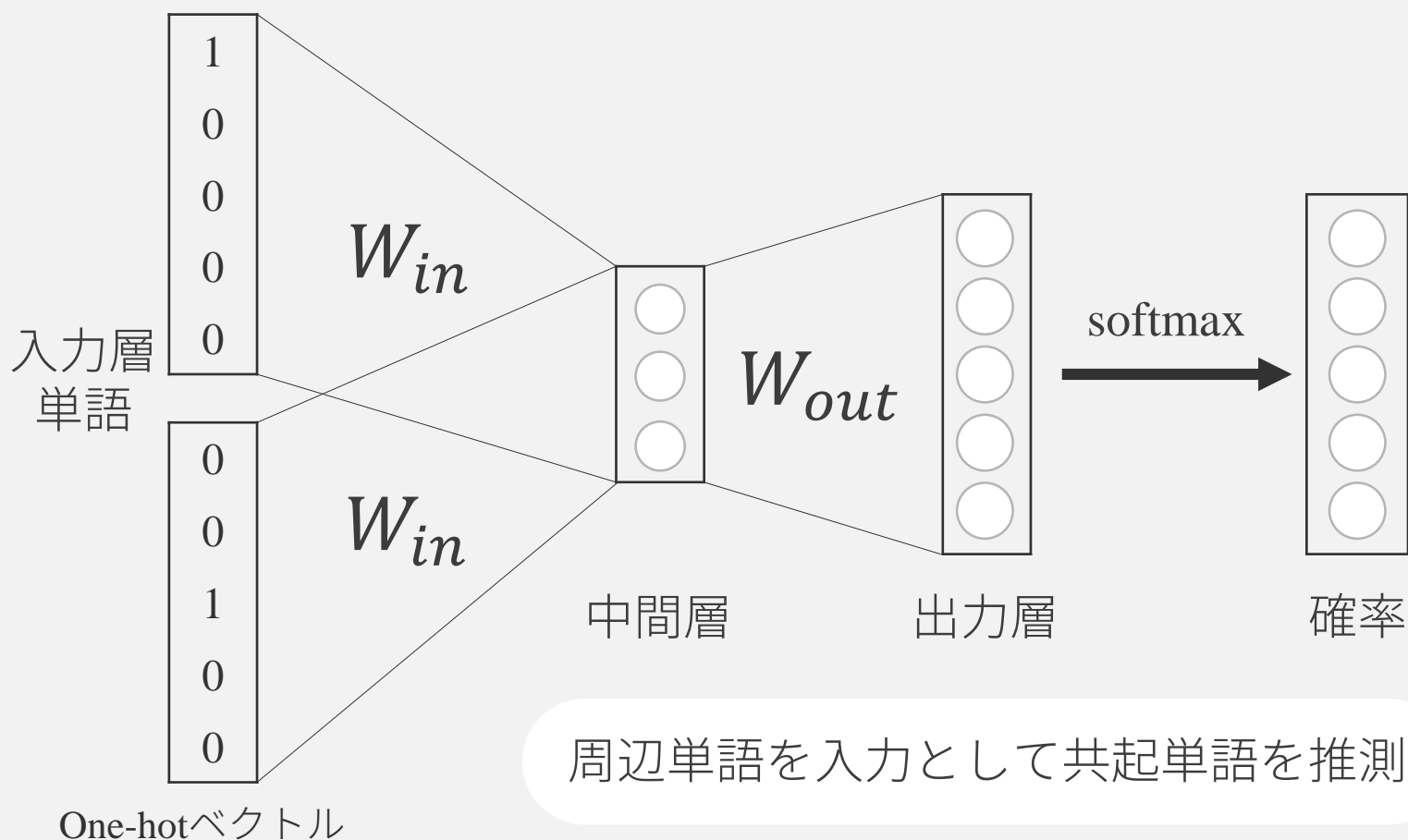
提案手法

提案手法の流れ



提案手法

CBoW (Continuous Bag-of-Words)



提案手法

C B O W による 共起情報の獲得

入力
データ

- 対象小説の犯人推定前まで（条件：不十分10%/25%）
- 追加小説(悪魔の足)の全データ

```
Query × +
1 PREFIX kgc: <http://kgc.knowledge-graph.jp/ontology/kgc.owl#>
2 #PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
3 #PREFIX kd: <http://kgc.knowledge-graph.jp/data/SpeckledBand/>
4 #PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5
6 SELECT ?id ?s ?v ?o ?k ?to ?time ?place ?reason
7 FROM <http://kgc.knowledge-graph.jp/data/SpeckledBand>
8 #FROM <http://kgc.knowledge-graph.jp/data/DancingMen>
9 #FROM <http://kgc.knowledge-graph.jp/data/ACaseOfIdentity>
10 #FROM <http://kgc.knowledge-graph.jp/data/DevilsFoot>
11 #FROM <http://kgc.knowledge-graph.jp/data/CrookedMan>
12 #FROM <http://kgc.knowledge-graph.jp/data/AbbeyGrange>
13 #FROM <http://kgc.knowledge-graph.jp/data/SilverBlaze>
14 #FROM <http://kgc.knowledge-graph.jp/data/ResidentPatient>
15 WHERE {
16   OPTIONAL {?id kgc:subject ?s .}
17   OPTIONAL {?id kgc:hasPredicate ?v.}
18   OPTIONAL {?id kgc:what ?o.}.
19   OPTIONAL {?id kgc:whom ?k.}.
20   OPTIONAL {?id kgc:to ?to.}.
21   OPTIONAL {?id kgc:when ?time.}.
```

SPARQLにて、
idに紐づいた単語を抽出

出力

共起単語の確率分布

提案手法

主成分分析による情報縮約

入力データ

マルチホットベクトル化した「まだらの紐」

出力

合成変数（主成分）

- マルチホットベクトル化により**単語の変数化**
- 相関を持った変数同士を主成分分析によって**合成**



解釈時には、主成分の持つ
因子負荷量の値を見て、分析

提案手法

回帰分析による'kill'に影響する**主成分抽出**

入力データ

主成分 + 目的変数

目的変数

$$L = \text{softmax}(CBoW(\text{contexts}))_{w_{kill}}$$

W_{kill} : *単語'kill', contexts :一場面中の単語

場面に対するkillのスコア

主成分の目的関数への回帰式

実験結果 (パラメータ)

次元数	100	エポック サイズ	300
ウィンドウ サイズ	all	バッチ サイズ	32
学習率	0.01	最適化	Adam

実験には殺人事件である
「まだらの紐」・「悪魔の足」・「僧坊荘園」
を用いる

実験結果(Test pattern.1)

犯人推定 (昨年度と同状況:簡易版)

犯人
Roylott

まだらの紐



['kill', 'Julia']をCBoWに入力し,人物のスコアをランキング

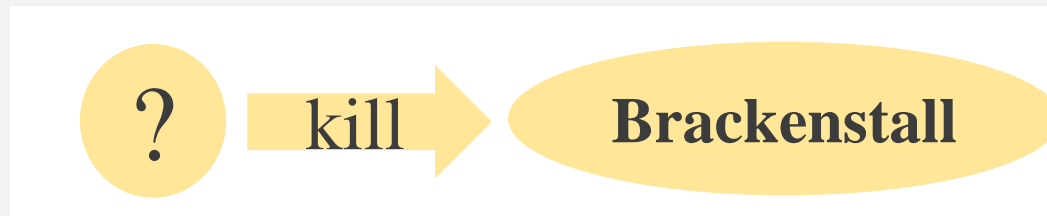
*ホームズ、ワトソン、被害者ジュリアを除いて

犯人を**穴埋め問題**として推定

実験結果(Test pattern.1)

犯人推定

僧坊莊園



犯人
Jack

他小説についても同様に調査

実験結果(Test pattern.2)

犯人推定（犯行状況の詳細）

犯人
Roylott

まだらの紐

Death_day_of_Julia

? kill Julia bedroom_of_Julia

犯行当時の状況を犯人推定の際の入力に加えて、
人物のスコアをランキング

*ホームズ、ワトソン、被害者ジュリアを除いて

実験結果(Test pattern.2)

犯人推定（犯行状況の詳細）

犯人
Jack

僧坊莊園

? kill Sir_Eustace_Brackenstall
Lady_Brackenstall beTied cord_of_bell

実験結果 (犯人当て:25%)

対象小説：25%	Pattern.1		Pattern.2	
「まだらの紐」 SpeckledBand ~ID300	+DevilsFoot		+DevilsFoot	
	Sum	3	Sum	2
	Ave	3	Ave	2
	Geom	3	Geom	2
「僧坊荘園」 AbbeyGrange ~ID310	+DevilsFoot		+DevilsFoot	
	Sum	3	Sum	3
	Ave	3	Ave	3
	Geom	3	Geom	3

順位が上昇

記載は対象小説の犯人の順位

→初期値による影響を考慮し、10試行の順位遷移により判断

- Sum: 順位の合計
- Ave: 順位の平均
- Geom: 順位の幾何平均

実験結果 (犯人当て:10%)

対象小説：10%	Pattern.1		Pattern.2	
「まだらの紐」 SpeckledBand ~ID360	+DevilsFoot		+DevilsFoot	
	Sum	2	Sum	2
	Ave	2	Ave	2
	Geom	2	Geom	2
「僧坊荘園」 AbbeyGrange ~ID357	+DevilsFoot		+DevilsFoot	
	Sum	1	Sum	1
	Ave	1	Ave	1
	Geom	1	Geom	1

推定ならず

一位で推定

記載は対象小説の犯人の順位

→初期値による影響を考慮し、10試行の順位遷移により判断

- Sum: 順位の合計
- Ave: 順位の平均
- Geom: 順位の幾何平均

実験結果

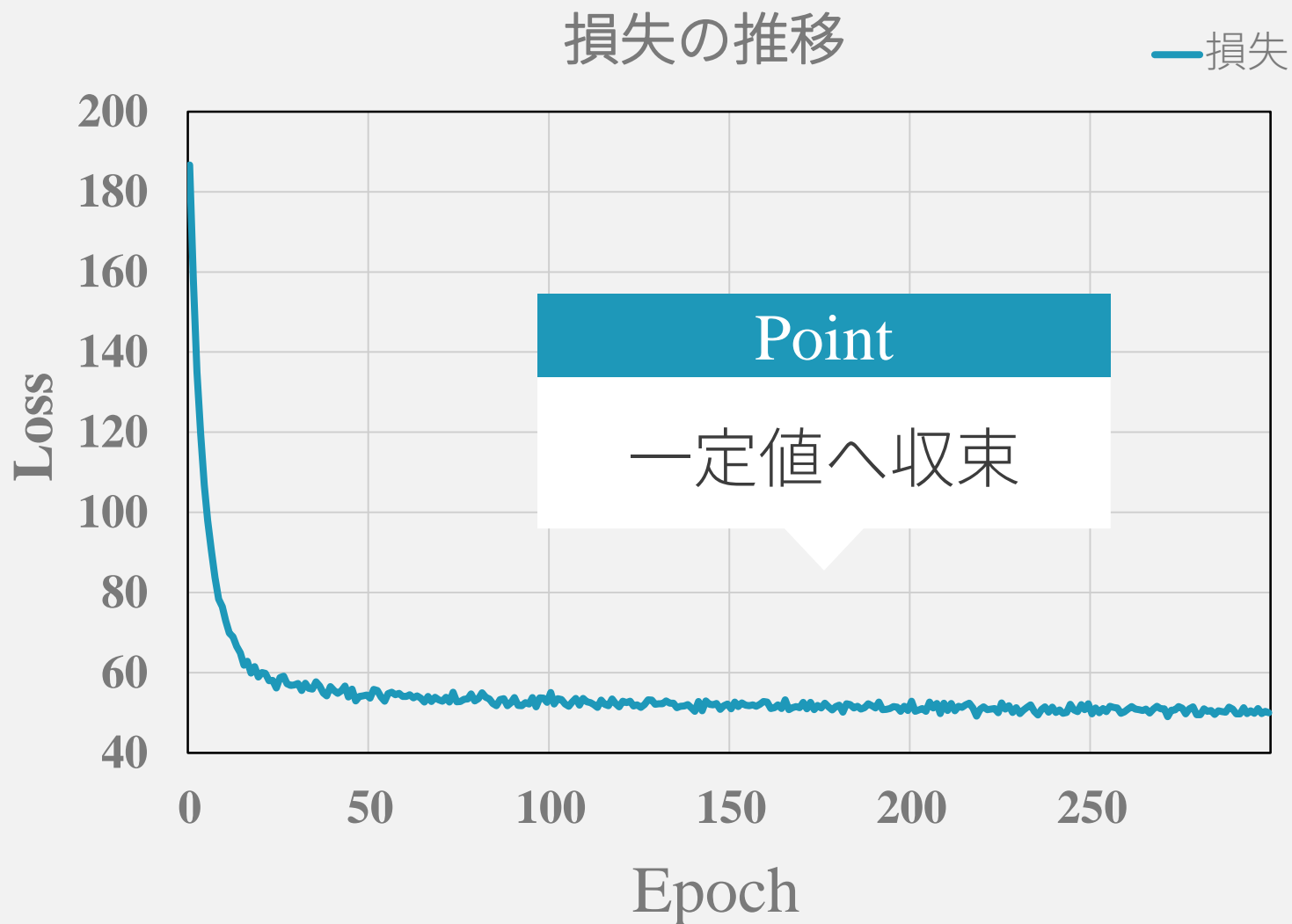
犯人推定理由の分析（条件10%）

- 小説「**まだらの紐**」の犯人は**第2位**で予想
- 小説「**僧坊荘園**」の犯人も**第1位**で予想

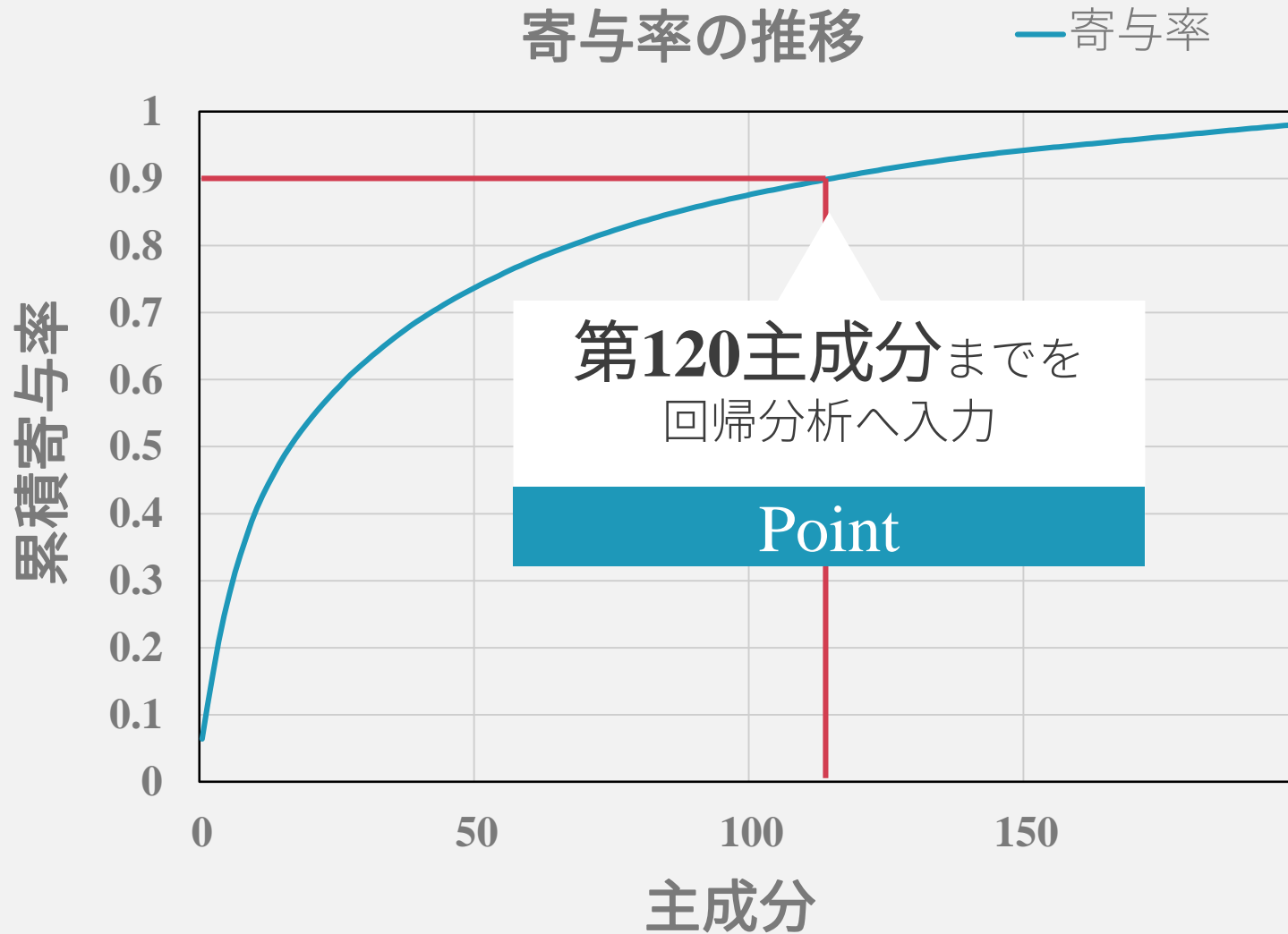
*ただし、「僧坊荘園」は初期値による変動が大きいため、回帰モデルが一意でないそのため、今回は考察から除外

1. なぜ、「**まだらの紐**」の犯人を**間違えたのか**？

実験結果 (CB_oW : まだらの紐)

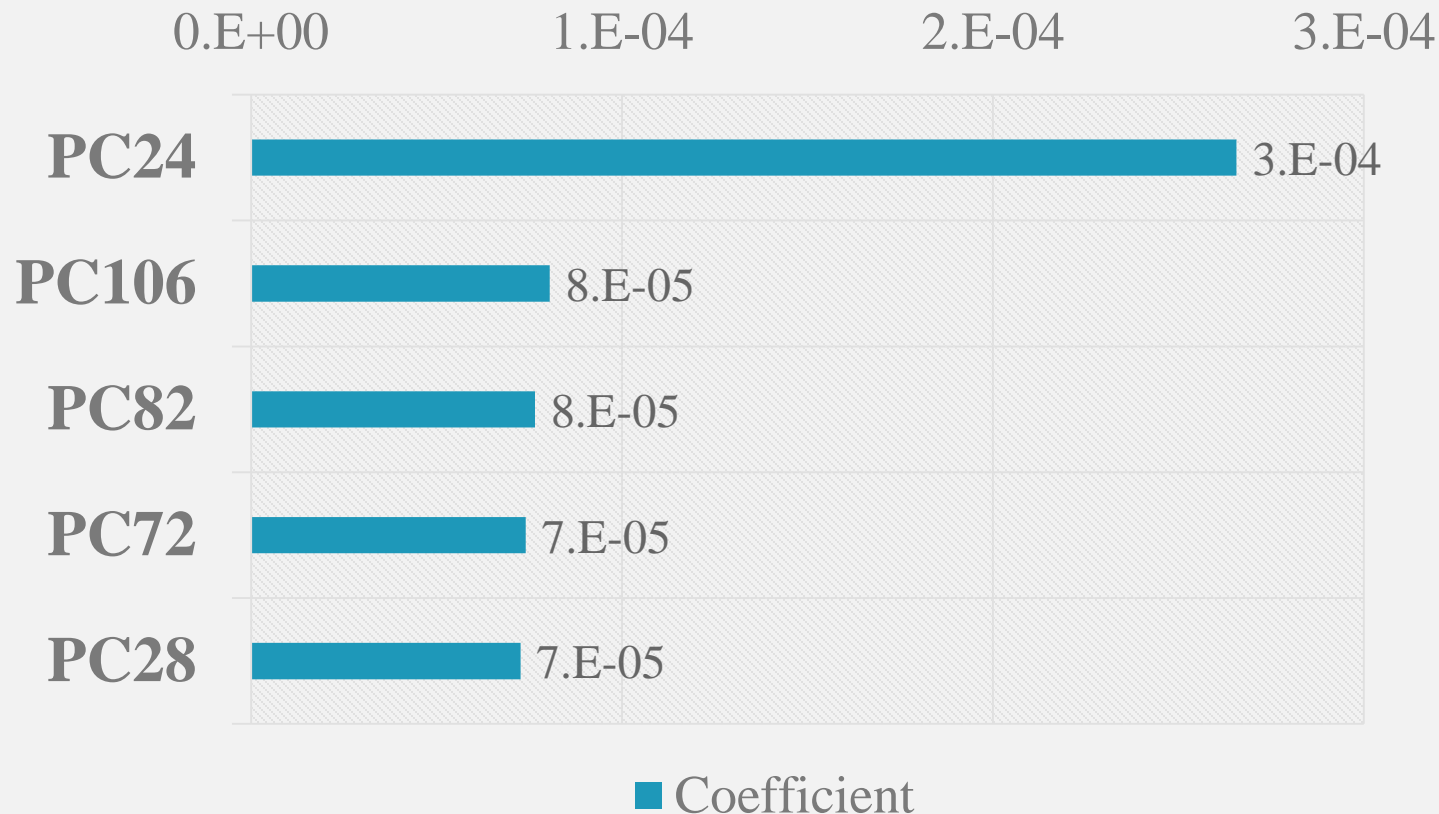


実験結果（主成分分析：まだらの紐）



実験結果（回帰分析：偏回帰係数：まだらの紐）

主成分ごとの偏回帰係数値：10試行平均



第24主成分についての検討を行う

分析・結果（まだらの紐）

因子負荷量の分析

まだらの紐

Death_day_of_Julia

? kill Julia bedroom_of_Julia



犯人'Helen'と予想

因子負荷量値の上位10個を見て解釈

分析・考察（まだらの紐）

因子負荷量（第24主成分）

money	0.55
getMarried	0.20

death_day_of_mother_of_Helen	0.14
------------------------------	------

cannotGo	0.13
----------	------

scream	0.13
--------	------

sister	0.12
--------	------

tak	0.12
-----	------

sp	0.11
----	------

Wa	0.10
----	------

death_	0.10
--------	------

きっかけ：母の死
動機：金銭（遺産）目的
の抽出

分析・考察（僧坊莊園）

| 考 察

第24主成分によって、
金欲による犯行動機を示唆

➡ では、**なぜ**犯人を間違えたのか？
（推定：Helen, 実際：Royslott）

分析・考察（まだらの紐）

推定理由の分析

money	0.55
getMarried	0.20
death_day_of_mother_of_Helen	0.14
cannotGo	0.13
scream	0.13
sister	0.12
take	0.12
spring	0.11
Watson	0.10
death_day_of_Julia	0.10

動機となる単語と
共起

Helenを示す代名詞

➡ 犯人を誤判定

分析・考察（僧坊莊園とまとめ）

考 察

- 単語sisterはJuliaとHelenの姉妹をさす言葉
→一つで複数の意味を持つ単語の影響を受けやすい
- 学習に用いた追加知識である「悪魔の足」には、「まだらの紐」に適用できる犯行手段に関する知識がないため、動機だけではRoyslottの推定はできない

ま と め

- 単語の共起によって推理過程を導出可能な一方、多義語に対するアプローチが必要
- 適切な知識を準備できれば犯人を推定可能

まとめと今後の課題

単語の共起情報を利用し、主成分回帰によって推定

- 小説「まだらの紐」の犯人を**第2位**で推定
- 推定理由を**主成分**によって分析
- 主成分の**学習過程**について考察

犯人**推定**と**分析**の両立

今後の課題

- 1 初期値依存性 入力データやさらなる層(attention)などの検討
- 2 限定的な知識 追加知識の導入
- 3 精度と説明性 回帰分析でなく, LRPなどの深層学習を用いた説明

より説明性の高いAIの開発を目指す

実行環境

プログラミング言語

Python3.7

プロセッサ

AMD Ryzen7 3700X
8-Core Processor

OS

Windows10

統合開発環境

Pycharm

共有について

可

モジュール

chainer

Thank you!

