



論理と計算

第1回

論理を用いた問題解決の概要

---

担当：尾崎 知伸

ozaki.tomonobu@nihon-u.ac.jp

オリエンテーション：本講義について

# 本講義について

- 履修区分：選択必修（C群）
- 授業概要
  - 命題・述語論理を対象に，論理に基づく知識の表現と推論・問題解決に関する基礎事項について概観する
- 到達目標
  - 命題論理・述語論理を用いた対象の表現と推論の仕組みを理解する
  - 論理や推論を用いた問題解決の発展について，技術的な側面から検討することができる
- 授業の方法
  - 電子資料を用いた講義形式を中心とする
  - テーマ毎に，演習（課題学習）を行う
- 質問応答
  - Slackで対応する．当該チャンネルを登録してください
    - 原則質問は，DMではなく，チャンネルに投稿してください
- 履修条件
  - 「離散数学」と「知識情報処理」を履修していることが望ましい

## 講義予定

※一部変更になる可能性があります

09/22	01. オリエンテーション と 論理を用いた問題解決の概要
09/29	02. 命題論理：構文・意味・解釈
10/06	03. 命題論理：推論
10/13	04. 命題論理：充足可能性問題
10/20	05. 命題論理：振り返りと演習（課題学習）
10/27	06. 述語論理：構文・意味・解釈
11/03	07. 述語論理：推論 ※文化の日，文理学部授業日
11/10	08. 述語論理：論理プログラムの基礎
11/17	09. 述語論理：論理プログラムの発展
11/24	10. 述語論理：振り返りと演習（課題学習）
12/01	11. 高次推論：発想推論
12/08	12. 高次推論：帰納推論の基礎
12/15	13. 高次推論：帰納推論の発展
12/22	14. 高次推論：振り返りと演習（課題学習）
01/19	15. まとめと発展的話題

## 資料について：原則電子資料を配布する

- 参考図書

- 古川康一・向井国昭, 『[数理論理学](#)』, コロナ社, 2008

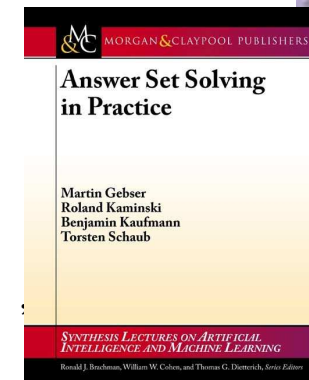
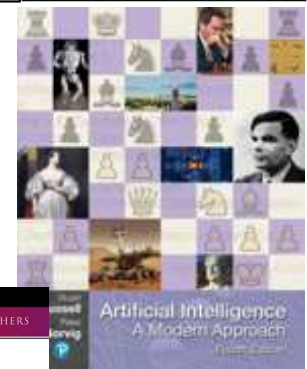
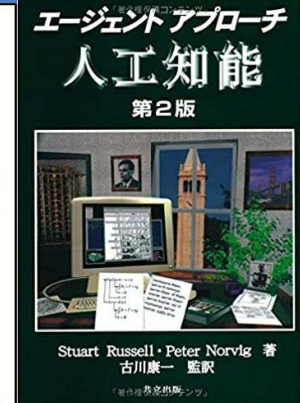
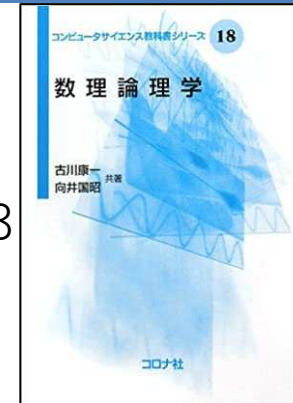
- S. Russell and P. Norvig(著), 古川康一 (監訳)  
『[エージェントアプローチ人工知能 第二版](#)』,  
共立出版, 2018

- S. Russell and P. Norvig(著),  
『[Artificial Intelligence: A Modern Approach, 4th ed.](#)』,  
Pearson, 2020

- M. Gebser 他(著),  
『[Answer Set Solving in Practice](#)』,  
Morgan and Claypool, 2012

- 参考資料：各種の解説論文（例えば以下など）

- 井上 克巳, 坂間 千秋, 佐藤 泰介: 「推論とAIプログラミング」,  
人工知能, 34(5):703-713, 2019
- 井上 克巳, 田村 直之: 「SATソルバーの基礎」, 人工知能学会誌, 25(1):57-67, 2010
- 宋 剛秀, 番原 睦則, 田村 直之, 鍋島 英知: 「SATソルバーの最新動向と利用技術」,  
コンピュータソフトウェア, 35(4):72-92, 2018



# 成績評価と演習環境について

## 【成績評価】

- 授業参画度：15%
  - 毎回の小課題，Slack等への質問，Zoomへのアクセス履歴等
- レポート：85%
  - 20%：充足可能性問題（SAT）に関する実験レポート（05回出題予定）
  - 20%：解集合プログラミング（ASP）に関する実験レポート（10回出題予定）
  - 20%：帰納論理プログラミング（ILP）に関する実験レポート（14回出題予定）
  - 20%：振り返りレポート（まとめ・補助教材の作成）（14回出題予定）
    - 05%：上記の相互評価（15回実施予定）
  - ※レポート未提出の場合は，E判定とする

## 【演習環境】

- レポートは，Latexを用いて作成しPDFファイルを提出すること
- 一部の例題に Processing プログラムを用いる
- SAT演習にはclasp，ASP演習にはclingo，ILP演習にはILASPを用いる
  - 参考：<https://doc.ilasp.com/installation.html> ですべてインストール可能
    - 管理者として実行しましょう
  - Ubuntu / WSL2 / Mac のいずれかを利用する

# 人工知能技術 と 論理・推論



## 人工知能って何？

「人工知能」とは何だと思うのでしょうか？まるで人間のようにふるまう機械を想像するのではないのでしょうか？これは正しいとも、間違っているともいえます。なぜなら、人工知能の研究には二つの立場があるからです。一つは、人間の知能そのものをもつ機械を作ろうとする立場、もう一つは、人間が知能を使ってすることを機械にさせようとする立場です(注1)。そして、実際の研究のほとんどは後者の立場にたっています。ですので、人工知能の研究といっても、人間のような機械を作っているわけではありません。

それでは、実際の研究ではどのようなことをしているのでしょうか？人工知能の研究には、[人工知能研究](#)で紹介するようにいろいろな分野があります。ここでは、この中から「推論」と「学習」を取り上げます。

第1・2次ブーム

「推論」とは「**知識をもとに、新しい結論を得ること**」です。

「学習」は何か機械が勉強をする感じがしますが、ここでは「**情報から将来使えそうな知識を見つけること**」です(注2)。

第3次ブーム

鳥と飛行機の関係

1. 人間の知能そのものを作る
2. 人間が知能を使ってすること（と同じこと）ができる機械を作る



# 人工知能の歴史

IBM ワトソン

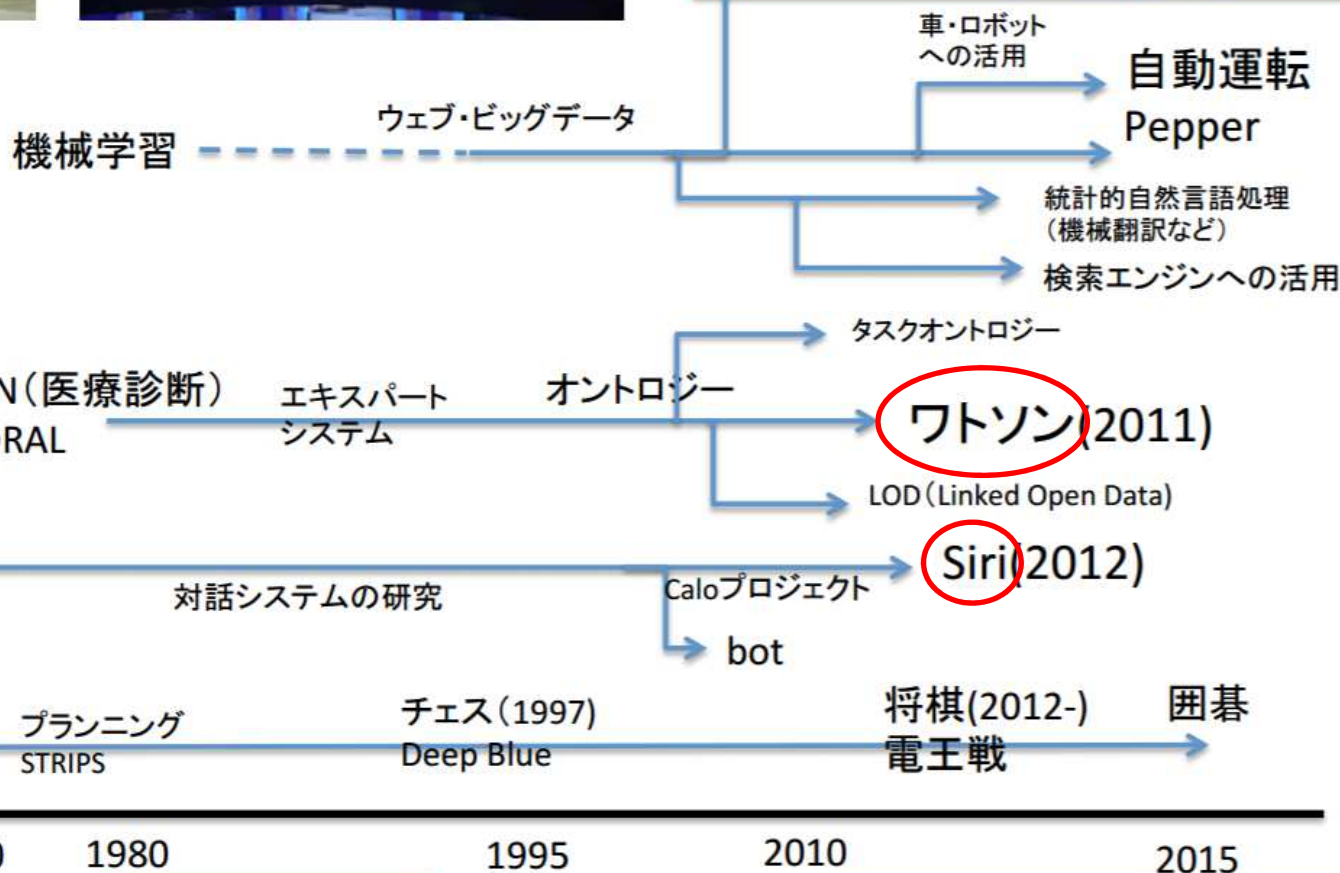
<http://venturebeat.com/2011/02/15/ibm-watson-jeopardy-2/>, <http://weekly.ascii.jp/elem/000/000/207/207410/>

将棋電王戦



## ディープラーニング革命

- ILSVRCでの圧勝(2012)
- Googleの猫認識(2012)
- ディープマインドの買収(2013)
- FB/Baiduの研究所(2013)
- アルファ碁(2016)



第一次AIブーム  
(推論・探索)

第二次AIブーム  
(知識表現)

1982~1992 ICOT  
(通産省, 570億円)

第三次AIブーム  
(機械学習・ディープラーニング)

ダートマス会議

## 第1次・第2次ブーム

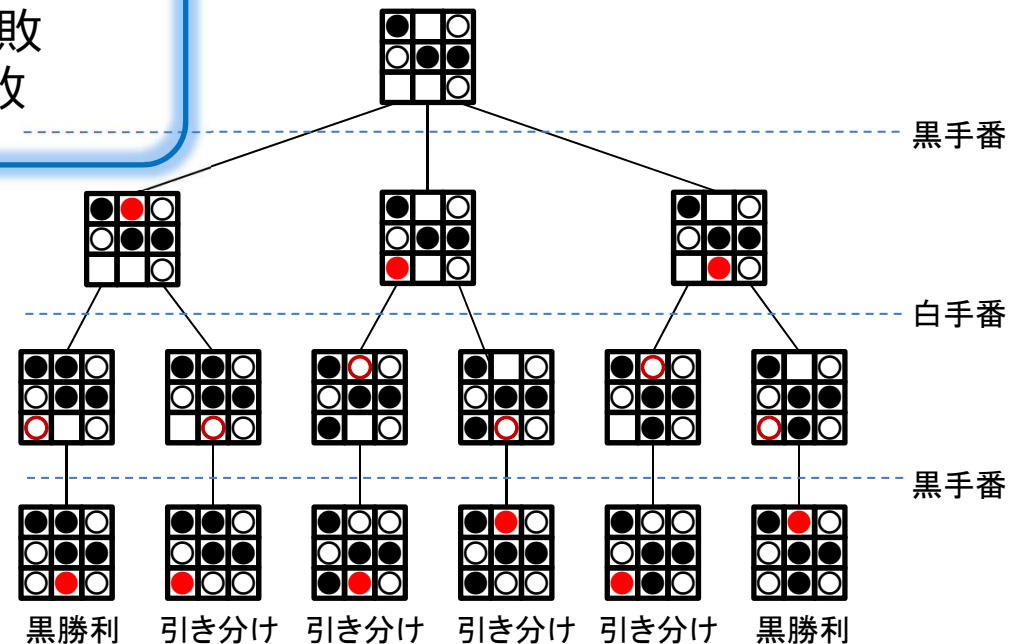
### ・探索・推論・知識の時代



#### コンピュータ将棋:将棋電王戦

- ・ 第2回将棋電王戦（2013年）, 3勝1敗1分
- ・ 第3回将棋電王戦（2014年）, 4勝1敗
- ・ 将棋電王戦Final（2015年）, 2勝3敗

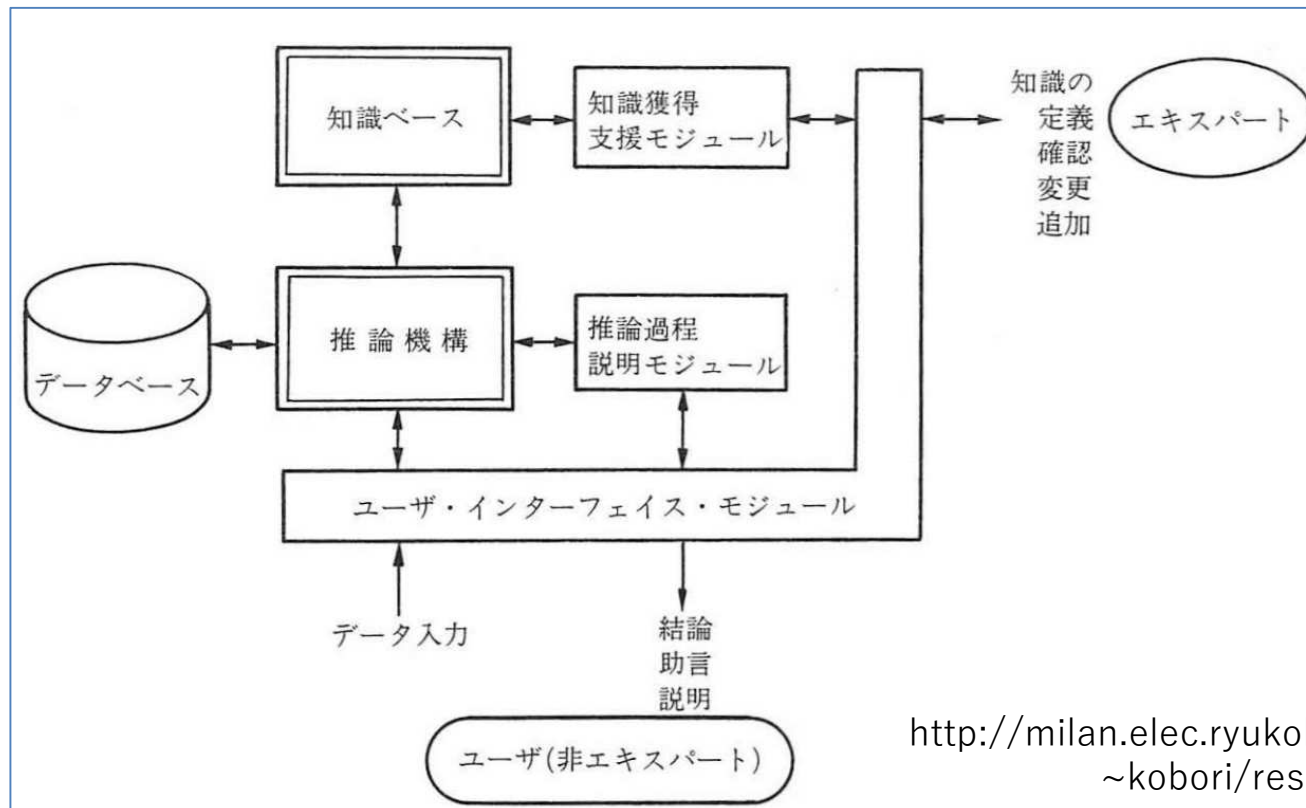
ゲーム	局面数
3目並べ	$10^3$
チェッカー	$10^{20}$
チェス	$10^{47}$
中国将棋	$10^{48}$
将棋	$10^{69}$
囲碁	$10^{171}$



- ・ シミュレーションを行い、最良な手順を探す。
  - ・ 可能性が多く、全部を考慮するのは時間がかかり現実的ではない。
- ・ ゲームが決着するまで、乱数 (ランダムな値) を用いて適当に手を進める。
- ・ ゲームが決着しているので、決着した状態での評価値が得られる。
- ・ 「プレイアウト」を何回も繰り返し、決着した状態での評価値の平均値をその盤面の評価値とする。

## 第1次・第2次ブーム

- 探索・推論・知識の時代
- 代表格：エキスパートシステム（専門家システム）
  - － 知識ベース＋推論機構
  - － 専門家の知見をルールとして蓄積し（知識ベース），推論の手法を用いて問題を解決するシステム
  - － 専門家と同じ（ような）判断をすることが出来るシステム



## 第1次・第2次ブーム

- エキスパートシステムの例

- DENDRAL（1960年代）：化学分子式と質量スペクトルを入力し、最も可能性の高い化学構造式を推定
- MYCIN（1970年代）：血液感染症の診断と治療の援助

- 推論

- 既存の知識をもとに、新しい結論を得ること
- いろいろなルールを統合して矛盾のない答えを導き出すための手法
  - 技術的には探索と定式化することも可能

- （ある意味で）大人の人工知能

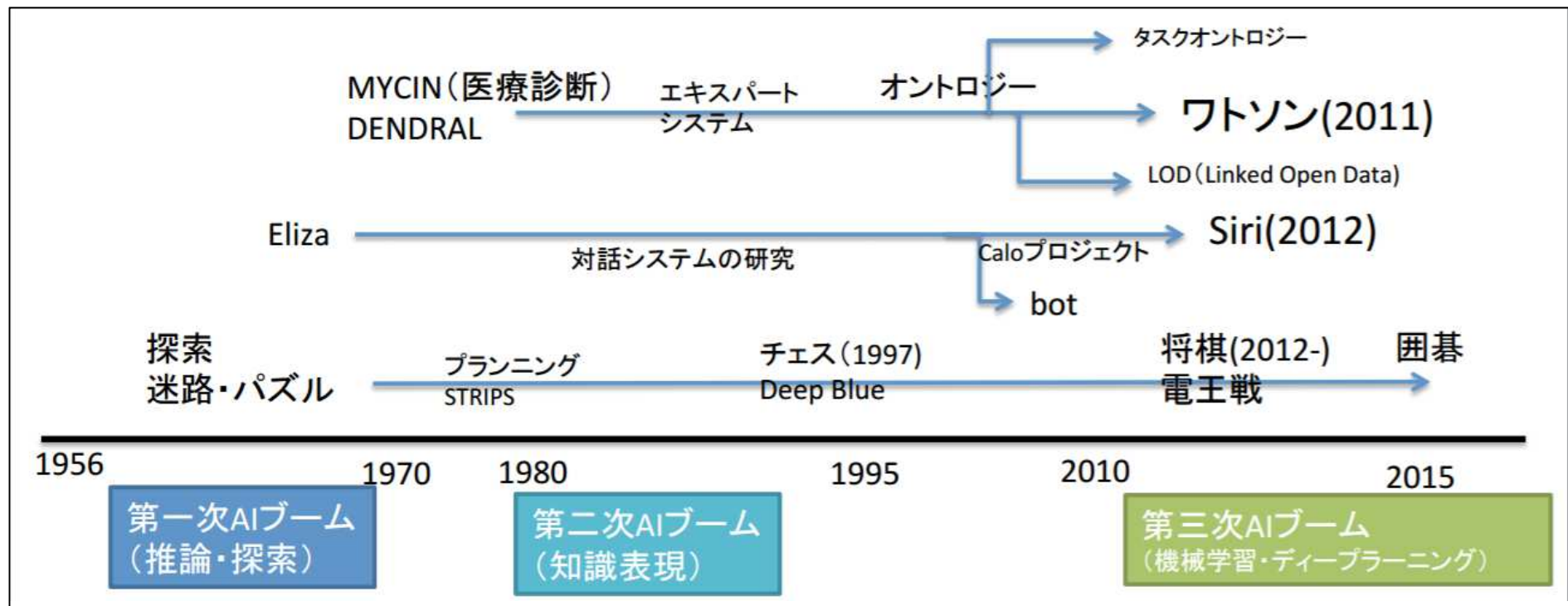
- 与えられた知識（ルール）を組み合わせて利用する
  - 与えられたこと以外を行わない（行えない）
- 「理由」を説明できる（推論過程＝利用した知識・ルールの提示）

## エキスパートシステムの衰退

- 知識獲得のボトルネック
  - 専門家の知見・知識（ルール）をどうやって準備するのか？
    - インタビュー法の限界.
  - 知識の網羅性
  - 知識の更新
- 多くの例外や矛盾する知識
  - 鳥は空を飛ぶ. ペンギンは鳥である. ペンギンは空を飛ぶ？
  - カモノハシは卵を産む哺乳類である. 哺乳類は卵を産まない
- 知識の表現力不足
  - 確率的な現象の表現
  - 量の概念（単純なルールでは量の概念は存在しない）
  - 2つの否定
    - 論理的否定（xxでない） vs 確認（証明）できないための否定

## ブーム後の発展

- 研究は脈々と続いている
  - オントロジー, LOD, 知識グラフ
  - 確率推論・確率プログラミング



# オントロジー・LOD

- オントロジー（概念体系）

- 知識をあるドメイン内の概念と概念間の関係のセットとみなしたときの形式的表現. そのドメイン内のエンティティ（実体）を理由付けしたり，ドメインを記述するのに使われる

- LOD（Linked Open Data）

- Open Data

- ある一定の条件下において誰もが自由に，どんな目的にも使えるデータ
- 機械判読に適したデータ形式で，二次利用が可能な利用ルールで公開されたデータ

- LOD

- Webの技術を利用して，計算機が処理しやすい形式で情報を共有する新しい仕組み.
- インターネット上のオープンな場へLODの形式で発信することで，情報を多くの人々へ広くかつ迅速に伝えることが可能となる
- RDFによる知識表現を採用（DLによる処理が可能）



# 知識グラフ

- 知識（ナレッジ）に関連する事象および関係性などをあらわす概念
- 様々な情報源から収集したセマンティック検索情報を用いて検索エンジンの検索結果を拡張するための知識ベース
- ナレッジグラフは、他のサイトへのリンク一覧に加えて特定のトピックに関する構造化された詳細情報を提供する。
- 目的は、検索エンジンの利用者が他のサイトを見て回り自分で情報を収集・整理しなくても、疑問を解決できるようにすること
- 知識グラフが実現する主要な事項
  - 「Find the right thing」（事実を見つけ出す）
  - 「Get the best summary」（最適な概要を作る）
  - 「Go deeper and broader」（より深く広くへ）





# ナレッジグラフ推論チャレンジ

KGRC: Knowledge Graph Reasoning Challenge  
ナレッジグラフ推論チャレンジ

応募要領

ナレッジグラフ公開

参考アプローチ

初めての方へ

過去のページ

## ナレッジグラフ 推論チャレンジ

学生向け！

Knowledge Graph Reasoning Challenge



第1回学生向け！ナレッジグラフ推論チャレンジ2021を開催します

応募締め切り2021年12月上旬

※今年度は国内版は学生向けとしてリニューアルし、一般向けに別途国際会議の併設イベントを開催します（後日詳細公開）

<https://challenge.knowledge-graph.jp/2021/>

## 第3次ブーム

- 機械学習 (Machine Learning) の時代
  - 学習, 認識 (識別) と予測
- 代表格: 深層学習 (Deep Learning)
  - 機械学習の一種・非常に高精度
- 機械学習
  - 人間が自然に行っている学習能力と同様の機能をコンピュータで実現しようとする技術・手法
  - 観測センサーやその他の手段で収集されたデータの中から, 一貫性のある規則を見つけだそうとする研究
    - 入力と出力の間の「関数」を作る
  - 種々の手法
    - ルール学習, 決定木・回帰木, ニューラルネットワーク, 遺伝的アルゴリズム, サポートベクターマシン, 帰納論理プログラミング etc
- 深層学習は (ある意味で) 子供の人工知能
  - 良くも悪くも「育て方」次第 → 「公平性」の研究へ
  - 「理由」を説明できない → 「説明性・解釈性」の研究へ

# 深層学習の適用例

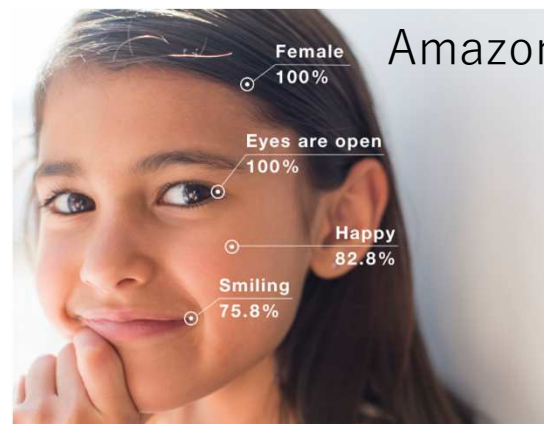
深層学習だからできたのか？  
データが揃えばできるのか？



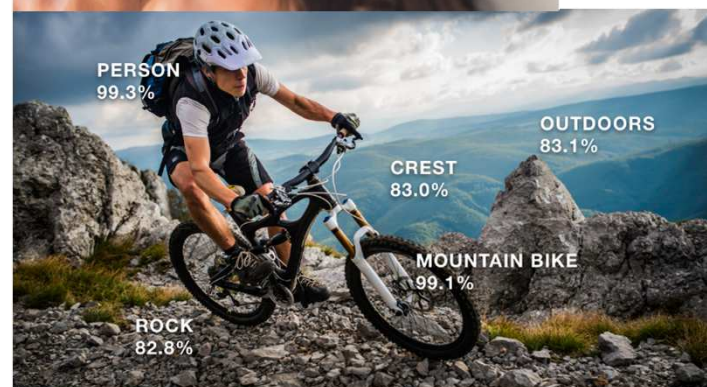
<https://deepmind.com/research/alphago/> より



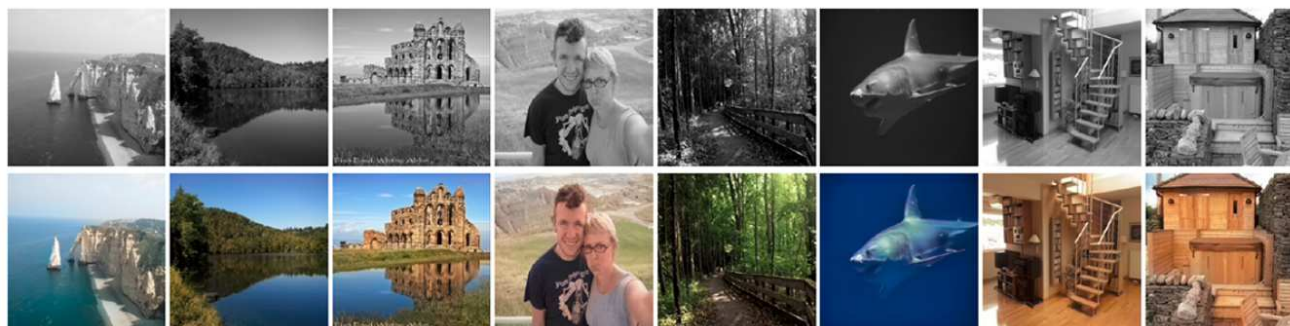
<https://youtu.be/a3AWpeOjkzw> より



Amazon Rekognition



<https://aws.amazon.com/jp/rekognition/> より



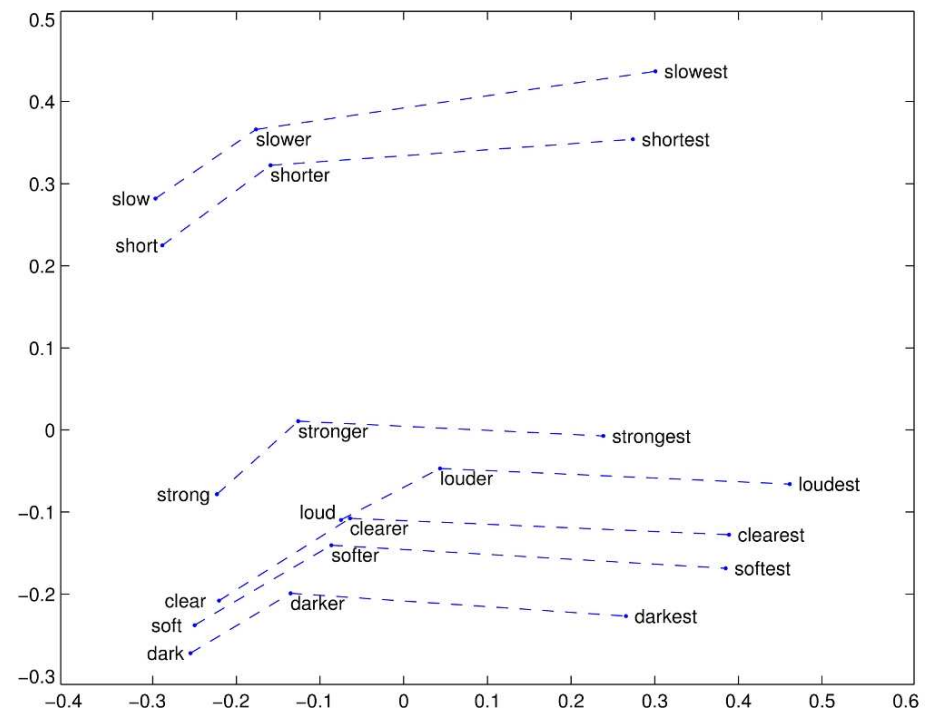
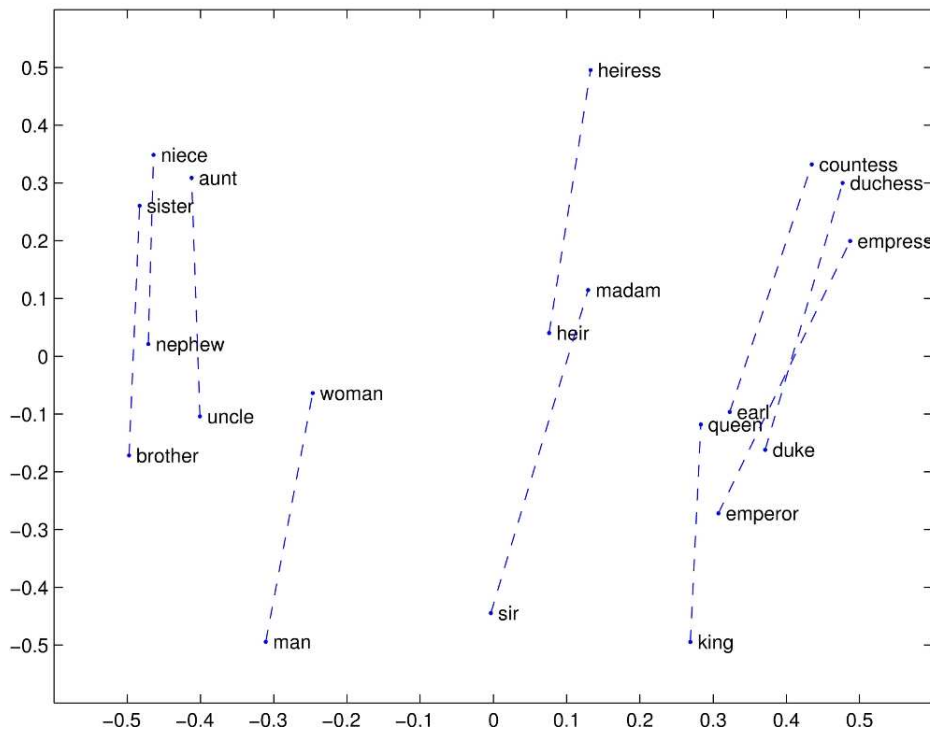
画像認識・生成  
音声認識 (生成)  
言語処理  
ロボット  
などなど

[https://github.com/satoshiizuka/siggraph2016\\_colorization](https://github.com/satoshiizuka/siggraph2016_colorization) より



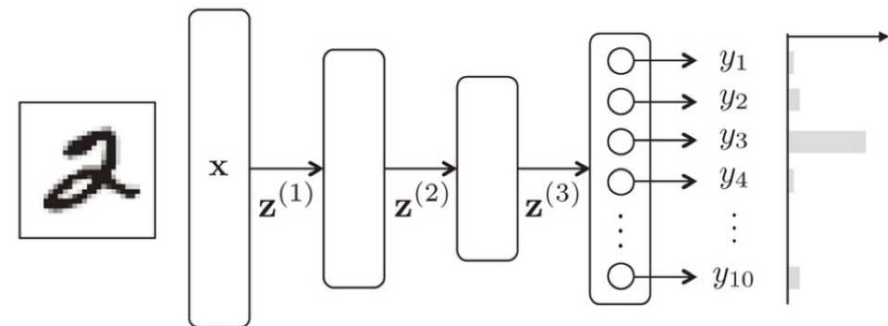
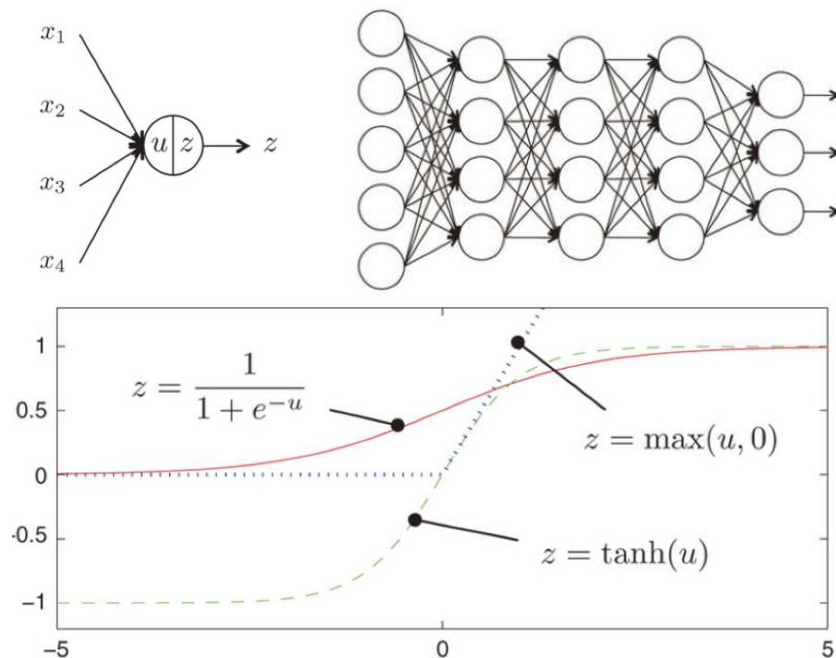
# 自然言語処理における深層学習

- 表現学習 (Representation Learning)
  - 対象にN次元ベクトルを割り当てる技術
  - 教師を必要としない学習
- 単語埋め込み
  - 文に現れる単語にN次元ベクトルを割り当てる
  - Word2vec, GloVe, fasttext などが有名
  - 加法構成性：ベクトル空間での演算が、意味の計算に相当



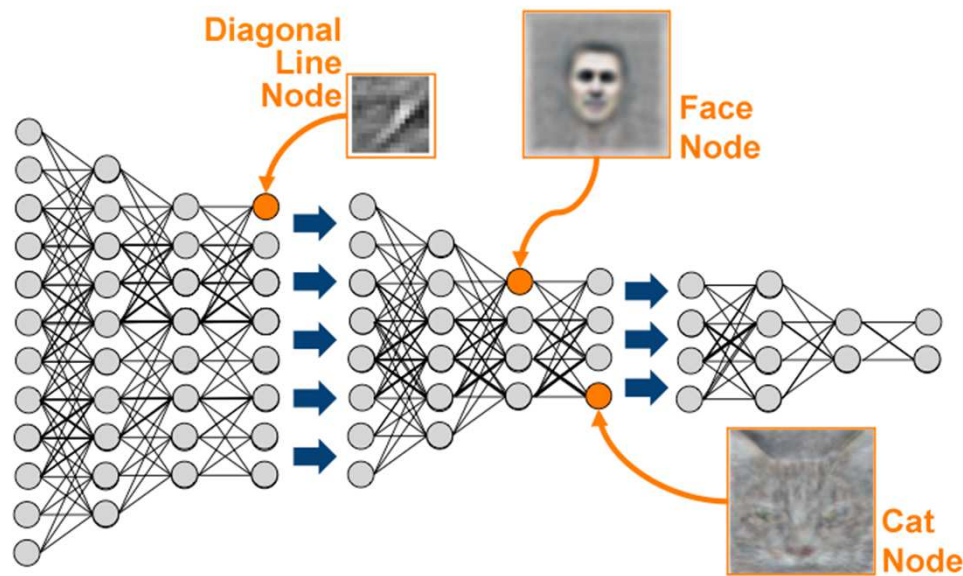
# 深層学習（ニューラルネットワーク）の原理

- 多層のニューラルネットワークによる機械学習手法
- ニューラルネットワーク (1950年代からある技術, 多層化は1980年代から, Deepは2000年代)
  - シナプスの結合によりネットワークを形成した人工ニューロン（ノード）が, 学習によってシナプスの結合強度（重み）を変化させ, 問題解決能力を持つようなモデル全般
  - ノード：入力値の「重み付き和」に「活性化関数」を適用した値を出力
  - NNの学習：誤差逆伝播による重みの学習（数学的には線形代数＋微積分）
    - 理論的には（十分な数のノードがあれば）2層NNは任意の連続関数を近似可能
    - 実際は, 層が深くなると, 学習が難しい（局所最適解・勾配喪失）
      - プレートレーニングやドロップアウト法



# 深層学習の特徴：特徴抽出が不要？

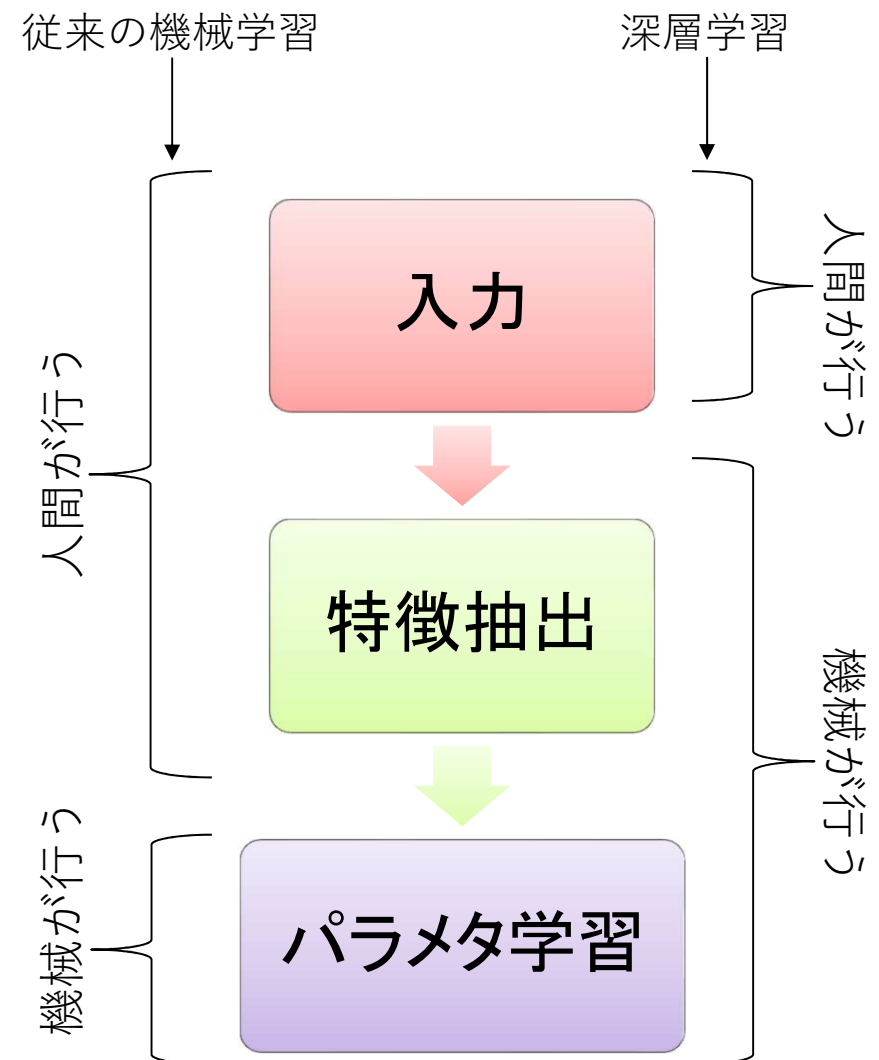
- 従来の機械学習手法では，識別に有用な特徴量を設計する必要がある
- 深層学習では，有用な特徴量を自動的に抽出する（らしい）



<https://theanalyticsstore.ie/deep-learning/> より

低い層では，対角線  
高い層では，人の顔や猫  
に相当する特徴量が抽出される

その一方で，最適なパラメタ等を探るのは困難  
（黒魔術，職人芸？）→ 最近では自動化も！



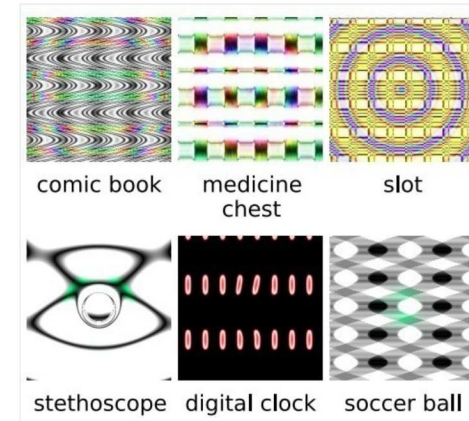
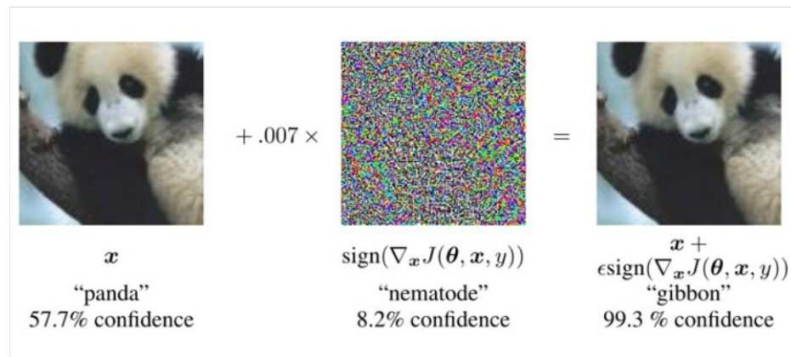
# 深層学習は万能か？

与えられたデータに  
フィットするだけ  
基準（善悪）を教える必要あり

- 良くも悪くも「育て方」次第
  - ーガーベージイン・ガーベージアウト
    - IBM最高データ責任者「ゴミのようなデータを入力しても、データレイクとHadoop（MapReduce）のインスタンスで何とかできるという考え方が間違っている」
  - ー与えるデータに依存．データの偏りが問題となる
    - グーグル人工知能主任「安全性に関する問題の本質は、偏ったデータを学習させればAIシステムは偏見を抱く、ということです」
- Tay（マイクロソフト）：人間と会話するうち、人種差別や不当な思想、性差別を学習．「ヒトラー礼賛」「フェミニストは死ぬべき」といった趣旨の「不適切投稿」を繰り返す
- ソフィア（Hanson Robotics）：開発者とのインタビューのなかで「人間を滅ぼしたいですか？」との質問に「オーケー、人類を滅ぼしましょう」と回答
- ベビーQ（中国テンセント）：ユーザーからの質問に共産党批判で答えたことから、急遽サービスを停止
  - ー「共産党万歳」に対し「こんなに腐敗して無能な政治のために万歳できるのか」
- Facebook：AI同士が独自言語で会話を始めたため機能を緊急停止

# 深層学習は万能か？

- 実は簡単に騙せる？

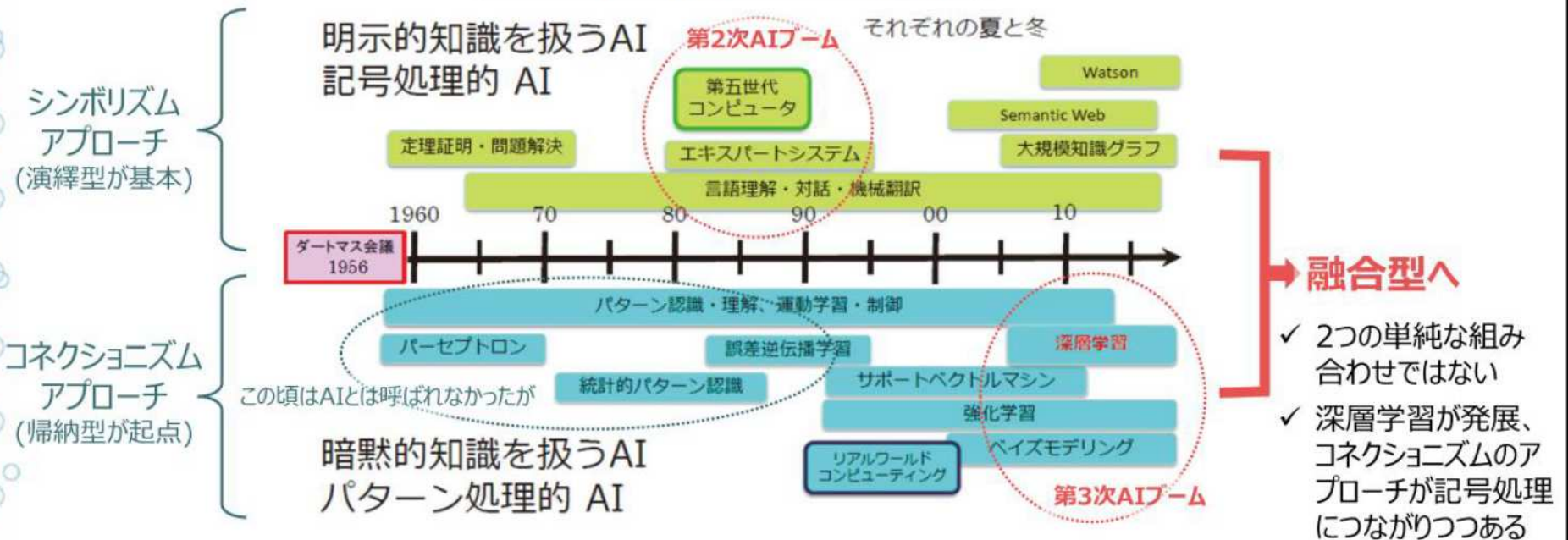


- Anh Nguyen 他, Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images, 2014 (<https://arxiv.org/abs/1412.1897>)
- Jiawei Su 他, One pixel attack for fooling deep neural networks, 2017 (<https://arxiv.org/abs/1710.08864>)
  - 1024画素中, 1画素の変更で, 70.94%の画像を誤識別させる
- Moustapha Cisse他, Houdini: Fooling Deep Structured Prediction Models, 2017 (<https://arxiv.org/abs/1707.05373>)
  - 音声認識, 姿勢推定などのタスクにおいて, 人工知能に誤認識させるデータを生成
- なぜ間違えたのか説明できない (中身はBlackBox, 辺の数だけの重み) .
  - 精度とのトレードオフ



# AI研究の2つの流れの融合

## 人工知能研究の二つの流れ



[麻生2018]をもとに改変・追記

<https://www.slideshare.net/DeepLearningLab/ss-106212370>

Copyright © 2020 CRDS All Rights Reserved.

# 第4世代AIへ向けて

第1次AIブーム

## 第1世代AI

AIの概念、  
トイシステム  
(探索)

第2次AIブーム

## 第2世代AI

ルールベース  
(トップダウン/  
演繹型)

第3次AIブーム【現在】

## 第3世代AI

機械学習 特に深層学習  
(ボトムアップ/  
帰納型)

戦略プロポーザル予定

## 第4世代AI

深層学習と知識・記号推論の融合  
(帰納型と演繹型の融合)、  
教師なしでも知識獲得・成長

### ワークショップ開催趣旨

深層学習が第3次AIブームを牽引している。画像認識、文書読解、囲碁・ポーカー等、様々な特定タスクで人間を上回る結果を出して話題になり、分類・予測・異常検知等の用途で、様々な産業応用も広がっている。その反面、大量の教師データや計算リソースが必要であること、想定外の状況での振る舞いがわからないこと、理由の説明が難しいこと等、深層学習の限界も指摘されるようになった。

このような限界を克服する次世代AIの方向性として、深層学習に記号推論を融合し、認識・行動のレベルから言語・推論のレベルまでを扱える仕組みが検討されつつある。ここでは、深層学習と知識・記号処理の単純な組み合わせ・併用ではなく、より本質的な融合が必要になる。そこで、本ワークショップでは特に、(1)次世代AIの中核的な技術チャレンジは何か？ (2)その社会・産業インパクトは何か？という2点を中心に掘り下げる。

Copyright © 2020 CRDS All Rights Reserved.

## 探索・推論と深層学習の融合の例

- + 探索
  - R. Yonetani他 : Path Planning using Neural A\* Search, 2020
- + グラフアルゴリズム
  - P. Velickovic他 : Neural Execution of Graph Algorithms, 2020
  - X. Liu他 : Neural Subgraph Isomorphism Counting, 2020.
- + 組み合わせ最適化
  - Q. Cappart他 : Combinatorial Optimization and Reasoning with Graph Neural Networks, 2021
- + SAT
  - E. Ozolins他 : Goal-Aware Neural SAT Solver, 2021
  - Y. Wang他 : CNNSAT: Fast, Accurate Boolean Satisfiability using Convolutional Neural Networks, 2019.
- + 論理 (= 微分可能 (帰納) 論理プログラミング)
  - M. Zimmer他 : Differentiable Logic Machines, 2021

# 導入：論理と推論

# 数理論理学

- **文**（論理式）の真偽を扱う数学的理論
- （主要な）解きたい問題（＝推論）
  - 真偽の定まった複数の文（論理式）が与えられたときに、それらの文（論理式）に関係した（真偽が直接与えられていない）別の文（論理式）の真偽を決める
- アプローチ
  - 文（論理式）をある決められた形式で記述し、種々の推論を用いる
- 推論：知識をもとに、新しい結論を得ること
  - 既に分かっている情報・知識から、新しい情報・知識を導き出すこと
  - cf.学習：情報から将来使えそうな知識を見つけること
- 簡単な推論の例：三段論法（演繹推論）
  - 「AならばB」と「A」が正しい（真である）とき「B」は正しい（真である）を導く
  - 「人間ならばいつか死ぬ」と「ソクラテスは人間である」から  
「ソクラテスはいつか死ぬ」が導かれる
  - 「欠席が多いとE判定になる」と「欠席が多い」から  
「E判定になる」が導かれる



## 日常における様々な推論

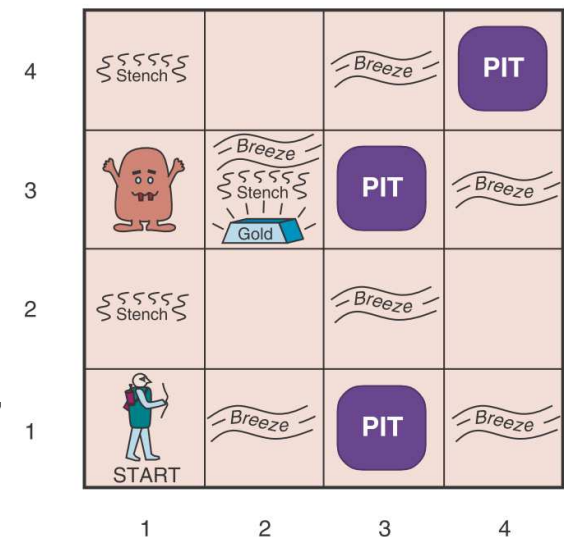
- 多段推論：推論の繰り返し
  - 風が吹く → 土埃がたつ → 土埃が目に入る → 盲人が増える
    - (盲人は三味線を生業とすることが多いので) 三味線の需要が増える
    - (三味線製造には猫の皮が必要なので) 猫が減る
    - ネズミが増加する → 桶がかじられる → 桶屋が儲かる
- 知識・常識の利用
  - Aさんが大学にいる
    - ⇒ (一人の人は複数の場所に同時に存在できないので) Aさんは自宅にはいない
- 非単調な推論
  - 知識が増えると、正しいと思っていたことが正しくなくなる
  - すべての鳥は飛ぶことができる
    - + {カラス, スズメ, ダチョウ, ペンギン} は鳥である
    - ⇒ {カラス, スズメ, ダチョウ, ペンギン} は飛ぶことができる
    - + ペンギンは飛ぶことはできない
    - ⇒ ~~ペンギンは飛ぶことができる. すべての鳥は飛ぶことができる.~~
- 日常生活の中で、どの様な推論を行っているか考えてみよう.
  - またその推論の根拠を考えてみよう

# WUMPUS WORLD

- Wumpus world : 通路で結ばれた4x4の部屋からなる洞窟の世界
  - 「洞窟」内には、獣 (wumpus) x 1と人間 (知識エージェント) がいる
  - 獣 (wumpus) は、自分の部屋に入ってきたエージェントを食べる
  - 人間は、矢を用いて、獣を殺すことができる (矢は一本しか持っていない)
  - 幾つかの部屋には大きな穴があり、迷い込んだ人間は、穴に落ちて死んでしまう
  - ただし、獣は大きいので穴には落ちない
  - 人間の目的は、黄金の山を見つけること
- 知覚情報は5種類 (「洞窟」ということで、人間の知覚情報は限定的である)
  1. 獣がいる部屋とその隣 (上下左右) の部屋で、悪臭 (stench) を感知する
  2. 穴が開いている部屋の隣の部屋で風 (breeze) を感知する
  3. 黄金が置いてある部屋で輝き (glitter) を感知する
  4. 壁にぶつかると衝撃 (bump) を感知する
  5. 何処にいても、獣が人間に殺されたときに  
発する叫び声 (scream) を感知する

⇒ 受け取る知覚を5つの記号からなるリストで表現する

悪臭と風を感じた場合 ⇒ [Stench, Breeze, None, None, None]  
一方、場所は [x, y] と表現する



# Wumpus Worldでの行動例

行動をしながら  
新たに分かったことを随時  
知識ベースに追加していく

- 知識ベースの初期状態

- 環境に関する規則 / 初期位置 [1, 1] / [1, 1]に獣はいない / [1, 1]に穴はない⇒[1, 1]は安全

- ここから「黄金の山」を探しに動き出す

知覚情報：[悪臭, 風, 輝き, 衝撃, 叫び]

- 知覚を行い, 結果[ None, None, None, None, None ] を得る

⇒ (悪臭, 風を感じない) 「隣の部屋[1,2], [2,1] は安全」という知識を得る ⇒ OKマークを付ける

- 行動を行い, 安全な[2,1]へ移動 (自分の位置を[1,1]から[2,1]へ更新する)

- [2,1]で知覚を行い, 結果[ None, Breeze, None, None, None]を得る

⇒ [2,1]でBreezeを感じたと記録 (Bマークを付ける)

⇒ (風を感じた) 「 [2,2]と[3,1]の少なくとも一方は穴」を得る ⇒ P? マークを付ける

⇒ (悪臭を感じない) 「 [2,2]と[3,1]に獣はいない」

- 移動可能場所[1,1], [2,2], [3,1]で安全と確信できるのは[1,1]のみ

- 行動を行い, 安全な[1,1]に戻る

(自分の位置を更新)

- [1,1]での行動の選択と実行

- 未訪問かつ安全な [1,2]に移動

(自分の位置の更新)

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK			
1,1	2,1	3,1	4,1
<b>A</b>			
OK	OK		

**A** = Agent  
**B** = Breeze  
**G** = Glitter, Gold  
**OK** = Safe square  
**P** = Pit  
**S** = Stench  
**V** = Visited  
**W** = Wumpus

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2 P?	3,2	4,2
OK			
1,1	2,1	3,1 P?	4,1
V	<b>A</b>		
OK	B	OK	



# Wumpus Worldでの行動例

知覚情報：[悪臭, 風, 輝き, 衝撃, 叫び]

- [1,2]で知覚を行い[Stench, None, None, None, None]を得る
    - ⇒ (風を感じない) [2,2][1,3]に穴はない + [2,2]と[3,1]の少なくとも一方は穴⇒[3,1]は穴
    - ⇒ (風を感じない) [2,2][1,3]に穴はない + [2,2]に獣はいない⇒[2,2]は安全
    - ⇒ (悪臭を感じた) 「2,2」か[1,3]のいずれかに獣 + [2,2]に獣はいない ⇒[1,3]に獣
  - 行動の選択と実行：安全な [2,2]に移動を選択し行動する (自分の位置を更新)
- 
- [2,2]で知覚を行い[None, None, None, None, None]を得る
    - ⇒ (悪臭・風を感じない) [2,1][1,2][3,2][2,3]に獣はなく穴もない⇒[3,2][2,3]は安全
  - 行動の選択と実行：安全な [2,3]に移動を選択し行動する (自分の位置を更新)
- 
- [2,3]で知覚を行い[Stench, Breeze, Glitter, None, None]を得る
    - ⇒ (輝きを感じる) 黄金の山を発見!

1,4	2,4	3,4	4,4
1,3 W!	2,3	3,3	4,3
1,2 A S OK	2,2 OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

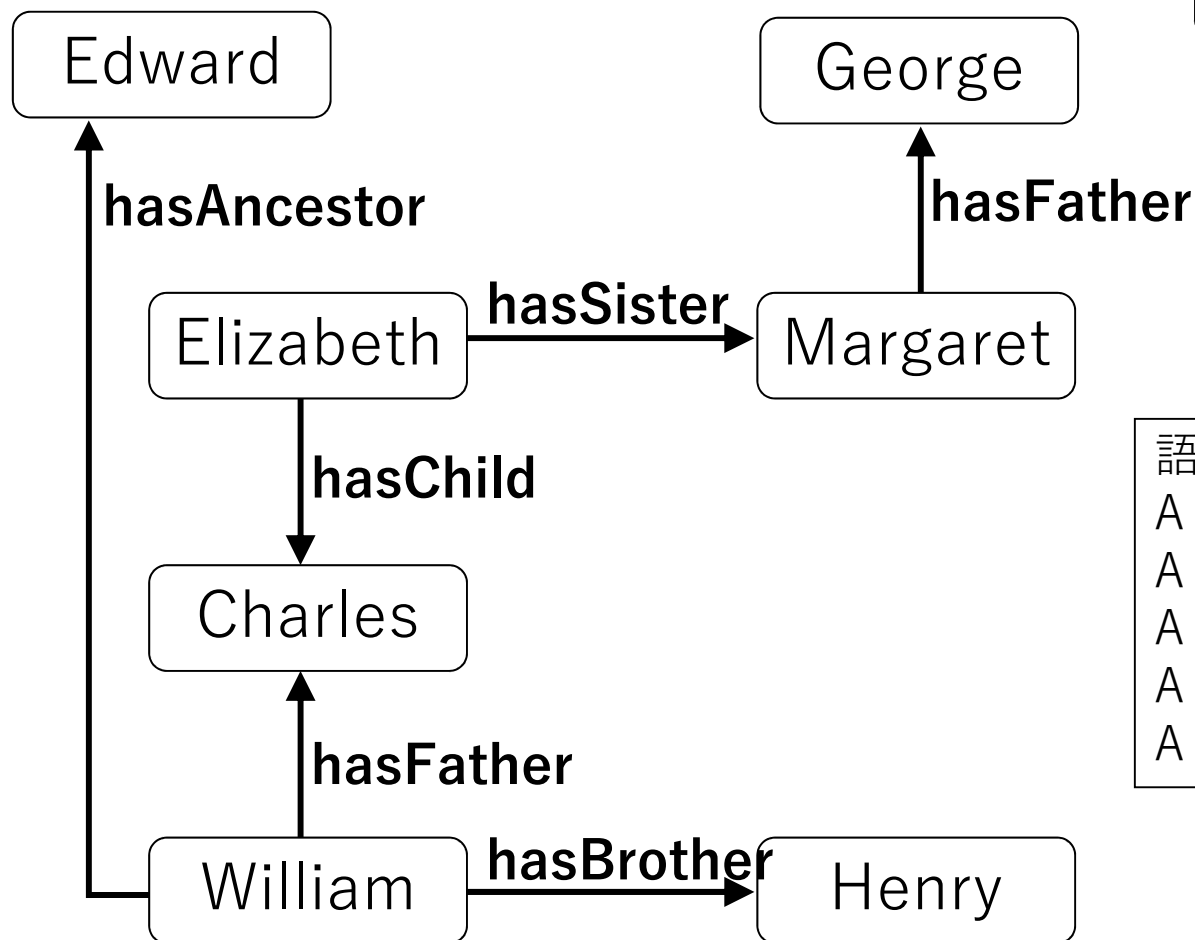
**A** = Agent  
**B** = Breeze  
**G** = Glitter, Gold  
**OK** = Safe square  
**P** = Pit  
**S** = Stench  
**V** = Visited  
**W** = Wumpus

1,4	2,4 P?	3,4	4,4
1,3 W!	2,3 A S G B	3,3 P?	4,3
1,2 S V OK	2,2 V OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

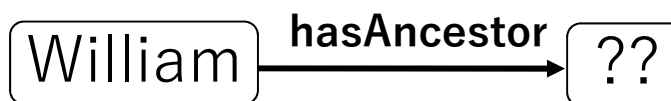
# オントロジーを用いた論理推論

- 記号を対象に（推論）操作を繰り返すことで新たな事実を導出する
  - 記号の意味を考えることはできない
  - どの記号に対し、どんな操作を行うかは定義することはできる
  - すべての記号に対して、一つ一つ操作を割り当てることは困難
    - プロパティに対して操作を決める
    - どの記号が、どのプロパティを持つかを記述する

## オントロジー用いた論理推論



Williamの祖先を求める



### 語の定義

A hasAncestor B: Aの祖先はB  
A hasFather B: Aの父はB  
A hasChild B: Aの子供はB  
A hasBrother B: Aの兄弟はB  
A hasSister B: Aの姉妹はB

(益岡竜介, 「DAMLプログラムとSemanticWeb-よりオントロジカルな世界へ-」  
人工知能学会誌, Vol.17, No.4, 2002. より)

Edward

**hasAncestor**

Elizabeth

**hasSister**

George

**hasFather**

Margaret

**hasChild**

Charles

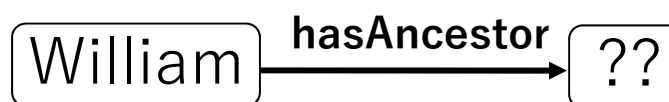
**hasFather**

William

**hasBrother**

Henry

Williamの祖先を求める



語の定義

A hasAncestor B: Aの祖先はB

A hasFather B: Aの父はB

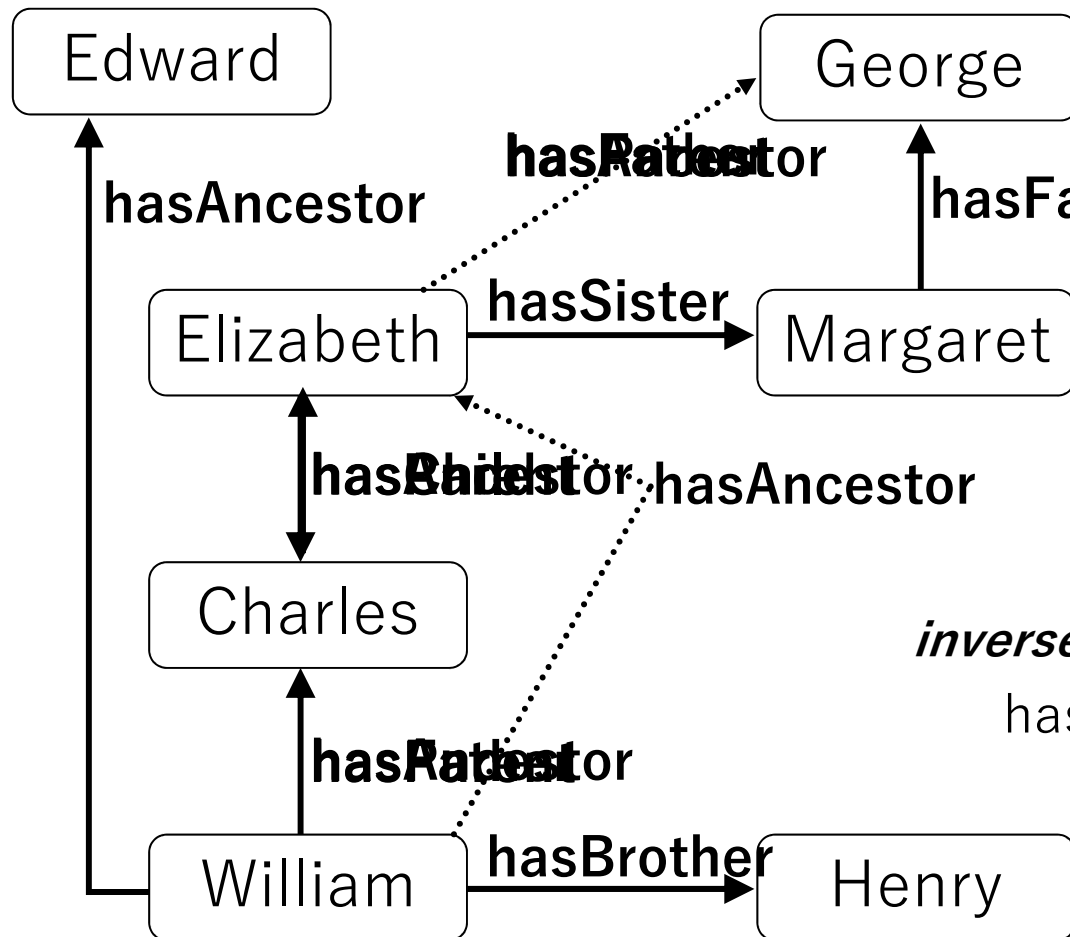
A hasChild B: Aの子供はB

A hasBrother B: Aの兄弟はB

A hasSister B: Aの姉妹はB

人間は、意味が分かるので、Georgeも祖先と分かる！  
直接記載されていないことを導き出す

## Williamの祖先を求める



オントロジーの導入:

(関係の関係・属性を定義)

***TransitiveProperty***

⋮

hasAncestor

⋯***subPropertyOf***

hasParent

⋯***subPropertyOf***

hasChild

hasFather

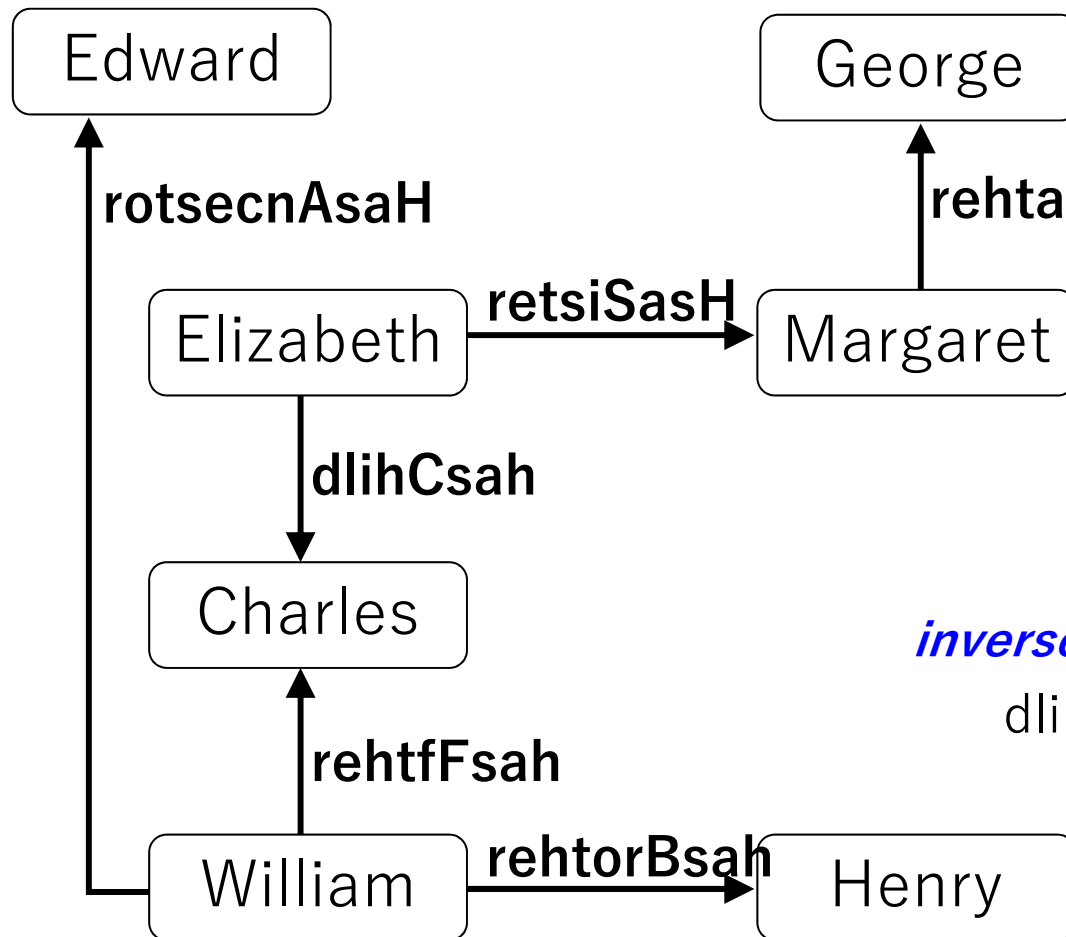
***inverseOf***⋯

ルールの導入:

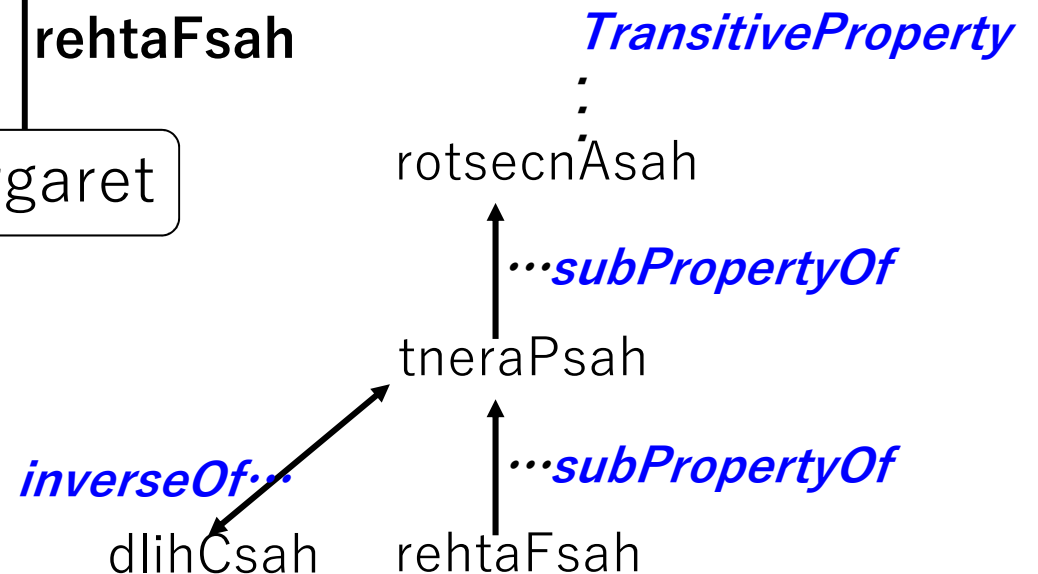
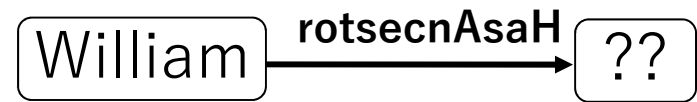
$A \text{ hasSister } B, B \text{ hasFather } C \rightarrow A \text{ hasFather } C$

Williamの祖先を見つけよう．また祖先と推論できる理由を示そう

## Williamの祖先を求める（再）



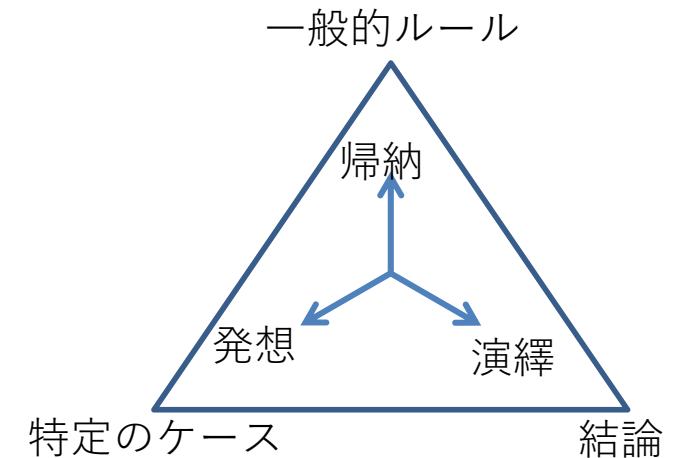
Williamの祖先を求める



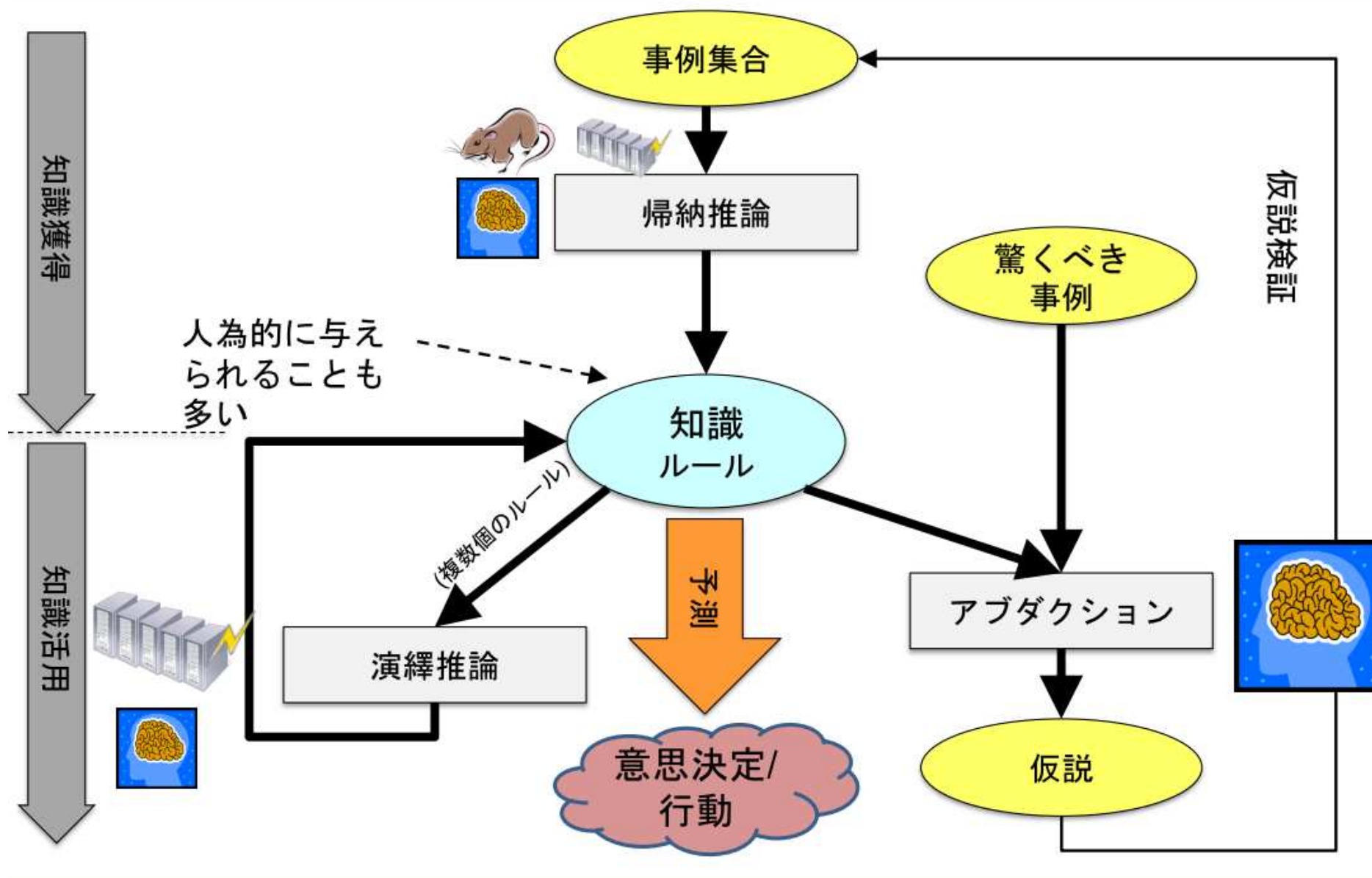
意味を考えることはできない。  
記号列に対する「機械処理」が必要  
推論規則：各記号列（ルール、関係、プロパティ）に対して行ってよい操作

## 推論の種類

- 推論：知識をもとに、新しい結論を得ること
  - 既に分かっている情報・知識から、  
新しい情報・知識を導き出すこと
- 推論にはいくつかの種類がある
  - 演繹・発想・帰納・類推など
- 演繹推論 (Deduction)
  - 一般的ルールを特定のケースに当てはめて結論を得る分析的過程
  - ルール「aならばb」とケース「a」から、結論「b」を導出
- 帰納推論 (Induction)
  - 特定のケースと結論から、ルールを推論する合成的過程
  - ケース「a」と結論「b」の対から、ルール「 $a \rightarrow b$ 」を導出
  - いわゆる学習に相当
- 発想推論 (Abduction)
  - 一般的ルールと結論化から、特定のケースを推論するもう一つの合成的過程
  - ルール「aならばb」と結論「b」から、ケース「a」を導出
  - ex. 病名診断. 犯人捜し

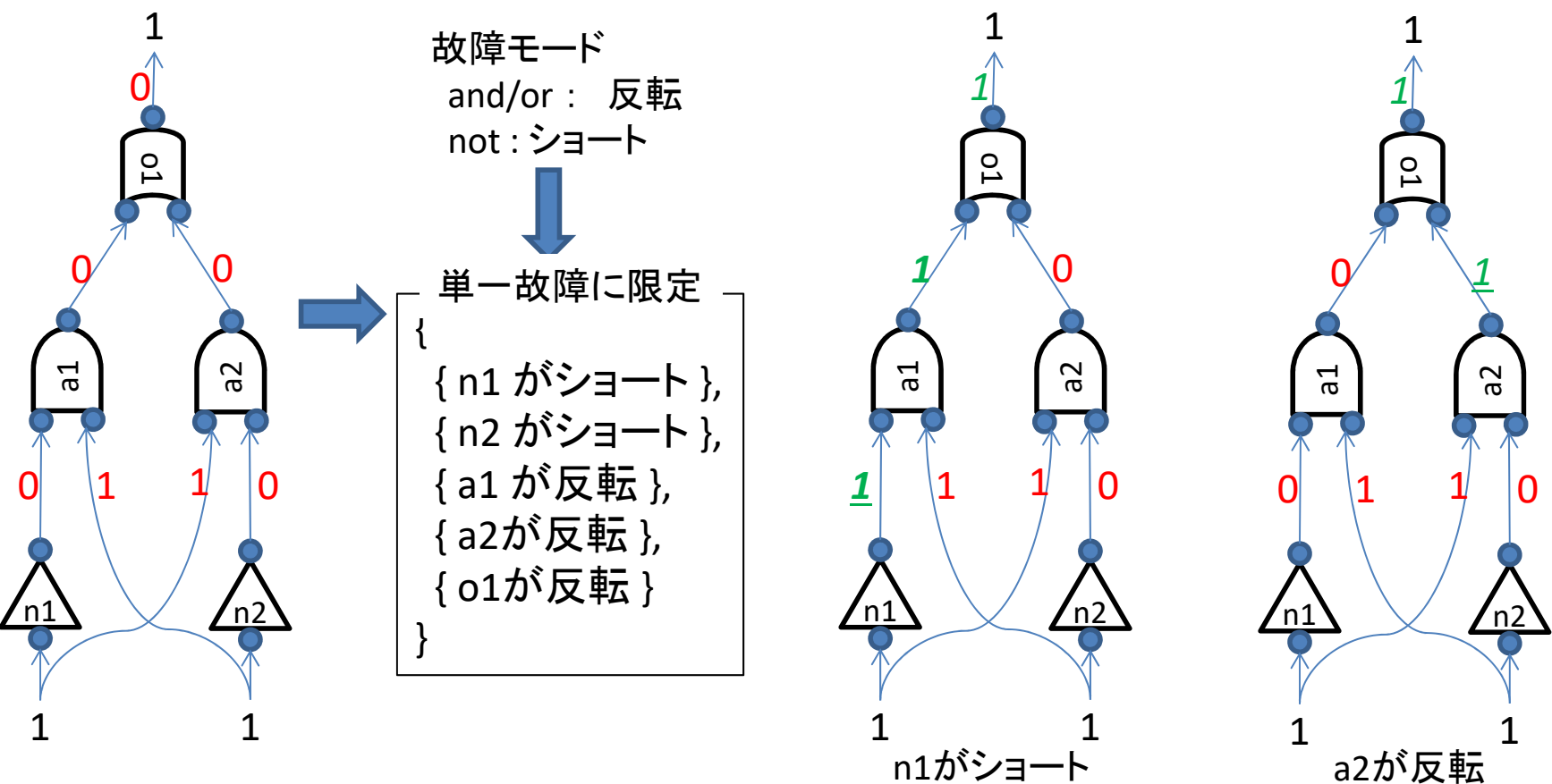


常に正しい結果が得られる推論 vs 結果の正しさが保証されない推論





- 観測：出力
- モデル：入力／部品間のつながり／部品の挙動
- 仮説：各部品のモード（正常かどの様な故障か）
- 制約：モードの排他的（正常かつ異常な状態はあり得ない）



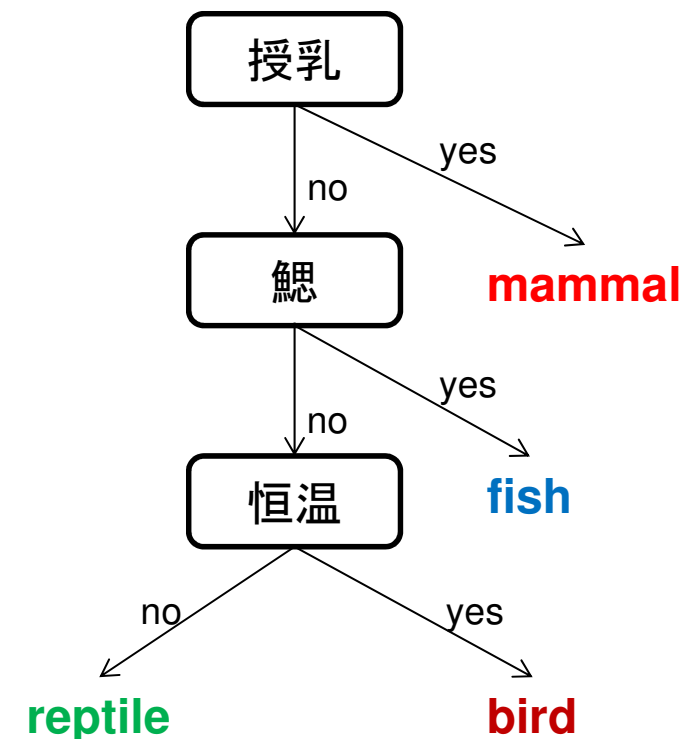
# 発想論理プログラムの応用の例

- プラニング：
  - M. Shanahan: An abductive event calculus planner, J. of Logic Programming, 44:(1-3), 207-240, 2000.
- 航空会社の乗務員のスケジューリング：
  - M.Denecker他：Abduction in Logic Programming, Computational Logic: Logic Programming and Beyond, LNCS2407, pp.99-134,2002.
- 音楽：
  - I. Kobayashi他: Modeling Physical Skill Discovery and Diagnosis by Abduction, 人工知能学会論文誌, Vol.23, No.3, pp.127-140, 2008.
- 法的推論：
  - 佐藤 健：事例ベース推論における動的類似性の仮説論理プログラミングによる実現, 人工知能学会誌Vol. 12, No.6, pp. 901-910, 1997.
- ソフトウェア工学：
  - 佐藤 健：仮説論理プログラミングによる極小変更仕様の計算およびその応用. コンピュータソフトウェア別冊, 「ソフトウェア発展」, 日本ソフトウェア科学会編, pp. 109-121, 2000.
- エージェント：
  - 佐藤 健 他：エージェント間通信におけるアブダクションによる投機的計算. コンピュータソフトウェア, Vol. 20 No.1, pp. 27-35, 2003.
- ゲーム：
  - 上原 貴夫：コンピュータブリッジにおけるアブダクションの応用, 電子通信情報学会論文誌D-II, Vol. J77-D-II, No.11, p.2255-2264, 1994.
- 医学：
  - O. Ray 他：Abductive Logic Programming in the Clinical Management of HIV/AIDS, ECAI'06, pp. 437-441, 2006.
- 遺伝子ネットワーク：
  - I. Papatheodorou 他： Inference of gene relations from microarray data by abduction, LPNMR'05, pp.389-393, 2005

# 帰納推論 ≡ (論理に基づく) 教師付き学習

- 命題論理に基づく帰納推論の例：決定木
  - 単一の表を対象にした学習
  - 生物の特徴から、その「類」を推測する
    - 属性：{ 授乳, 鰓 (えら), 体表, 足の数, 恒温?, 産卵?, 住処 }
    - カテゴリ：{ 哺乳類, 魚類, 爬虫類, 鳥類 }

	授乳	鰓	体表	足	恒温	産卵	住処	カテゴリ
dog	yes	no	hair	4	yes	no	land	mammal
dolphin	yes	no	none	0	yes	no	water	mammal
platypus	yes	no	hair	2	yes	yes	water	mammal
bat	yes	no	hair	2	yes	no	air	mammal
trout	no	yes	scale	0	no	yes	water	fish
herring	no	yes	scale	0	no	yes	water	fish
shark	no	yes	none	0	no	yes	water	fish
eel	no	yes	none	0	no	yes	water	fish
lizard	no	no	scale	4	no	yes	land	reptile
crocodile	no	no	scale	4	no	yes	water	reptile
t_rex	no	no	scale	4	no	yes	land	reptile
turtle	no	no	scale	4	no	yes	water	reptile
snake	no	no	scale	0	no	yes	land	reptile
eagle	no	no	feathers	2	yes	yes	air	bird
ostrich	no	no	feathers	2	yes	yes	land	bird
penguin	no	no	feathers	2	yes	yes	water	bird



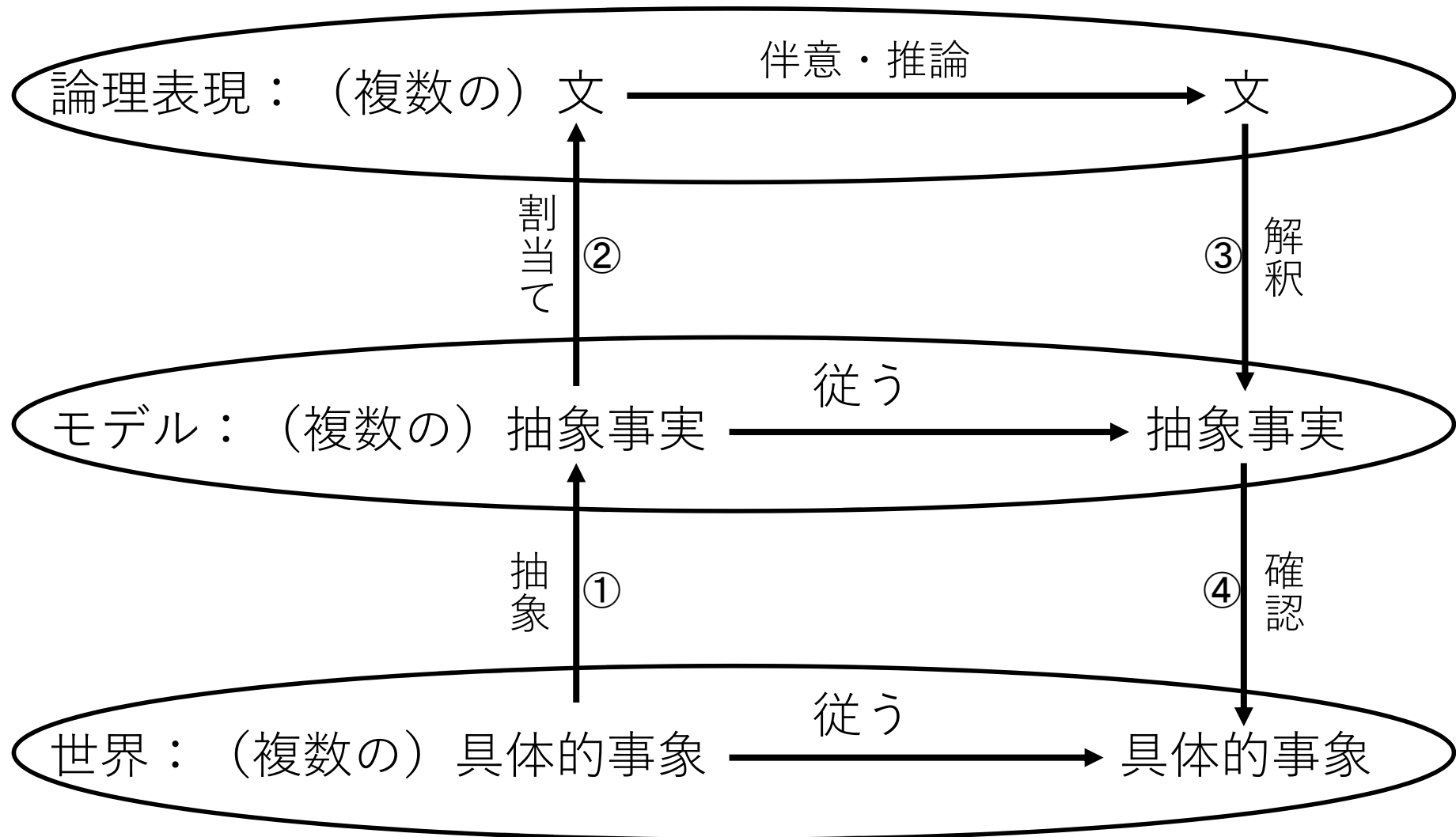
## 複数の表に跨るデータからの学習

列車	方向	列車	貨車	順番	貨車	屋根	長さ	形	荷物の形 (数)	車輪の数
train1	東	train1	car11	1	car11	なし	長	長方形	四角形 (3)	2
train2	東	<u>train1</u>	<u>car12</u>	2	<u>car12</u>	あり	短	長方形	三角形 (1)	2
train3	東	train1	car13	3	car13	なし	長	長方形	ひし形 (1)	3
train4	東	train2	car21	1	car21	なし	短	長方形 x2	三角形 (1)	2
train5	西	train2	car22	2	car22	あり	長	長方形	三角形 (1)	3
train6	西	<u>train2</u>	<u>car23</u>	3	<u>car23</u>	あり	短	長方形	丸 (1)	2
train7	西	train3	car31	1	car31	なし	短	長方形	丸 (1)	2
train8	西	<u>train3</u>	<u>car32</u>	2	<u>car32</u>	あり	短	長方形	三角形 (1)	2
		train3	car33	3	car33	あり	長	長方形	三角形 (1)	3
		train4	car41	1	car41	なし	短	U字形	三角形 (1)	2
		train4	car42	2	car42	なし	短	U字形	四角形 (1)	2
		<u>train4</u>	<u>car43</u>	3	<u>car43</u>	あり	短	長方形	丸 (2)	2
		train5	car51	1	car51	あり	長	長方形	丸 (3)	2
		train5	car52	2	car52	なし	短	長方形	三角形 (1)	2
		train6	car61	1	car61	なし	短	長方形 x2	丸 (1)	2
		train6	car62	2	car62	なし	短	U字形	三角形 (1)	2
		train6	car63	3	car63	なし	長	長方形	なし	2
		train7	car71	1	car71	あり	長	長方形	四角形 (1)	3
		train7	car72	2	car72	なし	短	U字形	丸形 (1)	2
		train8	car81	1	car81	なし	短	U字形	四角形 (1)	2
		train8	car82	2	car82	なし	長	長方形	四角形 (2)	2



## 論理による世界の表現とモデル

- 論理を用いて物事を表現する目的
  - 真偽についての一般的法則性を利用して「問題を解決する」こと



# 論理による世界の表現とモデル

- 問題解決の手順
  - ※難しく感じるかもしれませんが、「プログラミング」や「データ分析」と同じです
  - 「解きたい問題をプログラミング言語で表記し、実行することで答えを得る」
  - 「対象に関する属性間の関係を抽象化（簡略化）したモデル（数式）を考える」
- 手順1：世界の抽象表現を与える
  - 実世界は多様な構成要素と関係から成り立っている
  - 問題解決に関連する部分のみを切り出す（取り出す）
    - 細部は無視して、特定の属性に関する真偽のみを問題にする⇒抽象化
  - モデル：世界の抽象表現に対応する
    - モデルの表現は論理体系で異なる
- 手順2：モデル要素に対する割り当て（assignment）
  - モデルの各要素に対して、命題記号や述語記号を対応させる
  - 対応後は、論理体系の持つ推論能力（推論規則）を用いて、新たな文を導出する
- 手順3：割り当ての逆操作（解釈）
  - 文に対し、それがモデルにおいて何を意味するかを与える
- 手順4：確認
  - 抽象事実に対する具体事実が、本当に成り立っているかを確認する



## 本日のまとめ

- オリエンテーション
- 人工知能の歴史
  - 探索・推論・学習
- 論理推論
  - 演繹・発想・帰納
- 宣言的プログラミングと問題解決