Лекция 2: Представление знаний. Формальные системы

Панов А.И.
panov.ai@mipt.ru

Московский-физико технический институт
Центр когнитивного моделирования

2023 – Введение в методы искусственного интеллекта

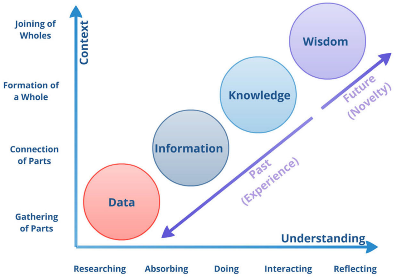
План лекции

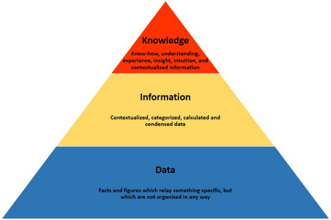
1. Представление знаний
2. Исчисление предикатов первого порядка
3. Формальная система
4. Истинность
5. Агент с базой знаний

Представление знаний

Знания и данные

- *Данные* – специализированный, но неорганизованный набор объективных наблюдаемых фактов (значений параметров)
- *Информация* – категоризованные данные в контексте
- *Знания* – интерпретируемая информация и данные для решения задач (*принятия решений*) со специальными процедурами, поддерживающими структурированность, непротиворечивость, автоматическое пополнение, логический вывод и т.д.

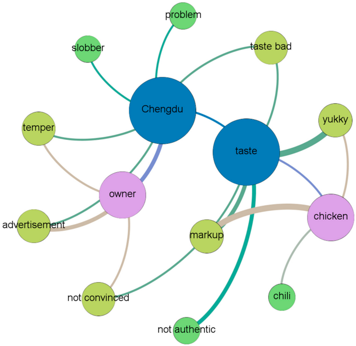




Data Mining vs **Knowledge Discovery**

Методы представлений знаний

- Системы правил
- Семантические сети
- Системы фреймов
- Дескриптивная логика



При описании многих способов представления знаний используется язык исчисления предикатов первого порядка

Исчисление предикатов первого порядка

Алфавит языка

Пусть L – язык исчисления предикатов первого, основные конструкции языка – *формулы*, определяемые на основе *алфавита*

Алфавит включает:

- счетное множество *переменных* $x, y, \ldots, z$,
- счетное множество *констант* $a, b, \ldots, c$,
- счетное множество *предикатных символов* $P, Q, \ldots,$
- счетное множество *функциональных символов* $f, g, \ldots,$
- символы для *логических связок* $\to$ (*влечет*) и $\neg$ (*не*),
- символ для *квантора* $\forall$ (*для любого*),
- скобки (,)

С каждым предикатным или функциональным символом будем связывать некоторое натуральное число n, равное числу его аргументов (n-местный или n-арный)

Формула языка

Формулы языка определяются индуктивным образом:

1. символ для обозначения переменной есть *терм*,
2. символ для обозначения константы есть терм,
3. если $t_1, t_2, \ldots, t_m, \ldots, t_n$ – термы, а f и g — функциональные символы арности m и n соответственно, то $f(t_1, t_2, \ldots, t_m)$ и $g(t_1, t_2, \ldots, t_n)$ также термы,
4. если $t_1, t_2, \ldots, t_m, \ldots, t_n$ – термы, а P и Q – предикатные символы арности m и n соответственно, то $P(t_1, t_2, \ldots, t_m)$ и $Q(t_1, t_2, \ldots, t_n)$ – атомарные формулы,
5. атомарная формула есть формула,
6. если A, B – формулы, то $A \to B$, $\neg A$, $\neg B$ – формулы,
7. если A – формула, то $\forall x A$ – формула,
8. всякое слово в алфавите языка является формулой тогда и только тогда, когда это можно показать с помощью конечного числа применений пп. 1–7

Язык исчисления предикатов первого порядка

Введение формул завершает одно из возможных определений языка исчисления предикатов первого порядка (ИППП).

Это определение *полное*, но можно определить и другие символы:

- логическая связка $\wedge$ (*и*): $A \wedge B = \neg(A \to \neg B)$,
- логическая связка $\vee$ (*или*): $A \vee B = \neg A \to B$,
- квантор $\exists$ (*существует*): $\exists x A(x) = \neg \forall x \neg A(x)$

Факты и предложения

- Вхождение переменной x в формулу A называется *связанным*, если эта переменная следует за кванторами существования или общности, предшествующими формуле A и находится в области их действия, иначе это *свободная* переменная
- Если в формуле A отсутствуют свободно входящие в нее переменные, то формула называется *замкнутой формулой* или *предложением*
- Атомарную замкнутую формулу называют *фактом*
- Если язык состоит только из предложений, то он называется *пропозициональным* языком, а буквы $A, B, \ldots,$ входящие в формулы этого языка — *пропозициональными переменными*

Аксиомы ИППП

Базовые аксиомы для логических связок:

- $A \to (B \to A)$,
- $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$,
- $(\neg A \to \neg B) \to (B \to A)$

Если в базовых аксиомах все переменные являются пропозициональными, то такое исчисление называется *пропозициональным исчислением* или *исчислением высказываний*

Аксиомы для кванторов:

- **аксиома генерализации**: $(\forall x)((A \to B) \to (A \to (\forall x)B))$, где x не является свободной переменной в A,
- **аксиома спецификации**: $\forall t A(t) \to A(x)$, где t – терм, а x не содержится в t в качестве свободной переменной

Правила вывода

- **Правило отделения**: если выводимо A и выводимо $A \to B$, то выводимо B
- **Правило подстановки**: в любую аксиому на место любой пропозициональной переменной можно подставить любое предложение, предварительно переименовав пропозициональные переменные подставляемого предложения так, чтобы они не совпадали с пропозициональными переменными аксиомы
- **Правило обобщения**: если выводимо A, то выводимо $\forall x A$, где x – свободная переменная в A

Пример вывода

Постулаты Аристотеля (P – пропозициональная переменная исчисления высказываний) – *теоремы*:

- $P \to P$ – закон тождества,
- $P \vee \neg P$ – закон исключенного третьего,
- $\neg(P \wedge \neg P)$ – закон противоречия

Доказательство закона тождества:

1. аксиома 1 и правило подстановки: $A \to ((P \to P) \to A))$,

Пример вывода

Постулаты Аристотеля (P – пропозициональная переменная исчисления высказываний) – *теоремы*:

- $P \to P$ – закон тождества,
- $P \vee \neg P$ – закон исключенного третьего,
- $\neg(P \wedge \neg P)$ – закон противоречия

Доказательство закона тождества:

1. аксиома 1 и правило подстановки: $A \to ((P \to P) \to A)$),
2. аксиома 2 и правило подстановки:
   $(A \to ((P \to P) \to C)) \to ((A \to (P \to P)) \to (A \to C))$,

Пример вывода

Постулаты Аристотеля (P – пропозициональная переменная исчисления высказываний) – *теоремы*:

- $P \to P$ – закон тождества,
- $P \vee \neg P$ – закон исключенного третьего,
- $\neg(P \wedge \neg P)$ – закон противоречия

Доказательство закона тождества:

1. аксиома 1 и правило подстановки: $A \to ((P \to P) \to A))$,
2. аксиома 2 и правило подстановки:
$(A \to ((P \to P) \to C)) \to ((A \to (P \to P)) \to (A \to C))$,
3. правило подстановки:
$\underbrace{(P \to ((P \to P) \to P))}_{X} \to \underbrace{((P \to (P \to P)) \to (P \to P))}_{Y}$,

Пример вывода

Постулаты Аристотеля (P – пропозициональная переменная исчисления высказываний) – *теоремы*:

- $P \to P$ – закон тождества,
- $P \vee \neg P$ – закон исключенного третьего,
- $\neg(P \wedge \neg P)$ – закон противоречия

Доказательство закона тождества:

1. аксиома 1 и правило подстановки: $A \to ((P \to P) \to A))$,
2. аксиома 2 и правило подстановки:
$(A \to ((P \to P) \to C)) \to ((A \to (P \to P)) \to (A \to C))$,
3. правило подстановки:
$\underbrace{(P \to ((P \to P) \to P))}_{X} \to \underbrace{((P \to (P \to P)) \to (P \to P))}_{Y}$,
4. правило отделения: часть X формулы является аксиомой, т. е. выводима, тогда выводима часть Y формулы
$\underbrace{(P \to (P \to P))}_{X'} \to \underbrace{(P \to P)}_{Y'}$,

Пример вывода

Постулаты Аристотеля (P – пропозициональная переменная исчисления высказываний) – *теоремы*:

- $P \to P$ – закон тождества,
- $P \vee \neg P$ – закон исключенного третьего,
- $\neg(P \wedge \neg P)$ – закон противоречия

Доказательство закона тождества:

1. аксиома 1 и правило подстановки: $A \to ((P \to P) \to A))$,
2. аксиома 2 и правило подстановки:
   $(A \to ((P \to P) \to C)) \to ((A \to (P \to P)) \to (A \to C))$,
3. правило подстановки:
   $\underbrace{(P \to ((P \to P) \to P))}_{X} \to \underbrace{((P \to (P \to P)) \to (P \to P))}_{Y}$,
4. правило отделения: часть X формулы является аксиомой, т. е. выводима, тогда выводима часть Y формулы
   $\underbrace{(P \to (P \to P))}_{X'} \to \underbrace{(P \to P)}_{Y'}$,
5. правило отделения: Y' – выводима

Формальная система

Формальная теория

Расширение на предметные области:

- Аксиомы ИППП (или математической логики первого порядка) называются *логическими аксиомами*, они описывают логические законы, справедливые всегда, независимо от предметной области
- Если же к языку и аксиомам ИППП добавить специальные функциональные символы и аксиомы, описывающие некоторую предметную область, то получим *формальную теорию*
- Добавление аксиом арифметики и символов арифметических операций приводит к *формальной арифметике*
- Добавление аксиом теории групп и символов групповых операций приводит к *формальной теории групп*

Логика высших порядков

Высшие порядки:

- в более сильных логиках используются и некоторые нелогические понятия, такие как множество,
- кванторы $\forall$ и $\exists$ действуют на некотором множестве U,
- логика второго порядка разрешает одному из кванторов действовать на подмножествах множества U и на функциях из степеней U в U,
- логика третьего порядка может использовать кванторы по множествам функций и т. д.

Формальные системы

Формальная система F представляет собой совокупность следующих объектов:

$$F = \langle T, R, A, \Pi \rangle$$

- T – счетное множество символов и операций над ними,
- R – множество правил грамматики, применение которых к символам из T позволяет строить правильно построенные формулы,
- A – множество аксиом,
- Π – множество правил вывода

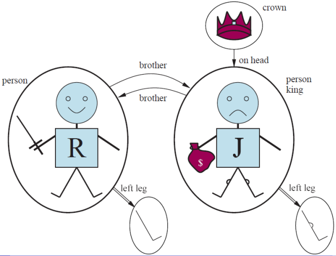
Если среди аксиом имеются нелогические аксиомы (аксиомы, описывающие некоторую предметную область), то формальная система называется *формальной теорией*

Вывод в формальной системе

- **Выводом** (или доказательством) в формальной системе называется конечная последовательность правильно построенных формул
$$A_1, A_2, \ldots, A_n,$$
таких что каждая из формул последовательности либо является аксиомой, либо получена из предыдущих формул последовательности с использованием аксиом и правил вывода
- Формула A_n называется выводимой формулой (или теоремой) формальной системы F

Алгебраические системы

- n-арным *отношением* R на M называется подмножество M^n: $R \subseteq M^n$
- Пусть M – непустое множество (*универсум*), тогда упорядоченная тройка $S = \langle M, G, R \rangle$, где универсум M называется основным множеством, G – множество n-местных функций из M^n в M, а R – семейство n-арных отношений на M, называется *алгебраической системой*
- Если $R = \emptyset$, то такая алгебраическая система называется *алгеброй*
- Если $G = \emptyset$, то тогда алгебраическая система называется *моделью*



Интерпретация

- Пусть заданы формальная система F с языком L и универсум M
- Зададим *интерпретирующее отображение* I, которое
  - всякому константному символу a языка L системы F ставит в соответствие некоторый элемент $m \in M$,
  - всякому n-местному предикатному символу P – n-местное отношение $R \subseteq M^n$,
  - всякому m-местному функциональному символу f – m-местную функцию $G : M^n \to M$
- При наличии интерпретирующего отображения алгебраическая система $S = \langle M, G, R \rangle$ называется *моделью* или интерпретацией формальной системы F

Истинность

Выполнимость и истинность

Вопрос об истинности или ложности формул языка L в некотором возможной модели S является нетривиальным

- Не существует регулярного метода, позволяющего по произвольной формуле языка L решать, истинна она или ложна в алгебраической системе (стандартной модели арифметики) $\langle N, +, *, \sigma, 0 \rangle$, где
  - N – множество целых чисел
  - $+, *, \sigma$ – операции сложения, умножения чисел и получения следующего числа, соответственно,
  - 0 – константа нуль (которую можно рассматривать как нульместную функцию),

Если A имеет вид $\neg B$ или $B \wedge C$, то ясно, что вначале надо решить вопрос об истинности B и C.

Определение истинности

Пусть A имеет вид $\forall x B$. Значения всякой свободной переменной в формуле B пробегают множество M, поэтому имеет смысл вопрос: является ли формула B истинной в модели S, если значением свободной переменной является m?

- Если для каждого m из M ответ на этот вопрос окажется утвердительным, то можно говорить, что A истинно в S
- Если же в M найдется элемент m, для которого ответ будет отрицательным, то будем говорить, что A ложно в S

Определение истинности формул в модели является индуктивной процедурой, основанной на индуктивном определении формулы языка L

Выполнимость в модели

Пусть P – n-местный предикатный символ, $x_1, x_2, \ldots, x_n$ – свободные переменные, I – интерпретирующее отображение модели Ω, тогда

1. атомарная формула $P(x_1, x_2, \ldots, x_n)$ *выполняется* в модели Ω, если найдется подстановка $a_1/x_1, a_2/x_2, \ldots, a_n/x_n$, где $a_1, a_2, \ldots, a_n$ – константы, такая что $(I(a_1), I(a_2), \ldots, I(a_n)) \in I(P)$;
2. формула $\forall x_1 \forall x_2 \ldots \forall x_n P(x_1, x_2, \ldots, x_n)$ истинна в модели Ω, если формула $P(x_1, x_2, \ldots, x_n)$ выполняется на любой n-ке $a_1, a_2, \ldots, a_n$;
3. формула $\neg A$ истинна в модели Ω, если A – ложна;
4. формула $A \to B$ истинна в модели Ω, если A – истинна и B – истинна, либо A – ложна и B – ложна, либо A – ложна, а B – истинна (в модели Ω);
5. формула может быть истинной только в силу пп. 1-4.

Предложение, истинное в каждой модели языка L, называется истинным (общезначимым)

Пример истинности

- Пусть в языке L определен трехместный предикатный символ *СЕМЬЯ*
- Интерпретирующее отображение I таково, что ставит в соответствие этому предикатному символу трехместное отношение **СЕМЬЯ**

Мать	Отец	Ребенок
Маша	Петя	Вася
Галя	Женя	Коля

- Модель включает универсум M, состоящий из имен членов всех семей и трехместное отношение **СЕМЬЯ**
- Предложение $\forall x \forall y \forall z\, СЕМЬЯ(x, y, z)$ ложно в построенной модели
- Предложение $\exists x \exists y \exists z\, СЕМЬЯ(x, y, z)$ истинно в построенной модели

Полнота и непротиворечивость формальных систем

- Теория, в которой множество теорем покрывает всё множество формул (все формулы являются теоремами, "истинными высказываниями"), называется **противоречивой**; в противном случае теория называется **непротиворечивой**
- Выяснение противоречивости теории – одна из важнейших и сложнейших задач формальной логики
- Теория называется **полной**, если в ней для любого предложения (замкнутой формулы) F выводимо либо само F, либо его отрицание $\neg F$; в противном случае, теория содержит недоказуемые утверждения (утверждения, которые нельзя ни доказать, ни опровергнуть средствами самой теории), и называется **неполной**

Теоремы Гёделя

Theorem (Теорема Гёделя о полноте)

Произвольное предложение языка L выводимо в исчислении предикатов первого порядка тогда и только тогда, когда оно истинно

Theorem (Первая теорема Гёделя о неполноте в форме Россера)

Существуе такая формула B, что если формальная система S непротиворечива, то в ней невыводимы обе формулы B и $\neg B$; иначе говоря, если система S непротиворечива, то она неполна, и B служит примером неразрешимой формулы.

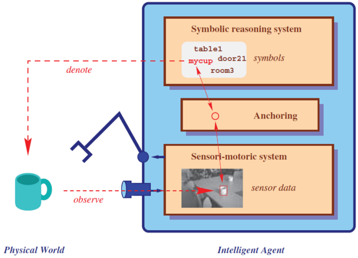
Theorem (Вторая теорема Гёделя о неполноте)

Если формальная арифметика S непротиворечива, то в ней невыводима формула, содержательно утверждающая непротиворечивость S

Агент с базой знаний

Использование знаний

- Агент обладает базой знаний, то есть набором высказываний на некотором языке, интерпретируемых в некоторой модели
- Рассуждения агенты: проверка выводимости некоторого высказывания и проверка его выполнимости в модели
- Проблема привязки (grounding) – философская проблема связи базы знаний и реальной среды (не модели!)
- Правильная работа сенсоров агента позволяет ослабить проблему привязки (символов)



Заключение

- Данные, информация и знания – это разные концепции, отличающиеся уровнем организации и наличием активных процедур (вывода)
- К способам представлениям знаний относятся логические системы, системы правил, семантические сети и т.д.
- Логика предикатов первого порядка – мощный инструмент представлений знаний со своей семантикой и синтаксисом
- Основной инструмент моделирования рассуждений – логический вывод
- Выполнимость (истинность) имеет смысл только при определении некоторой модели
- Ограничения ЛППП: теоремы Гёделя и проблема привязки

Спасибо за внимание!

mipt.ru

rairi.ru

raai.org

panov.ai@mipt.ru