

СРАВНЕНИЕ РАЗЛИЧНЫХ ПРЕДСТАВЛЕНИЙ ЗНАНИЙ ДЛЯ СЛОЖНЫХ СТРУКТУРИРОВАННЫХ ОБЪЕКТОВ ПРИ РЕШЕНИИ ЗАДАЧ ИИ*

Косовская Т. М.¹, доктор физико-математических наук, доцент, ✉ kosovtm@gmail.com

¹ Санкт-Петербургский государственный университет, Университетский пр., д. 28, Старый Петергоф, 198504, Санкт-Петербург, Россия

Аннотация

Задача представления знаний для сложных структурированных объектов является одной из актуальных задач ИИ. Это связано с тем, что многие исследуемые объекты представляют собой не единый неделимый объект, характеризующийся своими свойствами, а сложные конструкции, элементы которых обладают известными свойствами и находятся в некоторых, зачастую многоместных, отношениях между собой.

В работе подход к представлению таких знаний на основе логики первого порядка (формул исчисления предикатов) сравнивается с двумя широко распространёнными в настоящее время подходами, основанными на представлении информации о данных с помощью конечнозначных строк и на использовании графов.

Показано, что использование формул исчисления предикатов для описания сложных структурированных объектов, несмотря на NP-трудность решаемых задач, возникающих после формализации, реально имеют не бóльшую вычислительную сложность, чем два других подхода, о чём обычно не упоминают их сторонники.

Предложен алгоритм построения онтологии, не зависящий от способа описания объекта и основанный на выделении наибольшего общего свойства объектов из заданного множества.

Ключевые слова: *Сложный структурированный объект, представление данных, формулы исчисления предикатов, строка признаков, граф знаний, онтология.*

Цитирование: Косовская Т. М. Сравнение различных представлений знаний для сложных структурированных объектов при решении задач ИИ // Компьютерные инструменты в образовании. 2021. № 2. С. 41–57. doi: 10.32603/2071-2340-2021-2-41-57

1. ВВЕДЕНИЕ

Во многих задачах Искусственного Интеллекта (ИИ) осуществляется переход ДАННЫЕ => ИНФОРМАЦИЯ => ЗНАНИЯ => Мудрость, зачастую обозначаемый аббревиатурой

* Исследование поддержано Санкт-Петербургским государственным университетом, проект № 73555239.

DIKW [1–4]. В настоящей работе будет рассмотрен только переход ДАННЫЕ => ИНФОРМАЦИЯ => ЗНАНИЯ в зависимости от способа задания описания данных. Классическая пирамида знаний представлена на рисунке 1.



Рис. 1. Пирамида знаний

Что же такое Данные в задачах ИИ?

Любой исследователь в области ИИ имеет дело не с объектами, а с их описаниями. Нельзя «загрузить» в компьютер (а именно с этого начинается использование ИИ) конкретные объекты: человека, улицу, внутренние органы, мысли, обстановку квартиры, город, растение и т. п. Компьютер может иметь дело только с их описаниями на некотором формализованном языке: таблицы, формулы, тексты, изображения на экране дисплея или фото и т. п. При этом такие описания одних и тех же объектов для решения разных задач могут быть в значительной степени различны. Таким образом, данные записаны на формализованном языке, выбранном исследователем для решения конкретной задачи. Причём создание описаний должно быть достаточно простым (по крайней мере, для работающих в заданной области).

Для перехода от Данных к Информации при использовании различных языков описания объектов могут использоваться разные средства. Например, вычисление ковариационной матрицы, выделение общих подформул, вычисление производных (градиентов) или интегралов функций. Всё зависит от обоснованных предпочтений исследователя и его способности с помощью этих средств решить поставленную задачу выделения информации, то есть отбрасывание несущественных для данного исследования характеристик объекта. И этот переход осуществляется уже в рамках выбранного языка. Исследователь должен уметь трактовать, какая же информация была получена, переведя формализованный язык на понятный людям, работающим в этой области.

Для перехода от Информации к Знаниям следует создать правила, которые позволят обобщить сведения об объектах, по которым создавалась Информация, на новые объекты. Причём эти правила «привязаны» к тому языку, на котором описывались объекты.

Использование различных языков для решения одной и той же задачи зачастую вызывает непонимание разных исследователей, работающих в одной области. Могут привести пример очень сложной задачи объединения двух онтологий.

Пусть для двух пересекающихся множеств объектов созданы две онтологии. При этом каждая система создавалась на основе своей системы признаков. Для этого требуется выразить одни признаки через другие. Это достаточно сложно и похоже, например, на то, как договориться русскому и китайцу, какое поведение правильное, а какое нет.

Безусловно, выбор признаков (точнее, языка описания) в значительной степени важен при решении любой задачи ИИ. Описание в виде набора значений признаков [5–8] проще для понимания и работы нематематику, чем описания на языке исчисления предикатов (ИП). Кроме того, основные задачи, описанные такими наборами, имеют, как правило, полиномиальную от длины записи строки значений признаков вычислительную сложность. В то же время задачи, возникающие при описании сложных структурированных объектов с использованием ИП, как правило, NP-трудны, а следовательно, имеют экспоненциальную от длины записи описания объекта на языке ИП вычислительную сложность.

Однажды в разговоре с профессором Николаем Григорьевичем Загоруйко, одним из первых специалистов-математиков, которые начали заниматься онтологиями, он удивился, зачем привлекать предикатные формулы. Ведь для того чтобы идентифицировать машину марки Мерседес, достаточно иметь признак наличия трёхлучевой звезды. На что я возразила, что если мальчишки отвинтят эту звезду и прикрепят её к Запорожцу, то тот не станет Мерседесом, а Мерседес не перестанет быть Мерседесом. Наличие этого признака достаточно при выходе машины с завода или из салона, но недостаточно при отслеживании машин в уличном потоке.

В работе будет показано, что если описание сложного структурированного объекта на языке ИП промоделировать с помощью бинарной строки, то длина записи такой строки экспоненциальна от длины записи его описания на языке ИП.

В настоящее время широко распространено представление данных в виде графа [9, 10]. В частности, компания Google успешно применяет графы знаний, позволяющие эффективно находить сведения об объекте или явлении. Однако информации о том, как строить граф знаний, в доступных источниках достаточно мало.

С точки зрения автора настоящей статьи, одним из недостатков описания сложного структурированного объекта в виде графа является то, что с помощью графа хорошо моделируются бинарные отношения между элементами объекта, но уже моделирование трёхместного отношения (а тем более многоместного) вызывает трудности.

Ниже будет приведён модельный пример выделения трёхместного отношения между элементами графа. Показано, что задача такого выделения NP-трудна.

Очень близко к применению графа знаний находится применение онтологий [11–13]. В настоящей работе рассматривается способ построения онтологии для множества сложных структурированных объектов. Приведённый алгоритм не зависит от языка, на котором описаны объекты. Но используемое в алгоритме понятие «наибольшее общее свойство объектов» зависит от языка описания.

В статье будут рассмотрены и проведено сравнение двух подходов к описанию сложных структурированных объектов

- задание описания объекта с помощью строки значений (бинарных или конечных значений) признаков,
- задание описания объекта с помощью графа с подходом, основанным на задании описания объекта формулой исчисления предикатов (ИП) [14, 15].

Кроме того, будет рассмотрен подход к формированию онтологии сложных структурированных объектов.

Во всех этих подходах Данные — это сложные структурированные объекты. Информация об объекте — это его описание на выбранном языке. Знание о множестве рассматриваемых объектов — это выделение общих свойств этих объектов, отличающих их от объектов других множеств, и умение верифицировать объект, то есть применение алго-

ритма проверки того, удовлетворяет ли объект сформулированному целевому условию, задающему выделенные общие свойства объектов из заданного множества.

2. ИСПОЛЬЗОВАНИЕ ЯЗЫКА ИП ПРИ РЕШЕНИИ ЗАДАЧ ИИ

Использование языка ИП при решении задач ИИ было предложено многими авторами ещё в 60-е годы XX века, например, в [14]. Ниже будет использоваться логико-предметный подход, описанный в [15].

Сложный структурированный объект ω представляется как множество своих элементов $\omega = \{\omega_1, \dots, \omega_t\}$, на котором задан набор предикатов p_1, \dots, p_n , характеризующих свойства этих элементов (одноместные предикаты) и отношения между ними (многоместные предикаты).

Описанием $S(\omega)$ объекта ω называется множество истинных на ω литералов.

Для задания Знания о множестве объектов требуется выделить их общие свойства в виде элементарных конъюнкций предикатных формул. В [16] описан один из возможных способов создания такого описания по обучающей выборке, в основе которого лежит понятие изоморфизма предикатных формул [17], определение которого будет дано ниже в этом разделе.

Задача верификации того, что ω удовлетворяет некоторому заданному свойству $A(\bar{x})$, где $A(\bar{x})$ — формула ИП с предикатами p_1, \dots, p_n ¹, \bar{x} — список переменных, сводится к проверке логического следования

$$S(\omega) \Rightarrow \exists \bar{x}_{\neq} A(\bar{x}), \quad (1)$$

где $\exists \bar{x}_{\neq}$ означает проверку существования попарно различных значений переменных из списка \bar{x} .

В [15] доказаны NP-полнота проверки (1) и экспоненциальные оценки её вычислительной сложности

$$O(t^m) \text{ и } O(s^a)$$

при применении алгоритма перебора и поиска вывода в исчислении предикатов соответственно. Здесь t — количество элементов в объекте ω , m — максимальное количество переменных у предикатов, входящих в формулу $A(\bar{x})$, s — количество литералов в описании $S(\omega)$, a — наибольшее количество вхождений литералов с одним и тем же предикатным символом в формулу $A(\bar{x})$.

Введённые впоследствии автором понятия неполного логического следования и изоморфизма элементарных конъюнкций предикатных формул [17] позволили существенно уменьшить показатель в экспоненциальной оценке при многократной проверке следования (1) для заранее заданного множества различных правых частей.

Приведём здесь некоторые определения, которые позволят понимать, что же такое общее свойство сложных структурированных объектов.

Определение 1. [17] *Две элементарные конъюнкции*

$$A(x_1, \dots, x_n),$$

$$B(y_1, \dots, y_n)$$

¹ В качестве формул $A(\bar{x})$ достаточно рассматривать только элементарные конъюнкции с предикатами p_1, \dots, p_n .

атомарных предикатных формул называются **изоморфными**, если существуют такая элементарная конъюнкция

$$C(z_1, \dots, z_n)$$

и такие подстановки

$$\lambda_{A,C} = \begin{matrix} (x_1, \dots, x_n) \\ (z_{i_1}, \dots, z_{i_n}) \end{matrix}, \quad \lambda_{B,C} = \begin{matrix} (y_1, \dots, y_n) \\ (z_{j_1}, \dots, z_{j_n}) \end{matrix}$$

различных переменных из формулы $C(z_1, \dots, z_n)$ вместо различных аргументов формулы $A(x_1, \dots, x_n)$ и различных аргументов формулы $B(y_1, \dots, y_n)$ соответственно, что результаты этих подстановок совпадают с формулой $C(z_1, \dots, z_n)$ с точностью до порядка литералов

$$A\lambda_{A,C} = A(z_{i_1}, \dots, z_{i_n}) = C(z_1, \dots, z_n),$$

$$B\lambda_{B,C} = B(z_{j_1}, \dots, z_{j_n}) = C(z_1, \dots, z_n).$$

При этом подстановки $\lambda_{A,C}$ и $\lambda_{B,C}$ называются **унификаторами** формул $A(x_1, \dots, x_n)$ и $B(y_1, \dots, y_n)$ соответственно с формулой $C(z_1, \dots, z_n)$.

Понятие изоморфизма формул отличается от понятия их равносильности. Это, скорее, понятие равенства формул с точностью до имён переменных и порядка следования литералов. Более точно, изоморфные формулы задают одно и то же отношение между своими аргументами, быть может переставленными в другом порядке.

Задача проверки двух элементарных конъюнкций предикатных формул полиномиально эквивалентна «открытой» задаче проверки изоморфизма графов, про которую не доказана её NP-полнота и не найден полиномиальный алгоритм её решения.

Например, очевидно, что формулы

$$A(x, y) = p(x) \& p(y) \& q(x, y),$$

$$B(u, v) = p(u) \& p(v) \& q(v, u)$$

не равносильны, но изоморфны. Так как при введении в рассмотрение формулы

$$C(z_1, z_2) = p(z_1) \& p(z_2) \& q(z_1, z_2)$$

и подстановок

$$\begin{matrix} x, y \\ z_1, z_2 \end{matrix}, \quad \begin{matrix} u, v \\ z_2, z_1 \end{matrix}$$

имеем

$$A(z_1, z_2) = p(z_1) \& p(z_2) \& q(z_1, z_2) = C(z_1, z_2),$$

$$B(z_2, z_1) = p(z_2) \& p(z_1) \& q(z_2, z_1) = C(z_1, z_2).$$

Последнее равенство имеет место с точностью до порядка первых двух конъюнктивных членов.

При решении некоторых задач ИИ требуется выделить наибольшее общее свойство, которому удовлетворяют объекты из двух множеств. При этом описание (общее свойство) объектов каждого из этих множеств известно. Следующее определение относится как раз к такой задаче.

Определение 2. [17] *Элементарная конъюнкция*

$$C(z_1, \dots, z_k)$$

называется **наибольшей общей** (с точностью до имён переменных)² **подформулой двух элементарных конъюнкций**

$$A(x_1, \dots, x_n),$$

$$B(y_1, \dots, y_m),$$

если она изоморфна некоторым подформулам этих элементарных конъюнкций, но после добавления в неё хоть одного литерала становится не изоморфной ни одной подформуле либо $A(x_1, \dots, x_n)$, либо $B(y_1, \dots, y_m)$.

В качестве примера можно привести результат выделения наибольшей общей подформулы формул, задающих параллелограмм и трапецию. Такая подформула задаёт наибольшее общее свойство этих множеств объектов.

Всякий четырёхугольник может быть задан как упорядоченная четвёрка имён точек на плоскости (X, Y, Z, U) . Отрезок с концами x и y будем обозначать xy , длину такого отрезка $|xy|$.

На множестве четырёхугольников заданы предикаты

- четырёхместный предикат параллельности сторон $(xy \parallel zu)$,
- четырёхместный предикат равенства длин сторон $(|xy| = |zu|)$.

Множество всех параллелограммов вида (x, y, z, u) , определяемых как «четырёхугольник, у которого противоположные стороны равны и параллельны», задаётся формулой

$$(xy \parallel zu) \ \& \ (yz \parallel ux) \ \& \ (|xy| = |zu|) \ \& \ (|yz| = |xu|). \quad (*)$$

Множество всех трапеций вида (u, z, y, x) , определяемых как «четырёхугольник, у которого две противоположные стороны параллельны, а две другие не параллельны», задаётся формулой

$$(uz \parallel yx) \ \& \ \neg(zu \parallel xi). \quad (**)$$

Несложно видеть, что наибольшей общей подформулой (*) и (**) является формула $(xy \parallel zu)$, которая задаёт свойство четырёхугольника «две противоположные стороны параллельны» и изоморфна первому члену (**) с унификатором $\begin{matrix} (u, z, y, x) \\ (x, y, z, u) \end{matrix}$.

Из этого следует, что так определённые множества параллелограммов и трапеций имеют общее свойство, но не пересекаются, так как формула (*) не имеет ни одного конъюнктивного члена с отрицанием.

В некоторой литературе множество всех трапеций вида (u, z, y, x) определяется как «четырёхугольник, у которого две противоположные стороны параллельны» и оно задаётся формулой

$$(uz \parallel yx). \quad (***)$$

Наибольшей общей подформулой (*) и (***) является та же формула $(xy \parallel zu)$, но при этом множество параллелограммов является подмножеством множества трапеций, так как из (*) следует (***)).

В работе [18] описаны некоторые ранее опубликованные автором задачи, которые могут быть решены с помощью логико-предметного подхода и использования понятий изоморфизма предикатных формул и выделения наибольшей общей подформулы заданных предикатных формул, точнее, наибольшего общего свойства объектов из двух множеств:

- задание метрики в пространстве элементарных конъюнкций предикатных формул, что позволяет измерять степень похожести исследуемых объектов;

² Ниже слова «с точностью до имён переменных» для краткости будут опускаться.

- построение многоуровневого описания классов в задачах распознавания, существенно снижающего вычислительную сложность задач при их многократном решении с заданным множеством свойств, записанных в правой части (1);
- построение логических баз данных, устанавливающих структуру связей между элементами заданного множества объектов;
- построение логических онтологий, а именно, построение иерархии (ориентированного графа) в множестве сложных структурированных объектов, описанных формулами ИП так, что каждый объект из выделенного подмножества объектов обладает одним и тем же свойством, причём объекты из множества «дочерней» вершины обладают всеми свойствами объектов из множества «отцовской» вершины плюс некоторыми дополнительными;
- формирование предикатной сети, аналогичной нейронной сети, но для сложных структурированных объектов, описанных на языке ИП, и имеющей возможность перестраивать свою конфигурацию (количество слоёв и количество ячеек в слое) в процессе обучения;
- формирование нечёткой предикатной сети, позволяющей распознавать с вычисляемой степенью уверенности новые объекты, на которых сеть не была обучена.

3. ЛОГИЧЕСКИЙ ПОДХОД И ИСПОЛЬЗОВАНИЕ СТРОК ЗНАЧЕНИЙ ПРИЗНАКОВ ПРИ РЕШЕНИИ ЗАДАЧ ИИ

В обширной статье [7] приведена таблица использования различных логик для решения разных задач ИИ. Там, в частности, отмечается, что для решения задач формализации и верификации используются только временные и вероятностные логики. Часть этой таблицы приведена в таблице 1.

Таблица 1. Виды логики для области применения. Акронимы и аббревиатуры: FOL (First-Order Logic); DL (Description Logic); BDI (Belief–Desire–Intention (Logic)); TL (Temporal Logic); FL (Fuzzy Logic); PL (Probabilistic Logic); DR (Defeasible Reasoning); CLP (Constraint Logic Programming)

	FOL	DL	BDI	TL	FL	PL	DR	CLP
Formalization & Verification				X		X		
Cognitive Agents	X	X	X	X	X		X	
Planning & Task Allocation	X	X					X	X
Robotics & Control		X		X	X			X

В связи с этим использование языка ИП в этой статье будет отделено от стандартно используемых для описания объектов пропозициональных (булевских) переменных.

К логическому подходу также часто относят использование описания в виде конечно-значной строки, задающей значения (быть может, многозначные) признаков объекта.

Ниже будем считать, что все признаки бинарны. Это не сильно изменит длину записи строки, так как всякий k -значный признак можно заменить на $\lfloor \log k \rfloor$ бинарных признаков³.

Постановка задач, как правило, начинается с задания исходных данных и определения того, что такое описание объекта.

³ Хотя на практике k -значный признак обычно заменяют на k бинарных.

На объекте ω задан набор бинарных признаков q_1, \dots, q_m , характеризующих свойства объекта. Описанием $S(\omega)$ объекта ω называется бинарная строка $(\alpha_1, \dots, \alpha_m)$ где $\alpha_j = 1$, если и только если ω обладает признаком q_j .

В контексте настоящей работы Данные — это сложные структурированные объекты, Информация — описания объектов в виде бинарной строки, Знания — это совокупность свойств объектов из заданного множества.

Свойство объекта задаётся, например, строкой вида $(\beta_1, \dots, \beta_m)$, где $\beta_j \in \{0, 1, \lambda\}$, причём символ λ означает, что вместо него может находиться как 0, так и 1.

За десятилетия использования такого подхода выработано большое количество широко применяемых методов выделения общих свойств объектов, описания которых задаются в виде бинарной строки [19, 20].

Задача верификации того, что объект обладает заданным свойством, при этом сводится к проверке

$$(\beta_1, \dots, \beta_m) \text{ является подстрокой } (\alpha_1, \dots, \alpha_m). \quad (2)$$

Очевидно, что вычислительная сложность проверки (2) квадратична⁴ относительно длины записи строк.

Возникает вопрос: «Зачем использовать постановку задачи в терминах ИП, которая NP-полна, если в другой постановке она квадратична?». Кроме того, использование логико-предметного подхода требует более глубокого знания математической логики, чем знакомство с булевыми функциями и умения работать со строками.

Давайте разберёмся, какова длина записи бинарной строки, задающей описание сложного структурированного объекта.

Напомним, что сложный структурированный объект ω представлен как множество своих элементов $\omega = \{\omega_1, \dots, \omega_t\}$, на котором задан набор предикатов p_1, \dots, p_n , характеризующих свойства этих элементов и отношения между ними. Описанием $S(\omega)$ объекта ω называется множество истинных на ω литералов.

При моделировании такого описания с помощью бинарной строки необходимо ввести пропозициональные (булевские) переменные. При этом

- для каждого одноместного предиката p_i^1 требуется ввести t пропозициональных переменных $q_{i,1}^1, \dots, q_{i,t}^1$ таких, что

$$(q_{i,j}^1 = 1) \Leftrightarrow p_i^1(\omega_j);$$

- для каждого 2-местного предиката p_i^2 требуется ввести t^2 пропозициональных переменных $q_{i,1,1}^2, \dots, q_{i,t,t}^2$ таких, что

$$(q_{i,j,l}^2 = 1) \Leftrightarrow p_i^2(\omega_j, \omega_l);$$

⋮

- для каждого k -местного предиката p_i^k требуется ввести t^k пропозициональных переменных $q_{i,1,\dots,1}^k, \dots, q_{i,t,\dots,t}^k$ таких, что

$$(q_{i,j_1,\dots,j_k}^k = 1) \Leftrightarrow p_i^k(\omega_{j_1}, \dots, \omega_{j_k}).$$

⁴Автору могут возразить, что эта сложность линейна, но в теории сложности алгоритмов принято измерять сложность в числе шагов машины Тьюринга, а она квадратична для этой задачи. Имеется большое количество методов, позволяющих уменьшить эту оценку.

Суммируя эти оценки, получаем, что длина записи бинарной строки составит $O(t')$, где t' — наибольшее количество аргументов у предикатов p_1, \dots, p_n . Эта оценка совпадает с оценкой числа шагов проверки логического следования (1) при использовании переборного алгоритма [15]. Такая же оценка приведена в [21] для решения экономических задач при их моделировании бинарными строками.

Получили, что квадратичная оценка числа шагов решения задачи (2) квадратична только от длины записи бинарной строки, задающей описание сложного структурированного объекта. При этом сама длина записи этой строки экспоненциальна от длины записи описания сложного структурированного объекта на языке ИП.

Но ведь экспонента, возведённая в квадрат остаётся экспонентой. Следовательно, вычислительная сложность задачи (2) экспоненциальна от длины записи описания сложного структурированного объекта на языке ИП.

Выделение наибольшего общего свойства двух и более строк одинаковой длины заключается в нахождении такой строки, у которой в каждой позиции значение либо совпадает со значением в этой позиции у всех строк, либо равно специальному символу λ , если хотя бы в двух строках в этой позиции значения различны.

Такое выделение также полиномиально от длины записи этих строк и их количества. Но уже было показано, что сами длины строк экспоненциально зависят от длины записи описания объектов на языке ИП.

4. ПРЕДСТАВЛЕНИЕ ДАННЫХ С ПОМОЩЬЮ ГРАФОВ

Ещё одним широко распространённым способом описания структурированного объекта является его описание с помощью графа. Распространено мнение, что ВСЁ можно задать с помощью графов. При этом обход графа осуществляется достаточно быстро, а именно за линейное от числа вершин и рёбер шагов.

Описание сложного структурированного объекта может представлять собой граф, в вершинах которого находятся имена объектов (элементы множества исследуемых объектов), помеченные именами свойств этих объектов, а рёбра — бинарные отношения между этими объектами, причём рёбра помечены именами отношений. Рёбра могут быть как ориентированными, если соответствующее бинарное отношение не симметрично (например, предикат любить), так и не ориентированными (например, предикат дружить).

Таким образом, граф, описывающий сложный структурированный объект, является мультиграфом, у которого каждая вершина помечена конечным числом меток из заданного множества имён одноместных предикатов, а пары вершин соединены некоторым количеством ориентированных и неориентированных рёбер, каждое из которых помечено именем бинарного отношения из заданного множества бинарных отношений.

Это вполне соответствует тому, что имеется множество ω , элементами которого являются исследуемые объекты, на которых заданы одноместные предикаты, задающие свойства этих объектов, и двуместные предикаты, задающие бинарные отношения между объектами.

Недостатком описания структуры с помощью такого графа по сравнению с логико-предметным подходом является то, что в нём плохо представимы многоместные (с числом аргументов более двух) отношения. Для выделения многоместного отношения, присутствующего среди объектов, необходимо в уже построенном графе найти хотя бы 2 изоморфных подграфа, для которых можно установить такую биекцию между вершинами,

что каждая пара вершин в этой биекции обладает, по крайней мере, некоторыми одинаковыми свойствами. При этом в каждом множестве отношений между смежными вершинами имеется, по крайней мере, одно отношение между вершинами, являющимися их образами для найденной биекции.

Стоит ли говорить о том, что поиск двух изоморфных подграфов — это экспоненциально сложная задача⁵.

Для примера рассмотрим изображённый на рисунке 2 результат выделения подграфа трёхместного отношения.

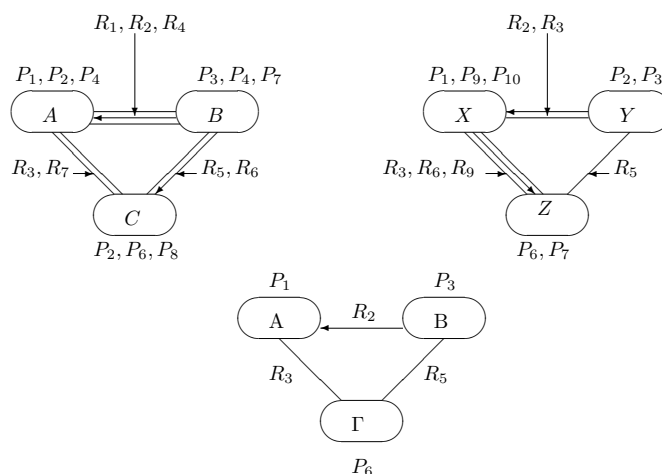


Рис. 2. Результат выделения подграфа трёхместного отношения.

Пусть в некотором графе, описывающем сложный структурированный объект (вообще говоря, в помеченном ориентированном мультиграфе, поскольку объекты в вершинах могут обладать несколькими свойствами, а пары вершин находиться в нескольких отношениях, не всегда симметричных), имеется 2 подграфа (их нужно ещё найти!) с вершинами A, B, C и X, Y, Z соответственно, изображённые в верхней части рисунка. Известны свойства P_1, \dots, P_{10} элементов, соответствующих вершинам, и бинарные отношения R_1, \dots, R_9 между ними.

В частности, элемент A обладает свойствами $P_1(A), P_2(A)$ и $P_4(A)$, элементы X и Y находятся в двух отношениях R_2 и R_3 , причём отношение R_2 не симметрично, то есть выполняется $R_2(Y, X)$, но нельзя гарантировать $R_2(X, Y)$.

В нижней части рисунка изображен граф, задающий трёхместное отношение $R^3(A, B, \Gamma)$, изоморфный подграфам графов в верхней части рисунка, что позволяет гарантировать $R^3(A, B, C)$ и $R^3(X, Y, Z)$.

Мне могут возразить, что если изначально известны многоместные отношения между элементами, то соответствующие подграфы можно снабдить метками этих многоместных отношений, приписанными как вершинам и их свойствам, так и рёбрам вместе с метками бинарных отношений.

Но не слишком ли много меток получают элементы графа?

Представление данных в виде графа нашло практическое применение в виде графа знаний [9]. Граф знаний — это семантическая технология и база знаний, позволяющая

⁵ Задача нахождения подграфа, изоморфного заданному, NP-трудна [22].

находить сведения об объекте или явлении, практическое использование которого началось в 2012 г. компанией Google. Тем не менее, стоит отметить, что нахождение сведений и извлечение знаний об объекте — это несколько различные задачи.

В этой же работе отмечается, что «Однако до сих пор не ясно, как воспользоваться данной технологией на практике из-за небольшого количества имеющейся информации по этой теме».

Графы знаний имеют большую аналогию как с базами данных, так и с онтологиями.

5. ОНТОЛОГИИ

В настоящее время большое внимание при рассмотрении задач ИИ уделяется онтологиям [11–13].

С одной стороны, онтология как предмет исследования философов представляет из себя абстракцию, которая учитывает одни свойства объектов или отношения между ними и не принимает во внимание другие.

С другой стороны, для математика онтология — это ориентированный граф, в котором множество объектов из «дочерней» вершины являются подмножеством объектов «отцовской» и обладают всеми свойствами и находятся в тех же отношениях, что и объекты «отцовской», но, кроме этого, обладают некоторыми специфическими свойствами. В поисковых системах онтологии используются, например, для поиска объектов с одинаковыми свойствами.

Чисто формально область знаний, для которой составляется онтология, представляет собой сложный структурированный объект, элементы которого сами являются сложными структурированными объектами, которые могут быть описаны средствами заданных свойств и некоторых известных отношений.

В качестве примера всем известной со школьных лет онтологии может служить онтология четырёхугольников, представленная на рисунке 3.

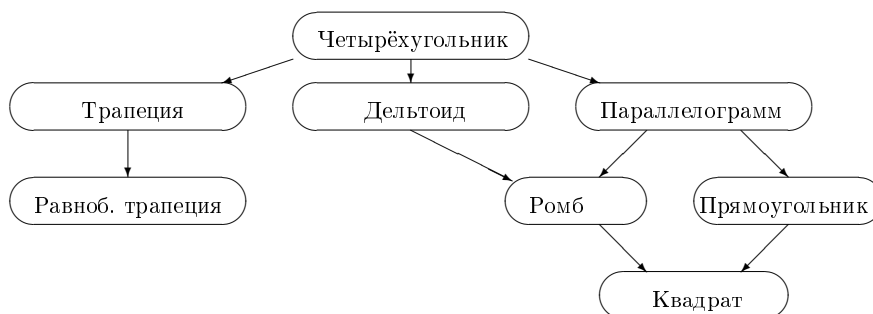


Рис. 3. Онтология четырёхугольников

Этот пример показателен, поскольку каждый конкретный четырёхугольник является сложным структурированным объектом, в котором вершины и стороны находятся в известных отношениях.

Если с использованием онтологий в настоящее время всё более или менее понятно, то как её составить по представительной выборке — это вопрос.

В [23] описан способ построения онтологии для сложных структурированных объектов, описанных на языке ИП. В основе его лежит понятие изоморфизма предикатных формул.

Опишем здесь одну из возможных постановок задачи построения онтологии по представительной выборке. В этой постановке не будет делаться различия, на каком языке описаны объекты.

Постановка задачи построения онтологии.

Дано множество сложных структурированных объектов $\Omega = \{\omega^1, \dots, \omega^m\}$ и описания объектов $S(\omega^1), \dots, S(\omega^m)$.

Требуется построить ориентированный граф, у которого вершина с нулевым заходом помечена множеством Ω_0 и соответствует множеству различных⁶ описаний, каждое из которых совпадает с некоторым описанием объекта из Ω .

Если из вершины Ω_k выходят дуги к вершинам, помеченным множествами $\Omega_{k_1}, \dots, \Omega_{k_r}$, то

- $\Omega_k = \cup_{i=1}^r \Omega_{k_i}$;
- для каждого $i = 1, \dots, r$ объекты из Ω_{k_i} обладают всеми свойствами, общими для всех объектов из Ω_k ;
- если $i \neq j$, то наибольшие общие свойства элементов из Ω_{k_i} и Ω_{k_j} различны.

Алгоритм решения задачи построения онтологии.

1. Для каждой пары описаний $S(\omega^i)$ и $S(\omega^j)$ ($1 \leq i < j \leq m$), входящих $S(\Omega)$, выделяем их максимальное общее свойство S_{ij}^1 .
Если среди выделенных максимальных общих свойств имеются одинаковые, то оставляем только одно из них и перенумеровываем эти свойства без повторений $S_1^1, \dots, S_{n_1}^1$.
Из вершины, помеченной Ω_0 , проводим ориентированные ребра к вершинам, помеченным $\Omega_1^1, \dots, \Omega_{n_1}^1$, где Ω_i^1 — множество объектов, максимальным общим свойством которых является S_i^1 .
2. Повторяем (при $l = 2, \dots, L$)⁷ процесс выделения максимальных общих свойств для пар S_i^{l-1} и S_j^{l-1} ($1 \leq i < j \leq n_{l-1}$), получив максимальные общие свойства S_{ij}^l .
Если среди выделенных максимальных общих свойств имеются одинаковые, то оставляем только одно из них и перенумеровываем эти свойства без повторений $S_1^l, \dots, S_{n_l}^l$.
Из вершин, помеченных Ω_i^{l-1} и Ω_j^{l-1} , проводим ориентированные рёбра к вершине, помеченной свойством Ω_k^l , объекты которой обладают общим для объектов из Ω_i^{l-1} и Ω_j^{l-1} свойством S_k^l . Если среди выделенных максимальных общих свойств имеется совпадающее с ранее выделенным, то вершину, соответствующую ранее выделенному максимальному свойству, отождествляем с вновь выделенной (вместе с рёбрами, инцидентными этой вершине).
3. Алгоритм завершает работу, если ни в одной паре висячих вершин построенного графа нет объектов, обладающих общими свойствами.

⁶ Для описаний на языке ИП различные — это не изоморфные, а совпадающие — это изоморфные.

⁷ Процесс завершится, так как на каждой итерации длины подформул уменьшаются.

Результат отождествления вершин Ω_k^{l+1} и $\Omega_{k'}^{l'}$, соответствующих максимальному общему свойству S_k^{l+1} , совпадающему с ранее выделенным $S_{k'}^{l'}$, представлен на рисунке 4.

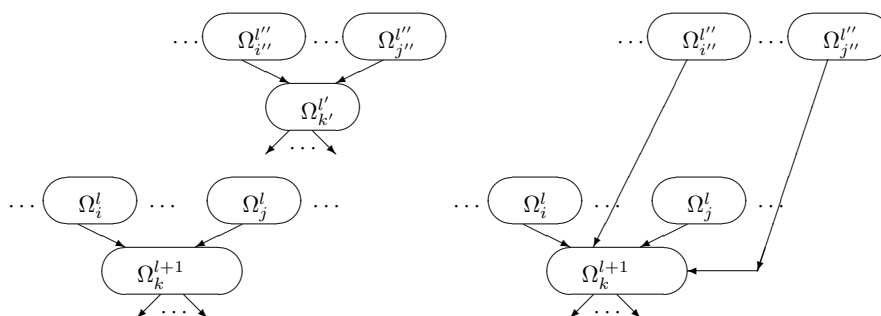


Рис. 4. Отождествление вершин Ω_k^{l+1} и $\Omega_{k'}^{l'}$, соответствующих одинаковым максимальным общим свойствам

При этом возникновение «порочного круга», при котором в графе получается ориентированный цикл, невозможно, так как длины записей наибольших общих свойств (по крайней мере, при использовании описаний на языке ИП) уменьшаются при прохождении по ориентированному пути.

Следует отметить, что описанный алгоритм построения онтологии не зависит от способа описания данных. При этом выделение максимального общего свойства существенно зависит от способа их описания.

Представление описания сложного структурированного объекта на языке ИП — это выделение максимальной общей подформулы. А совпадение максимальных общих свойств — это изоморфизм соответствующих формул.

Как указывалось в разделе 3, максимальное общее свойство объектов при задании описания сложного структурированного объекта строкой признаков — это строка, в которой стоит значение признака, если оно совпадает для всех рассматриваемых объектов, и символ λ в противном случае. Совпадение максимальных общих свойств — это равенство таких строк. Заметим, что в процессе работы алгоритма длина записи максимального общего свойства не изменяется⁸.

При этом не стоит забывать, что длина записи строки значений бинарных (или многозначных) признаков экспоненциально зависит от длины записи описания сложного структурированного объекта на языке ИП.

Как указывалось в разделе 4, максимальное общее свойство объектов при задании описания с помощью графа — это наибольший общий подграф помеченного ориентированного мультиграфа.

В любом случае алгоритм работает за экспоненциальное число шагов от длины записи описания объекта на языке ИП.

⁸ Но можно уменьшать длину записи максимального общего свойства, если указывать номер позиции в строке и значение существенного признака.

6. ЗАКЛЮЧЕНИЕ

В работе представлено сравнение трёх дискретных подходов к описанию сложных структурированных объектов: формализация описаний с помощью формул ИП, представление описаний объектов с помощью бинарных (или конечнозначных) строк, представление описаний объектов с помощью графов. Все эти подходы позволяют осуществлять переход от Данных к Информации и к Знаниям.

Показано, что использование формул ИП для описания сложных структурированных объектов, несмотря на NP-трудность решаемых задач, возникающих после формализации, реально имеют не бóльшую вычислительную сложность, чем два других подхода. Точнее,

- если объект описан средствами ИП, то при моделировании такого описания с помощью бинарной строки длина такой строки экспоненциально зависит от длины записи соответствующей предикатной формулы;
- при представлении описания объекта с помощью графа напрямую можно задавать только бинарные отношения между элементами сложного структурированного объекта, при выделении же многоместных отношений возникают экспоненциально сложные задачи поиска изоморфных подграфов.

Рассмотрена задача выделения наибольшего общего свойства сложных структурированных объектов из заданного множества. При всех трёх подходах к описанию исследуемых объектов эта задача экспоненциальна относительно длины записи описания объекта на языке ИП. При этом, несмотря на её полиномиальность относительно длины записи описания сложного структурированного объекта с помощью строки признаков, реальное время экспоненциально относительно длины записи описания объекта на языке ИП.

Безусловно, начальные сведения о математической логике на уровне булевых функций и пропозициональных формул, обработка строк, а также представление графов и работа с ними на компьютере изучаются во всех учебных заведениях, готовящих специалистов в областях, связанных с ИИ. Математическую логику, включающую исчисление предикатов и машинно-ориентированные исчисления, изучают далеко не во всех таких учебных заведениях.

Поэтому, несмотря на то, что ИП имеет бóльшую адекватность человеческому восприятию (мы рассуждаем и воспринимаем информацию в терминах отношений между элементами области рассуждения, а не в терминах свойств этой области), использование формул ИП для представления информации можно найти, в основном, в учебниках по информатике и ИИ.

Список литературы

1. *Bellinger G., Castro D., Mills A.* Date, Information, Knowledge, and Wisdom // *Systems-thinking.org*, 2004. [Online]. URL: <https://systems-thinking.org/dikw/dikw.htm> (date: 20.06.2021).
2. *Godbout A. J., Godbout G. M.* Filtering Knowledge: Changing Information into Knowledge Assets // *Journal of Systemic Knowledge Management*. 1999. Vol. 1, no. 3.
3. *Van Meter H. J.* Revising the DIKW Pyramid and the Real Relationship Between Data, Information, Knowledge and Wisdom // *Law, Technology and Humans (Symposium: What is Real about Law and Technology?)*. 2020. Vol. 2, № 2.
4. *Cooper P.* Data, information, knowledge and wisdom // *INFORMATICS*. 2017. Vol. 18, № 1. pp. 55–56. doi: <https://doi.org/10.1016/j.mpaic.2016.10.006>

5. *Darwiche A.* Three Modern Roles for Logic in AI. 2020. URL: <https://arxiv.org/pdf/2004.08599.pdf> (date: 20.06.2021).
6. *Amarilli A.* 2019. Provenance in Databases and Links to Knowledge Compilation // KOCOON workshop on knowledge compilation. Arras, France, Dec, 17, 2019. 2019. [Slides]. URL: https://a3nm.net/work/talks/kocoon2019/amarilli2019provenance_slides.pdf (date: 20.06.2021).
7. *Gelfond M.* Logic Based Approach to AI. URL: <https://fmfi-uk.hq.sk/Informatika/Uvod%20Do%20Umelej%20Inteligencie/clanky/logic-ai.pdf> (date: 20.06.2021).
8. *Calegari R., Ciatto G., Denti E., Omicini A.* Logic-Based Technologies for Intelligent Systems: State of the Art and Perspectives // Information. 2020. Vol. 11, № 3. P. 167. doi: 10.3390/info11030167
9. *Гурин В. С., Костров Е. В., Гавриленко Ю. Ю., Саада Д. Ф., Ильюшин Е. А., Чижов И. В.* Представление знаний в виде графа: основные технологии и подходы // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 4. С. 932–944. doi: 10.25559/SITITO.15.201904.932-944
10. *Zhang Z., Cui P., Zhu W.* Deep Learning on Graphs: A Survey // Journal of latex class files. 2015. Vol. 14, № 8. P. 1–24.
11. *Микони С. В.* О качестве онтологических моделей // Онтология проектирования. 2017. Т. 7, № 3. С. 347–360. doi: 10.18287/2223-9537-2017-7-3-347-360
12. *Noy N. F., McGuinness D. L.* Ontology Development 101: A Guide to Creating Your First Ontology // Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880. March 2001. Stanford: Stanford Knowledge Systems Laboratory. 2001.
13. *Mykhailiuk A., Petrenko M.* Machine learning and ontologies as two approaches for building intellectual information systems // International Journal "Information Technologies & Knowledge". 2019. Vol. 13, № 1. P. 55–75.
14. *Нильсон Н.* Искусственный интеллект. Методы поиска решений. М.: Мир, 1973.
15. *Косовская Т. М.* Доказательства оценок числа шагов решения некоторых задач распознавания образов, имеющих логические описания // Вестн. С.-Петербург. ун-та. Сер. 1. 2007. Вып. 4. С. 82–90.
16. *Kosovskaya T. M.* Fuzzy Recognition by Logic-Predicate Network // Adv. Sci. Technol. Eng. Syst. J. 2020. Vol. 5, № 4. P. 686–699. doi: 10.25046/aj050482
17. *Косовская Т. М., Петров Д. А.* Выделение наибольшей общей подформулы предикатных формул для решения ряда задач искусственного интеллекта // Вестник СПбГУ. Прикладная математика. Информатика. Процессы управления. 2017. Т. 13. Вып. 3. С. 250–263. doi: 10.21638/11701/spbu10.2017.303
18. *Косовская Т. М.* Изоморфизм формул исчисления предикатов в задачах Искусственного Интеллекта // Исследования по прикладной математике и информатике. Зап. научн. сем. ПОМИ. 2021. Vol. 499. P. 38–52.
19. *Bishop C. M.* Pattern Recognition and Machine Learning // NY.: Springer, 2006. 758 p.
20. *Oh J., Yun K., Maoz U., Kim T.-S., Chae J.-H.* Identifying depression in the National Health and Nutrition Examination Survey data using a deep learning algorithm // Journal of Affective Disorders. 2019. Vol. 257. P. 623–631. doi: 10.1016/j.jad.2019.06.034
21. *Рассел С., Норвиг П.* Искусственный интеллект: современный подход, 2-е изд.: Пер. с англ. М.: Вильямс, 2006. 1408 с.
22. *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982.
23. *Косовская Т. М.* Выделение изоморфных подформул как средство для создания логической онтологии // 13-я Российская Мультиконференция по Проблемам Управления. Материалы конференции «ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В УПРАВЛЕНИИ» (ИТУ-2020) 7–8 октября 2020 г. Санкт-Петербург, 2020. С. 84–86.

Поступила в редакцию 25.05.2021, окончательный вариант — 17.06.2021.

Косовская Татьяна Матвеевна, доктор физико-математических наук, доцент, профессор кафедры информатики математико-механического факультета СПбГУ, ✉ kosovtm@gmail.com

Computer tools in education, 2021

№ 2: 41–57

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2021-2-41-57

Comparison of Different Knowledge Representations for Complex Structured Objects in Solving AI Problems

Kosovskaya T. M.¹, Doctor of science, Professor, ✉ kosovtm@gmail.com

¹Saint Petersburg State University, 28 Universitetskiy pr., Stary Peterhof, 198504, Saint Petersburg, Russia

Abstract

The problem of knowledge representation for a complex structured object is one of the actual problems of AI. This is due to the fact that many of the objects under study are not a single indivisible object characterized by its properties, but complex structures whose elements have some known properties and are in some, often multiplace, relations with each other.

An approach to the representation of such knowledge based on first-order logic (predicate calculus formulas) is compared in this paper with two currently widespread approaches based on the representation of data information with the use of finite-valued strings or graphs.

It is shown that the use of predicate calculus formulas for description of a complex structured object, despite the NP-difficulty of the solved problems arising after formalization, actually have no greater computational complexity than the other two approaches, what is usually not mentioned by their supporters.

An algorithm for constructing an ontology is proposed that does not depend on the method of describing an object, and is based on the selection of the maximum common property of objects from a given set.

Keywords: *Complex structured object, data representation, predicate calculus formulas, feature string, knowledge graph, ontology.*

Citation: T. M. Kosovskaya, "Comparison of Different Knowledge Representations for Complex Structured Objects in Solving AI Problems", *Computer tools in education*, no. 2, pp. 41–57, 2021; doi:10.32603/2071-2340-2021-2-41-57

References

1. G. Bellinger, D. Castro, and A. Mills, "Date, Information, Knowledge, and Wisdom," in *Systems-thinking.org*, 2004. [Online]. Available: <https://systems-thinking.org/dikw/dikw.htm>
2. A. J. Godbout and G. M. Godbout, "Filtering Knowledge: Changing Information into Knowledge Assets," *Journal of Systemic Knowledge Management*, vol. 1, no. 3, 1999.
3. H. J. van Meter, "Revising the DIKW Pyramid and the Real Relationship Between Data, Information, Knowledge and Wisdom," *Law, Technology and Humans*, vol. 2, no. 2, pp. 69–80, 2020; doi: 10.5204/lthj.1470
4. P. Cooper, "Data, information, knowledge and wisdom," *INFORMATICS*, vol. 18, no. 1, pp. 55–56, 2017; doi: <https://doi.org/10.1016/j.mpaic.2016.10.006>
5. A. Darwiche, "Three Modern Roles for Logic in AI," in *Arxiv.org*, 2020. [Online]. Available: <https://arxiv.org/pdf/2004.08599.pdf>

6. A. Amarilli, "Provenance in Databases and Links to Knowledge Compilation," in *KOCOON workshop on knowledge compilation, Arras, France, Dec, 17, 2019*, 2019. [Slides]. Available: https://a3nm.net/work/talks/kocoon2019/amarilli2019provenance_slides.pdf
7. M. Gelfond, "Logic Based Approach to AI," in *fnfi-uk.hq.sk*, 2005. [Online]. Available: <https://fmfi-uk.hq.sk/Informatika/Uvod%20Do%20Umelej%20Inteligencie/clanky/logic-ai.pdf>
8. R. Calegari, G. Ciatto, E. Denti, and A. Omicini, "Logic-Based Technologies for Intelligent Systems: State of the Art and Perspectives," *Information*, vol. 11, no. 3, p. 167, 2020; doi: 10.3390/info11030167
9. V. S. Gurin, E. V. Kostrov, Yu. Yu. Gavrilenko, D. F. Saada, E. A. Il'yushin, and I. V. Chizhov, "Knowledge Graph Essentials and Key Technologies," *Modern Information Technologies and IT-Education*, vol. 15, no. 4, pp. 932–944, 2019 (in Russian); doi: 10.25559/SITITO.15.201904.932-944
10. Z. Zhang, P. Cui, and W. Zhu, "Deep Learning on Graphs: A Survey," *Journal of latex class files*, vol. 14, no. 8, pp. 1–24, 2015.
11. S. V. Mikoni, "On the Quality of Ontological Models," *Ontology of Designing*, vol. 7, no. 3, pp. 347–360, 2017; doi: 10.18287/2223-9537-2017-7-3-347-360
12. N. F. Noy and D. L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," *Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880*, Stanford, CA, USA: Stanford Knowledge Systems Laboratory, 2001.
13. A. Mykhailiuk and M. Petrenko, "Machine learning and ontologies as two approaches for building intellectual information systems," *International Journal "Information Technologies & Knowledge"*, vol. 13, no. 1, pp. 55–75, 2019 (in Russian).
14. N. G. Nilsson *Problem-Solving Methods in Artificial Intelligence*, Moscow: Mir, 1973 (in Russian).
15. T. M. Kosovskaya, "Dokazatel'stva otsenok chisla shagov resheniya nekotorykh zadachraspoznavaniya obrazov, imeyushchikh logicheskie opisanii" [Proofs of the number of steps bounds for solving of some pattern recognition problems with logical description], *Vestnik of St-Petersburg University. Ser. 1*, no. 4, pp. 82–90, 2007 (in Russian).
16. T. M. Kosovskaya, "Fuzzy Recognition by Logic-Predicate Network," *Adv. Sci. Technol. Eng. Syst. J.*, vol. 5, no. 4, pp. 686–699, 2020; doi: 10.25046/aj050482
17. T. M. Kosovskaya and D. A. Petrov, "Extraction of a maximal common sub-formula of predicate formulas for the solving of some artificial intelligence problems," *Vestnik S.-Petersburg Univ. Ser. 10. Prikl. Mat. Inform. Prots. Upr.*, vol. 13, no. 3, pp. 250–263, 2017 (in Russian); doi: 10.21638/11701/spbu10.2017.303
18. T. M. Kosovskaya, "Isomorphism of predicate formulas in artificial intelligence problems," *Zap. Nauchn. Sem. POMI*, vol. 499, pp. 38–52, 2021 (in Russian).
19. C. M. Bishop, *Pattern Recognition and Machine Learning*, New York: Springer, 2006.
20. J. Oh, K. Yun, U. Maoz, T.-S. Kim, and J.-H. Chaea, "Identifying depression in the National Health and Nutrition Examination Survey data using a deep learning algorithm," *Journal of Affective Disorders*, vol. 257, pp. 623–631, 2019; doi: 10.1016/j.jad.2019.06.034
21. S. J. Russel and P. Norvig, *Artificial Intelligence. A Modern Approach*, Moscow: Izdatelskii dom "Vil'yams", 2006 (in Russian).
22. M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Moscow: Mir, 1982 (in Russian).
23. T. M. Kosovskaya, "Isomorphous Subformules Isolation as a Means to Construct Logical Ontology," in *Proc. 13th Russian Multiconference on Management Problems. "Information technology management (ITU-2020), 7–8 Oct. 2020, St. Petersburg, St. Petersburg, Russia: Concern CSRI Elektropribor, 2020*, pp. 84–86 (in Russian).

Received 25-05-2021, the final version — 17-06-2021.

Kosovskaya Tatiana Matveevna, doctor of physical and mathematical Sciences, associate Professor, Professor of the Department of Informatics of the faculty of mathematics and mechanics of Saint Petersburg state University, ✉ kosovtm@gmail.com