

УДК 004.08

## ИНТЕЛЛЕКТУАЛЬНЫЕ ПЕРЕХОДЫ В ИСКУССТВЕННО ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ

**Попов С.В.**

*ООО «Научно-внедренческая фирма БП+», Москва, e-mail: s-v-popov@yandex.ru*

Рассматривается процесс формирования новых знаний в искусственно интеллектуальных системах, адаптирующихся к неопределенности предметной области (ПО). В связи с этим перечисляются основные формы информации: декларативная, операционная и фактическая, используемые для описания ПО в системах искусственного интеллекта. Между ними устанавливаются соотношения, касающиеся, с одной стороны, выразительности при описании ПО, и с другой – использования при решении задач. Основное требование формирования поведения систем в условиях неопределенности состоит в использовании только той информации, которая выводится из условия задачи и описания ПО. То есть система полностью предоставлена самой себе и ее приспособление к среде определяется без внесения извне новых принципов функционирования. Примером выступает игра «Крестики-нолики», для которой выводятся различные стратегии, основываясь лишь на правилах игры. Рассматривается несколько стратегий, исходными данными для которых является статистика выигрышных и проигрышных партий. То есть исходно возможность программы состоит в совершении правильных ходов и подсчета их в выигранных/проигранных партиях. Стратегия, основанная на выборе статистически выигрышных ходов, приводит к тому, что программа выигрывает чаще. Далее стратегия, основанная на статистике выигрышей/проигрышей, уточняется за счет установления корреляции между выигрышными шагами и числом свободных линий. Термин линии присутствует в условии задачи и не является внешним. Поэтому можно считать, что программа сама переходит к новой стратегии, т.е. осуществляет интеллектуальный переход, основываясь лишь на правилах игры.

**Ключевые слова:** искусственный интеллект, формирование знаний, декларативная информация, операционная информация, фактическая информация, стратегия поведения

## INTELLECTUAL TRANSITIONS IN ARTIFICIALLY INTELLIGENT SYSTEMS

**Popov S.V.**

*LLC «Nauchno-vnedrencheskaya firma BP+», Moscow, e-mail: s-v-popov@yandex.ru*

The problem of formation of new knowledge in artificially intelligent systems that adapt to the uncertainty of the subject area (PO) is considered. In this regard, the main forms of information are listed: declarative, operational and actual, used to describe the software in artificial intelligence systems. Establish relation, concerning, on the one hand, the expressiveness in the description, and on the other – use when solving problems. The main requirement for the formation of the behavior of systems under uncertainty is to use only the information that is derived from the conditions of the problem and the description of the software. That is, the system is completely left to itself and its adaptation to the environment is determined without introducing new principles of functioning from the outside. An example is the game TIC-TAC-toe, which displays a variety of strategies based only on the rules of the game. We consider several strategies, the initial data for which is the statistics of winning and losing games. I. e. initially, the possibility of the program consists in making the right moves and counting them in the won/lost games. Strategy based on the selection of statistically winning moves, leads to the fact that the program wins more often. Further, the strategy based on the statistics of wins/losses is refined by establishing a correlation between the winning steps and the number of free lines. The term line is present in the problem condition and is not external. Therefore, we can assume that the program itself is moving to a new strategy, i.e. carries out an intellectual transition, based only on the rules of the game.

**Keywords:** artificial intelligence, knowledge formation, declarative information, operational information, factual information, behavior strategy

В связи с потребностью использования автоматов там, где раньше работал лишь человек, Искусственный Интеллект (ИИ) перешел от теоретизирования в практическую плоскость. ИИ лишь начал формироваться как отдельная научная отрасль, традиционно это было объединение нескольких различных теоретических и прикладных областей, таких как математика, логики, механика [1–3]. Поэтому сейчас ИИ находится в состоянии методологической неопределенности, так как используемые в ИИ термины, традиционно понимаемые на содержательном уровне, требуют конструктивного уточнения. Например, обучение, знание, по-

нимание, обобщение, интеллектуальная деятельность и т.п. широко используют применительно к поведению автоматов, однако на самом деле это суть ярлыки, которыми снабжаются некоторые формально определенные функции программ. И сейчас необходимо уточнить, когда можно действительно говорить о проявлении интеллекта, а не о тех или иных математических, логических и пр. задачах. Исходя из этого, в статье автор описывает, что происходит реально, когда интеллект, независимо от того, искусственный он или естественный, решает те или иные задачи. Это предполагает следующие этапы осознания этой проблемы.

1. Необходимо рассмотреть, какие типы информации встречаются при описании той или иной предметной области и каким целям они служат.

2. Следует понять соотношения, существующие между этими представлениями.

3. Подлежит выделить информационные преобразования, которые с большой достоверностью можно признать интеллектуальными.

4. Наконец, следует предложить формальный аппарат для описания этих преобразований.

Исходя из этого, цель статьи состоит в том, чтобы на примере продемонстрировать, когда можно говорить об интеллектуальном акте и какие условия для этого необходимы.

*Декларативная информация.* При описании предметной области (ПО) используются три вида информации: декларативная, операционная и фактическая. Под декларативным представлением обычно понимают совокупность утверждений и фактов, имеющих место в данной ПО. При этом подразумевается, что они выступают в качестве утверждений или данных, используемых впоследствии для построения умозаключений. Сама логическая система обычно подразумевается, например это могут быть общепринятые положения («здравый смысл»), специально оговоренная логическая система: пропозициональное исчисление, исчисление первого порядка, модальная логика или прикладное исчисление, учитывающее особенности ПО и т.п. В результате содержательная информация является аксиоматикой ПО, все элементы и закономерности последней выводятся в качестве следствий из описания. Декларативное представление близко к содержательному изложению, с которым повседневно имеет дело субъект. Можно говорить, что декларативная информация в такой ее трактовке является знаниями о ПО. Ее отличительная черта – относительная компактность, в связи с тем, что главным инструментом получения новых данных о ПО служит логический аппарат. Именно использование логики, как универсального механизма формирования новых знаний, позволяет формулировать декларативное описание ПО компактно, что впоследствии при иных формах информации имеет более громоздкое выражение. Тем не менее компактное описание ПО позволяет строить сложные умозаключения в силу того, что логическая система может обладать полнотой.

*Операционная информация.* Под операционным представлением понимается совокупность тех или иных алгоритмов, по-

зволяющих путем вычисления породить все элементы ПО. Эти алгоритмы базируются на утверждениях и фактах декларативного описания. В результате каждый элемент ПО получается на заключительном шаге соответствующего вычисления. Говоря об операционном представлении, необходимо различать алгоритм и определяемое им вычисление. Алгоритм представлен формальной конструкцией, например блок-схемой, переписываемыми правилами или описанием классов, если речь идет об объектно-ориентированном программировании. Соответствующее вычисление представляет собой последовательность шагов, переводящих вычислитель в новые состояния. Сам алгоритм может быть компактным, однако определяемые им вычисления – достаточно сложными.

Связь между вычислениями и декларативным описанием выглядит следующим образом. Во-первых, вычисление можно представить как вывод в логическом исчислении, если каждый его элементарный шаг сопоставить с логическим правилом. В результате каждый элемент ПО, полученный как заключительное значение вычисления, можно трактовать как логическое следование из декларативного описания ПО. В вычислении правила вывода приобретают вид процедур, генерирующих новые данные. Таким образом, вычисления суть выводы, полученные из фактов и утверждений декларативного описания. С другой стороны, вычисления можно представить в логических терминах, например, в виде бинарной программы, описывающей единичные означивания логической формулы в расширенном базисе. Упомянутая формула представляет функцию, вычисляемую программой.

Можно увидеть аналогию между операционным представлением, которое ориентируется на получение конкретного результата в ситуациях массового характера, и умениями и навыками, приобретаемыми субъектом в результате работы в конкретной ПО. После того, как им выделены типичные навыки, использование всего декларативного описания ПО становится излишним. Типичным операционным представлением ПО служат программы, которые направлены на решение конкретных задач, хотя и носят универсальный характер в рамках спецификаций этих задач. То есть каждая программа получает корректный результат при любых допустимых входных данных. Это может быть вычисление траектории, но не вообще, а лишь того тела, которое удовлетворяет условиям среды, в которой оно перемещается.

Таким образом, вычисления можно рассматривать как выводы прикладного исчисления или бинарные программы, перечисляющие все допустимые вычисления программ. Главное, что вычисления суть производные от умений и навыков, формируемых при исследовании ПО.

*Фактическая информация.* Под фактической представлением ПО понимается множество ее элементов без указания как они получены. То есть фактическая информация не ссылается ни на декларативное, ни на операционное описание ПО, это просто множество примеров, полученных экспериментально, выведенных из утверждений декларативного представления или полученных в результате операционного представления. Фактическая информация для ПО может представлять собой как конечное, так и бесконечное множество. Например, совокупность всех следствий из конечного числа посылок с  $n$  различными переменными в пропозициональном исчислении конечна, если наложить условие, что каждое следствие имеет те же  $n$  переменных. С другой стороны, модель прикладного исчисления может быть бесконечной и представлять фактические данные, описывающие прикладное исчисление.

Итак, мы перечислили виды информации, служащей для описания ПО, основываясь на отношении к логическим исчислениям: декларативное представление – это аксиоматика ПО, операционное – выводы, а фактическое – совокупность атомарных следствий. Все эти типы представления используются искусственно интеллектуальными системами в случаях адаптивного поведения. В связи с таким разделением естественен вопрос о соотношении сложности описаний в различных терминах. Это соотношение будем определять, сравнивая с декларативным описанием ПО. Иными словами, мы берем за основу аксиоматику прикладного исчисления, которое описывает ПО. Одна и та же ПО может представляться разными прикладными исчислениями, однако все они должны быть эквиваленты, т.е. обладать совпадающими моделями.

*Сложность представления информации.* Как правило, декларативное представление более экономно по сравнению с операционным и фактическим, так как подразумевает использование логического исчисления. При этом фактические данные, как правило, получаются в результате экспериментов (обучения) и для интерпретации не подразумевают логического аппарата. Для их понимания необходимо знание понятийного базиса, в котором они описываются. Таким образом, возникает естественная задача – насколько

усложняет описание ПО отсутствие логического аппарата. Например, по конечному набору посылок можно получить множество следствий, мощность которого не менее экспоненты от числа посылок. Для графа с  $n$  узлами можно построить базис, состоящий из максимальных пустых подграфов, и число таких базисов не менее экспоненты от  $n$ . Для логической функции можно построить функциональную схему, сложность которой есть экспонента от числа переменных. Если подразумевать полноту описания ПО фактическими данными, то это влечет перебор большого числа атомарных единиц описания. И сложность такого описания существенно выше, чем сложность декларативного и операционного представления.

*Возникновение различных видов представления информации.* При исследовании новой ПО обычно вначале происходит накопление новых фактических данных, которые непосредственно связаны с решаемой задачей. Этот процесс, как правило, называется обучением, им сопровождается практически любая деятельность субъекта или объекта, который впоследствии должен принимать самостоятельные решения в этой ПО. Сбор фактов сам по себе не требует привлечения глубоких внешних знаний, надо лишь констатировать те или иные явления. Происходит простое накопление фактов, описывающихся в некотором простом понятийном базисе.

Следующий этап – обобщение, влечет возникновение новых понятий и соотношений, которые отсутствовали в фактическом описании. И это приводит к переходу к другому типу описания ПО – декларативному или операционному. Обобщение есть интеллектуальный акт, так как в результате формируется новая терминология, которая отсутствовала на этапе сбора фактов, и новые соотношения. Обобщение приводит к появлению новой декларативной информации, описывающей соотношения между объектами ПО, или операционной – представляющей новые умения и навыки, отсутствующие до этого момента. У субъекта обобщение есть результат озарения, когда он догадывается ввести новое понятие, которое по-новому представляет ПО, по сравнению с тем, что получено на стадии обучения. В формальных системах, которыми являются программы, механизм обобщения необходимо предварительно ввести, чтобы позволить программе установить соотношение между фактами, выраженными в одном понятийном базисе, и иным представлением, выраженным декларативно или операционно в ином понятийном базисе. То есть в программе, способной к обобщению, дол-

жен быть априорно заложен механизм озарения или инсайта и указано направление, в котором это озарение проявится при определенном стечении обстоятельств. Один способ такого программируемого озарения состоит в анализе условия решаемой задачи и выделении в ней тех понятий, в которых можно по-новому представить фактические данные. Обучение программы подразумевает, что в нее заложена возможность представлять данные, выраженные в одной понятийной системе, в других терминах. Но это только возможность без указания, какие конкретно новые соотношения должны быть получены. В этом состоит проявление автономного интеллекта, когда программа обладает лишь потенциалом к формированию новых закономерностей без указания, каких конкретно. Только тогда программа будет способна к обобщению, связанному с формированием новых понятий, дополняющих понятийную базу фактического представления.

Таким образом, можно заключить, что эволюционное обучение интеллектуальной программы начинается со сбора фактической информации, затем формируются новые умения и навыки путем перехода к новой понятийной базе. В результате программа будет обладать новой стратегией принятия решений в неопределенной ситуации. Тем самым просматривается первый этап перехода, который можно назвать интеллектуальным, так как в данном случае происходит выбор между различными способами обобщения фактических данных при формировании новой понятийной базы.

Сейчас мы остановимся лишь на формировании операционной семантики ПО, так как декларативная семантика, с одной стороны, требует еще одного уровня обобщения, а с другой – из практических соображений. Современные робототехнические системы в настоящее время подразумеваются как деятельные механизмы, а не как размышляющие субъекты. Именно поэтому мы исследуем умения и навыки, которые могут приобрести программы в результате эволюции. Для этого рассмотрим пример эволюционирования стратегии поведения программы игры в крестики-нолики.

*Пример обучения программы игре в крестики-нолики.* Автор поставил перед собой задачу эволюционного обучения программы игре в крестики-нолики, не используя внешние данные, которые нельзя получить кроме как из формулировки задачи и экспериментов.

1. На первом этапе был осуществлен сбор фактических данных – статистика шагов, чаще приводящих к выигрышу или проигрышу партии. Было проиграно

несколько партий, каждая из которых завершалась выигрышем или проигрышем первого игрока (компьютера). Для каждого шага из конкретной позиции собиралась статистика, т.е. сколько партий, в которые входит этот шаг, закончились выигрышем и сколько проигрышем. Для каждого игрока получилось две таблицы, одна содержала статистику положительных исходов, другая – отрицательных. Эта статистика дает хорошее представление об удачном дебюте, но не позволяет создать однозначно выигрышную стратегию. Статистически, выбирая только наиболее выгодные шаги для первого игрока, можно получить выигрыш в большинстве случаев, если оба игрока придерживаются наиболее выгодных шагов, то статистически происходит ничья. В случае, если первый игрок использует такую статистически выигрышную стратегию и играет с человеком, то человек достаточно легко выигрывает. То есть статистически выигрышная стратегия не является абсолютно выигрышной и поэтому требует уточнения. Обязательным условием такого уточнения служит, что запрещено вносить какие-либо знания, которые не выводятся непосредственно из фактических данных и условия игры. То есть программа должна получить новую стратегию логически обоснованную на известных данных.

2. Поэтому следующим этапом формирования абсолютной выигрышной стратегии является уточнение выигрышной стратегии для финала, так как именно в финале часто встречаются невыгодные для программы шаги. С этой целью уточняется статистика так называемых предвыигрышных шагов, т.е. если из некоторой позиции возможен выигрышный ход, то остальные шаги из нее не берутся во внимание, хотя они могут иметь положительную выигрышную статистику. Такое уточнение стратегии не привлекает для своей формулировки каких-либо внешних знаний, кроме условия задачи. Полученная в результате стратегия уже оказывается абсолютно выигрышной, однако ее недостатком является то, что программа в конце партии не всегда выбирает наиболее рациональные шаги, часто сводя партию вничью в случаях, когда ее можно выиграть. Строго говоря, и эта стратегия не выходит из множества статистических, она основывается на статистике выигрышных и проигрышных ходов.

3. На следующем этапе статистика связывается с понятием, которое используется в спецификации игры, но пока не встречалось. Это понятие линии, под которой здесь понимается вертикаль, горизонталь или диагональ из трех клеток. Исходно задача фор-

мулируется в том числе и в терминах линий, и поэтому это понятие не новое. Однако, на первый взгляд, между статистикой и линией связи нет. Но это только на первый взгляд. При более тщательном анализе оказывается, что статистически более выгодные ходы характеризуются тем, что занимают больше свободных линий, чем ходы, обладающие худшей выигрышной статистикой. Под свободной для первого игрока линией понимается такая, в которой отсутствует знак, поставленный вторым игроком. Аналогично определяется свободная линия для второго игрока. Так, если для первого игрока (программы) имеются два хода  $p \rightarrow p_1$  и  $p \rightarrow p_2$  со статистиками соответственно  $s_1$  и  $s_2$ , где  $s_1 < s_2$ , то второй ход покрывает не меньше свободных линий, чем первый. Тем самым статистически выигрышная стратегия приобретает уже иную формулировку в терминах свободных линий. Следует отметить, что термин свободная линия не привнесен извне, он является составной частью спецификации задачи. Отсюда следует формулировка новой стратегии, которая является модификацией статистической: надо использовать те ходы, которые приводят к занятию большего числа свободных линий. То есть мы уходим от статистической формулировки стратегии и приходим к формулировке в иных терминах. Это возможно сделать лишь установив корреляцию между статистикой ходов и числом занимаемых свободных линий. Понятно, что такая возможность вычисления корреляции в формулировке задачи не заложена, ее необходимо заложить в программу, быть может, наряду с другими механизмами, позволяющими увидеть связь различных понятий при описании ПО. В этом и состоит, как представляется, основная проблема ИИ на современном этапе: какие механизмы обобщения надо априорно закладывать в программу,

что апостериорно они позволяли формулировать стратегии поведения в новых терминах, отличных от тех, которые используются на этапе сбора фактической информации.

### Выводы

Показано, как эволюционирует стратегия, начавшись со статистически выигрышной, основанной на фактических данных этапа обучения. Затем статистика уточнялась исключением «глупых» шагов, и, наконец, устанавливалась корреляция между статистически выигрышными ходами и числом свободных линий на игровом поле. Статистика определялась на этапе обучения сборам фактических данных и их классификации, термин свободные линии принадлежит формулировке задачи. И для перехода от одной стратегии к другой программе предоставлена возможность устанавливать соотношение между понятийными базисами. В этом состоит интеллектуальный переход в искусственно интеллектуальных системах – возможностью воспользоваться заложенными в них ресурсами совершенствования [4, 5]. Как показывает опыт, на существующем этапе развития ИИ без такой возможности программа эволюционировать не будет.

### Список литературы

1. Тьюринг А. Может ли машина мыслить. М.: Ленанд, Едиториал УРСС, 2016. 128 с.
2. Рапопорт Г.Н., Герц А.Г. Биологический и искусственный разум. Часть 4. Восприятие внешнего мира индивидуальными носителями интеллекта. М.: Ленанд, 2015. 184 с.
3. Рутковский Лешек Методы и технологии искусственного интеллекта. М.: Горячая линия Телеком, 2012. 520 с.
4. Редько В.Г. Моделирование когнитивной эволюции. На пути к теории эволюционного происхождения мышления. М.: Ленанд, 2015. 256 с.
5. Макаров И.М., Лохин В.М., Манько С.В., Романов М.П. Искусственный интеллект и интеллектуальные системы управления. М.: Наука, 2012. 336 с.