

Исследование возможностей генетического алгоритма для извлечения релевантных прецедентов в системах поддержки принятия решений

К.Е. Сердюков^а, Т.В. Авдеенко^а, Е.С. Макарова^а

^аНовосибирский государственный технический университет, 630073, пр-т К. Маркса, 20, Новосибирск, Россия

Аннотация

В статье рассматриваются достоинства и недостатки прецедентного подхода для представления знаний о предметной области. В качестве одного из недостатков выделяется недостаточное быстродействие извлечения прецедентов в режиме реального времени, а также недостаточная релевантность извлекаемых прецедентов решаемой задаче. Для решения указанных проблем в настоящей работе мы предлагаем применение генетического алгоритма для извлечения прецедентов. Рассматривается формализация задачи, а также алгоритм ее решения. Приводятся результаты тестирования алгоритма. В заключении приводятся перспективы применения метода для адаптации прецедентов.

Ключевые слова: база знаний; генетический алгоритм; прецедентный подход, система поддержки принятия решений

1. Введение

Возникновение Систем поддержки принятия решений (СППР), Decision Support Systems (DSS), в середине 1960-х было связано с модельно-ориентированным подходом, популярным в то время. Первые СППР были ограничены типами реализованных в них моделей, в основном детерминированными, и практически не использовали оперативную информацию об обстоятельствах, в которых необходимо было принимать решения. К тому же, решения на основе детерминированных оптимизационных моделей, полученные в результате работы таких систем, были неочевидны и не всегда объяснимы с точки зрения лица, принимающего решения (ЛПР), что существенно препятствовало их практическому применению.

Ситуация кардинально изменилась с 1990-х, когда СППР начали интегрировать сначала с оперативными базами данных, а затем со специализированными хранилищами, построенными по технологии OLAP (On-line Analytical Processing). Появилась возможность оперативного принятия решений на основе объективной информации, накопленной в хранилищах данных. В современную эпоху интернета, когда ситуация, в которой приходится принимать решения, меняется буквально на глазах, концепция СППР претерпевает кардинальные изменения. Информации становится настолько много (даже появилось устойчивое выражение "большие данные"), что данные сами по себе перестали представлять большую ценность. То, что действительно ценно - это знания (появился даже термин "большие знания"), которые можно извлечь из данных, для решения актуальной проблемы в конкретной проблемной области. Задача получения таких (информационно насыщенных) качественных знаний, и организация их в специальном образом спроектированную базу знаний, сейчас, в эпоху больших данных, актуальна как никогда. Именно знания в виде понятных человеку (когнитивных) конструкций, очищенные от несущественного информационного мусора, дают возможность организовать поддержку принятия решений на совершенно ином качественном уровне, когда ЛПР не просто получает рекомендации от СППР, но и понимает, почему в конкретной ситуации нужно принять именно такое решение.

Методы представления и организации знаний традиционно развиваются в рамках научной дисциплины "Искусственный интеллект". В процессе развития данной научной области сформировались два базовых подхода к декларативному представлению знаний: правило-ориентированный (rule-based) и прецедентный (case-based). Появление в 1970-х экспертных систем, основанных на знаниях, связано с подходом к представлению знаний в виде правил. Именно с этими системами связаны первые реальные коммерческие успехи в области искусственного интеллекта. К 1992 году было внедрено около двух тысяч экспертных систем на основе правил [1].

Однако, несмотря на успехи, уже в самом начале развития систем на основе правил стали очевидны их недостатки. Основной проблемой стала проблема извлечения знаний из источников информации и представление их в форме правил. Чаще всего эксперты интуитивно принимают решения, основываясь на своем обширном опыте, не задумываясь, какое правило они применяют в том или ином случае. Разбиение специфического поведения эксперта на отдельные блоки, называемые правилами, является ключевой проблемой (узким местом) в создании правило-ориентированных систем. Другой проблемой является несоответствие между реальной сложностью проблемной области и очень простой структурой правила в ранних экспертных системах. В настоящее время эта проблема частично решается путем введения объектно-ориентированного описания частей правила.

С другой стороны, с 1980-х альтернативная парадигма представления знаний и рассуждений привлекает все более число приверженцев. Рассуждение на основе прецедентов, case-based reasoning (CBR), позволяет решать новые проблемы на основе адаптации опыта решения подобных проблем в прошлом, как это в действительности делает человек. В статье [2] приведены основы данного метода, предлагается обобщить знание о прошлых случаях и сохранять их в форме сценариев, которые могут быть использованы для выработки решений в подобных ситуациях.

Позднее Шенк [3] продолжил исследование роли, которую играет память о предшествующих ситуациях (прецедентах) представленных в виде некоторого контейнера знаний, при принятии решений и в процессе обучения.

В настоящее время в исследованиях по искусственному интеллекту прецедентный подход является одним из ключевых направлений, которое достаточно быстро развивается. В [4] вводится следующее определение прецедента: «прецедент – это описание проблемы или ситуации в совокупности с подробным указанием действий, предпринимаемых в данной ситуации или для решения данной проблемы». Таким образом, прецедент как единица знания включает в себя:

- описание ситуации;
- решение, которое было принято в этой ситуации;
- результат применения решения.

Существуют различные способы представления прецедентов – от простых (линейное представление) до более сложных иерархических представлений. Прецедент в общем случае включает описание проблемы, а также решение проблемы. В случае, если прецеденты из базы знаний были использованы для решения конкретной практической задачи, дополнительным компонентом в описании прецедента может быть результат (или прогноз) применения прецедента (положительный или отрицательный). Интересно отметить, что еще в работе [5], на которую часто ссылаются как на философский базис прецедентного подхода, отмечается то, что естественные понятия предметной области часто не могут быть описаны простым линейным набором свойств (атрибутов), а требуют более сложных структур для их описания.

СВР-подход к представлению знаний позволил преодолеть ряд ограничений, присущих системам на основе правил [6]. Он не требует явной модели представления знаний прикладной области, поэтому сложная проблема извлечения знаний трансформируется в задачу накопления прецедентов принятия решений. Реализация системы сводится к идентификации существенных признаков прецедента и последующему описанию прецедентов принятия решений в соответствии с этими признаками, что, конечно, является более простой задачей, чем построение явной модели знаний прикладной области.

К настоящему времени стали очевидными следующие достоинства прецедентного подхода [7,8]:

- Возможность использовать накопленный опыт напрямую, без непосредственного участия специалиста, предложившего решение схожей задачи из заранее сформированной базы;
- Сокращение времени на разработку и принятие нового решения благодаря имеющемуся опыту решения подобных задач;
- Для очень похожих задач снижается вероятность принятия ошибочного решения;
- Можно использовать хорошо развитую технологию баз данных для хранения больших объемов прецедентов;
- Перспективным представляется применение методов машинного обучения к СВР-системам с целью извлечения знаний в явном виде, а также расширения базы прецедентов.

В то же время существуют принципиальные ограничения традиционного прецедентного подхода. В-первых, при описании прецедентов специалисты часто ограничиваются только общими знаниями или описаниями проблемы, не углубляясь в сам процесс нахождения решения и ограничившись только результатами. Таким образом, структура прецедента принятия решения в данной проблемной области не соответствует сложности предметной области. Во-вторых, по мере накопления базы знаний количество прецедентов растёт, что негативно сказывается на производительности самой системы и, соответственно, на качестве принимаемого решения. На основании указанных недостатков можно выделить наиболее актуальные на сегодняшний день требования к проектированию СВР-систем:

- Необходимость в четкой индексации и организации систем сравнения прецедентов;
- Требование к выбору релевантных прецедентов, а не просто подобных на основании концепции "близости";
- Интерпретируемость извлеченных прецедентов применительно к конкретной решаемой задаче;
- Формирование решения даже в случае отсутствия схожих прецедентов в базе знаний.

В настоящей работе мы рассматриваем подход, который показывает обнадеживающие результаты в решении вышеуказанных проблем СППР, основанных на методе прецедентов. В основе предлагаемого подхода лежит метод извлечения прецедентов на основе генетического алгоритма. В следующем разделе мы рассматриваем основные особенности реализации СППР на основе прецедентного подхода с использованием так называемого СВР- цикла. В разделе 3 приводится формальное описание схемы генетического алгоритма для решения задачи извлечения прецедентов. В разделе 4 приводятся результаты тестирования реализованных алгоритмов. В разделе 5 мы формулируем выводы по работе, предлагаем перспективы дальнейшего исследования.

2. Особенности реализации СППР на основе прецедентного подхода

Прецедент можно представить различными способами: от описания множества фактов в виде строк в базе данных (таблица объект-свойство), до сложной иерархической фреймовой структуры, которая опирается на модель предметной области. Важно понимать, что при выборе представления прецедента необходимо исходить из общих целей системы. Главной проблемой при представлении прецедента является определение необходимой информации, которую необходимо включить в описание прецедента для осуществления быстрого и удобного поиска.

Механизм принятия решения на основе прецедентов представлен на рис. 1. Сначала выявляется некоторая ситуация, далее эта ситуация сопоставляется с информацией из базы прецедентов, на основании чего принимается решение. Соответствующая информация формирует контейнер, который называется прецедентом и сохраняется в базу прецедентов для использования в последующих ситуациях. Ситуация для которой был сохранен прецедент называется опорной, или базовой [9].



Рис. 1. Механизм принятия решения на основе прецедентов.

Таким образом, задача метода рассуждений на основе прецедентов состоит в решении текущей проблемы в соответствии со следующими четырьмя этапами, формирующими так называемый CBR-цикл, или цикл4R (Retrieve, Reuse, Revise, Retain), представленный на рис. 2. Основными этапами CBR-цикла являются:

- *Извлечение (Retrieve)* наиболее подходящего или подобного прецедента (множества прецедентов) из базы прецедентов (базы знаний);
- *Повторное использование (Reuse)* извлечённого прецедента для решения текущей проблемы;
- *Пересмотр и адаптация (Revise)* при необходимости получения более конкретного и точного решения;
- *Сохранение (Retain)* полученного решения в базе знаний как новый прецедент для дальнейшего его использования.



Рис. 2. CBR-цикл.

Первым и наиболее исследованным этапом CBR-цикла является извлечение прецедентов. Основной проблемой при извлечении прецедентов является выбор метода, с помощью которого вычисляется мера сходства (подобия). Метод задается на этапе создания CBR-системы. Часто используется метод ближайшего соседа, в основе которого лежит измерение степени совпадения значений свойств, определяющих прецедент. В работах [10-13] предлагались меры, основанные на введении весовой функции, учитывающей значимость каждого из признаков, составляющих описание прецедента. Однако, несмотря на многочисленные исследования в данной области, остаются проблемы недостаточной релевантности извлеченных прецедентов, а также недостаточной скорости их извлечения. Кроме того, проблема адаптации извлеченных прецедентов к реальным условиям, в которых принимаются решения, еще очень далека от

сколько-нибудь приемлемого решения. Нам представляется перспективным использование эволюционных подходов, в частности, генетического алгоритма, как с точки зрения скорости извлечения прецедентов, так и с точки зрения релевантности извлеченных прецедентов текущей ситуации. Также интересным направлением исследований представляется адаптация прецедента к текущей ситуации путем эволюционного развития.

3. Формальная постановка задачи

Предположим, что в СППР для поддержки принятия решений имеется база знаний, состоящая из n прецедентов $Case_i, i = \overline{1, n}$. Поставим задачу извлечения подмножества прецедентов $Retrieved = \{Case_{i_1}, Case_{i_2}, \dots, Case_{i_m}\}, i_k \in \{1, \dots, n\}$, наилучшим образом соответствующих решению текущей проблемы $Target$, определяемой множеством признаков $Target^j, j = \overline{1, m}$. Отметим, что каждый прецедент $Case_i$ определяется множеством признаков, часть из которых $Case_i^j, j = \overline{1, m}$, в точности соответствует признакам целевой проблемы.

Генетический алгоритм решает проблемы поиска в сложных пространствах решений на основе эволюционных принципов [14]. Формализуем задачу извлечения прецедентов в терминах генетического алгоритма следующим образом. Предположим, что популяция особей содержит P хромосом $X_p, p = \overline{1, P}$, каждая из которых представляет собой бинарный вектор размерности n , состоящий из генов, кодирующих присутствие или отсутствие соответствующего прецедента $Case_{i_k}$ в подмножестве извлеченных прецедентов $Retrieved$:

$$X_p = [X_p^1, X_p^2, \dots, X_p^n]^T, \text{ где } X_p^i = \begin{cases} 1, & \text{если хромосома } X_p \text{ соответствует извлечению прецедента } Case_i \in Retrieved \\ 0, & \text{если хромосома } X_p \text{ соответствует неизвлечению прецедента } Case_i \notin Retrieved \end{cases}$$

Таким образом, каждая хромосома соответствует определенному подмножеству извлеченных прецедентов и характеризуется определенным значением обобщенной функции непригодности $UF(X_p)$ (unfitness function), которая имеет тем большее значение, чем менее схожи целевая проблема и подмножество извлеченных прецедентов в целом:

$$UF(X_p) = \sum_{i=1}^n gap(Target, Case_i), \tag{1}$$

где $gap(Target, Case_i)$ – расхождение между целевой проблемой и прецедентом $Case_i$, вычисленное как взвешенная сумма расхождений по всем признакам

$$gap(Target, Case_i) = \sum_{j=1}^m w_j * \delta(Target^j, Case_i^j), \tag{2}$$

где веса w_j задают значимость извлекаемых признаков.

Значения расхождений $\delta(Target^j, Case_i^j)$ между отдельными признаками вычисляются различным образом для категориальных и количественных признаков. Для категориальных переменных

$$\delta(Target^j, Case_i^j) = \begin{cases} 0, & \text{если значения признака } j \text{ совпадают для целевой проблемы и прецедента} \\ 1, & \text{если значения признака } j \text{ для целевой проблемы и прецедента различаются} \end{cases}$$

Для количественных признаков $\delta(Target^j, Case_i^j) = \frac{|Target^j - Case_i^j|}{\max_i |Case_i^j| - \min_i |Case_i^j|}$

Реализованный на основе данной формальной модели метод последовательно реализует три операции генетического алгоритма: отбор, кроссовер и мутацию. Начальная популяция формируется случайным образом. Дальнейший отбор особей производится на основе функции непригодности. Чем ниже значение функции непригодности хромосомы, тем выше ее репродуктивная способность. Мы используем одноточечный кроссовер для скрещивания хромосом, а также полагаем вероятность мутации, равную 0,05.

В следующем разделе приводятся результаты расчетов реализованного алгоритма на тестовых данных.

4. Исследование работы генетического алгоритма в задаче извлечения релевантных прецедентов

Исследования предлагаемого метода проводится на двух наборах тестовых данных.

В качестве первого набора для тестирования разработанного алгоритма мы использовали базу данных Adult сайта UCI [15]. В базе прецедентов содержится 32561 записей, характеризующихся следующими атрибутами как числового, так и номинального типа:

- age – возраст, признак числового типа;
- workclass – рабочий класс, признак номинального типа;
- fnlwgt – конечный «вес» (влияние) человека, специализированный показатель числового типа;
- education – последнее оконченное обучение, признак номинального типа;
- education-num – количество оконченных различных типов обучения, признак числового типа;

- marital-status – семейное положение, признак номинального типа;
- occupation – профессия, признак номинального типа;
- relationship – отношения, признак номинального типа;
- race – раса или этническая группа, признак номинального типа;
- sex – пол, номинальный показатель;
- capital-gain – доходы, прирост капитала, признак числового типа;
- capital-loss – расходы, убывание капитала, признак числового типа;
- hours-per-week – работает часов в неделю, признак числового типа;
- native-country – страна рождения, признак номинального типа.

Целевым (классифицирующим) атрибутом, позволяющим судить о релевантности извлекаемых прецедентов, является годовой доход более или менее 50 тысяч долларов ($\leq 50K$ или $>50K$). Вычисления проводились при следующих настройках генетического алгоритма:

- Размер популяции – 100 хромосом.
- Количество популяций – 10 поколений.
- Позиция кроссовера – 50%.
- Вероятность мутации – 5%.

Позиция кроссовера показывает расположение места деления во время одноименной процедуре – при рекомбинации двух хромосом две новые получаются на основе половины родительских хромосом.

Результаты (Таблица 1) получены для различных вариантов запуска: точное совпадение номинальных показателей и три варианта разброса числовых показателей – 20%, 10% и без разброса. Для каждого варианта проведено 10 запусков, после чего вычислены средние значения. Для сравнения фиксируется время работы, количество выводимых прецедентов и категория (результат), под которую попадает большинство извлеченных прецедентов в данном варианте.

Таблица 1. Результаты работы алгоритма: запрос на получение списка молодых людей в возрасте 30 лет, закончивших только бакалавриат

| № | Точное совпадение номинальных и границы разброса числовых показателей: | | | | | | | | |
|----|--|--------|-------------------|------------|--------|-------------------|------------------------|--------|-------------------|
| | 20% | | | 10% | | | 0% (Точное совпадение) | | |
| | Время (с) | Кол-во | Результат | Время (с.) | Кол-во | Результат | Время (с.) | Кол-во | Результат |
| 1 | 136 | 1849 | $\leq 50K$ | 127 | 1253 | $\leq 50K$ | 101 | 89 | $\leq 50K$ |
| 2 | 121 | 1807 | $\leq 50K$ | 114 | 1224 | $\leq 50K$ | 95 | 85 | $\leq 50K$ |
| 3 | 116 | 1858 | $\leq 50K$ | 120 | 1226 | $\leq 50K$ | 94 | 89 | $\leq 50K$ |
| 4 | 115 | 1814 | $\leq 50K$ | 138 | 1237 | $\leq 50K$ | 102 | 90 | $\leq 50K$ |
| 5 | 116 | 1803 | $\leq 50K$ | 112 | 1212 | $\leq 50K$ | 94 | 82 | $\leq 50K$ |
| 6 | 137 | 1780 | $\leq 50K$ | 134 | 1250 | $\leq 50K$ | 89 | 86 | $\leq 50K$ |
| 7 | 134 | 1857 | $\leq 50K$ | 126 | 1261 | $\leq 50K$ | 100 | 74 | $\leq 50K$ |
| 8 | 135 | 1805 | $\leq 50K$ | 125 | 1224 | $\leq 50K$ | 89 | 100 | $\leq 50K$ |
| 9 | 151 | 1804 | $\leq 50K$ | 115 | 1227 | $\leq 50K$ | 89 | 90 | $\leq 50K$ |
| 10 | 146 | 1820 | $\leq 50K$ | 108 | 1241 | $\leq 50K$ | 97 | 85 | $\leq 50K$ |
| Ср | 130.7 | 1819.7 | $\leq 50K$ (~65%) | 121.9 | 1235.5 | $\leq 50K$ (~70%) | 95 | 87 | $\leq 50K$ (~63%) |

Исследования показывают, что ограничения, накладываемые на числовые показатели (20% отклонение, 10% отклонение, 0% отклонение), существенно снижают количество извлеченных прецедентов (при точном совпадении это число значительно уменьшается). Время обработки также уменьшается за счёт снижения необходимого количества извлекаемых прецедентов. В то же время есть исключения из этого наблюдения. Так, например, абсолютно точный результат для количественной переменной "возраст" (30 лет) дал меньший процент совпадений, чем при отклонении в 10%. В данном случае больший интервал охвата количественной переменной "возраст" позволил сдвинуть решение в определённую сторону. Результаты, выведенные по запросу, позволяет сделать вывод, что большинство молодых людей в возрасте 30 лет, закончивших только первую ступень высшего образования – бакалавриат, получают менее 50 тысяч долларов в год.

Второй тест был проведён непосредственно над числовыми данными. Для этого была выбрана база данных Iris. В данной базе хранится информация по цветкам семейства Iris, заданная следующими атрибутами:

- Длина чашелистика;
- Ширина чашелистика;
- Длина лепестков;
- Ширина лепестков;

На основе данных показателей определяется один из трёх классов цветка - Iris Setosa, Iris Versicolour или Iris Virginica. В базе хранится 150 записей. Все настройки остаются генетического алгоритма остаются точно такими же, как и в предыдущем тесте, то есть популяция в 100 хромосом, с 10 популяциями, 50% позицией кроссовера и шансом мутации 5%.

Для тестового варианта необходимо определить, к какому классу будет относиться цветок с показателями - Длина чашелистика, 6.3 см / ширина чашелистика, 2.3 см / Длина лепестков, 4.4 см / Ширина лепестков, 1.3 см. Цветок с данными показателями относится к виду *Iris Versicolour*.

Значения Н/Н обозначает неопределённость, то есть количество совпадений по нескольким классам одинаковое. Результаты представлены в таблице 2.

Таблица 2. Результаты работы алгоритма: запрос на получение списка цветов

| № | Точное совпадение номинальных и границы разброса числовых показателей: | | | | | | | | |
|----|--|--------|----------------------------------|------------|--------|----------------------------------|------------------------|--------|----------------------------------|
| | 20% | | | 10% | | | 0% (Точное совпадение) | | |
| | Время (с) | Кол-во | Результат | Время (с.) | Кол-во | Результат | Время (с.) | Кол-во | Результат |
| 1 | 0.56 | 3 | <i>IrisVersicolour</i> | 0.35 | 1 | <i>IrisVersicolour</i> | 0.05 | 1 | <i>IrisVersicolour</i> |
| 2 | 0.31 | 2 | Н/Н | 0.15 | 2 | Н/Н | 0.09 | 1 | <i>IrisVersicolour</i> |
| 3 | 0.48 | 3 | <i>IrisVersicolour</i> | 0.23 | 1 | <i>IrisVersicolour</i> | 0.12 | 1 | <i>IrisSetosa</i> |
| 4 | 0.54 | 3 | <i>IrisVersicolour</i> | 0.1 | 1 | <i>IrisVersicolour</i> | 0.1 | 2 | <i>IrisVersicolour</i> |
| 5 | 0.12 | 1 | <i>IrisVersicolour</i> | 0.12 | 1 | <i>IrisVersicolour</i> | 0.12 | 2 | Н/Н |
| 6 | 0.22 | 1 | <i>IrisVersicolour</i> | 0.09 | 1 | <i>IrisVersicolour</i> | 0.09 | 1 | <i>IrisVersicolour</i> |
| 7 | 0.37 | 1 | <i>IrisVersicolour</i> | 0.17 | 2 | <i>IrisVersicolour</i> | 0.05 | 2 | Н/Н |
| 8 | 0.38 | 1 | <i>IrisVersicolour</i> | 0.26 | 1 | <i>IrisVersicolour</i> | 0.07 | 1 | <i>IrisVersicolour</i> |
| 9 | 0.43 | 1 | <i>IrisVersicolour</i> | 0.12 | 1 | <i>IrisVirginica</i> | 0.11 | 1 | <i>IrisVersicolour</i> |
| 10 | 0.31 | 3 | Н/Н | 0.13 | 1 | <i>IrisVersicolour</i> | 0.09 | 1 | <i>IrisSetosa</i> |
| Ср | 0.372 | 1.9 | <i>IrisVersicolour</i> (~83%) | 0.172 | 1.2 | <i>IrisVersicolour</i> (~83%) | 0.089 | 1.3 | <i>IrisVersicolour</i> (~70%) |

По результатам исследования можно сделать следующие выводы. Для начала стоит заметить, из-за малого объёма самой базы, количество извлекаемых прецедентов также мало. Это не позволило определить различие в количестве извлекаемых прецедентов, как в предыдущем тесте. Тем не менее, алгоритм позволил с достаточно большой точностью извлечь прецеденты, относящиеся к тому же классу, что и исходный запрос, а именно, к классу *IrisVersicolour*.

Теперь что касается непосредственно оценки точности полученных результатов. Допустимые отклонения 20% и 10% количественных признаков показали одинаково высокий уровень точности – в районе 83%. В тоже время наиболее вариант, представленный без разброса (0%), имеет наименьшую точность классификации. Связано это с тем, что вероятность точного совпадения числовых признаков пренебрежимо мала, а последний вариант требует действительно «точного» совпадения всех чисел. А так как генетический алгоритм работает на основе случайных чисел, то требование большей точности от конечного результата не всегда ведёт к повышению этой точности в действительности. Данный результат подтвердился для двух наборов тестовых данных. В обоих случаях оптимальный результат получался в районе 10% разброса, не слишком большим, чтобы не исключать множество неподходящих чисел, и не слишком маленьком, чтобы обеспечить достаточный уровень сходимости.

5. Заключение

Таким образом, мы рассмотрели различные модели представления знаний в системах поддержки принятия решений, их достоинства и недостатки. Выделены основные этапы метода рассуждений по аналогии, особенности (СВР-цикла). Проведена формализация генетического алгоритма для задачи извлечения подмножества прецедентов, наиболее близких решаемой задаче. Приводятся результаты проведенных исследований на тестовых данных, свидетельствующих о хороших перспективах использования генетического алгоритма в методе прецедентов как для повышения быстродействия, так и для повышения релевантности извлеченных прецедентов. Представляется перспективным интеграция генетического алгоритма с генерированием нечетких правил [16] для осуществления адаптации извлеченных прецедентов к решаемой задаче.

Благодарности

Работа поддержана грантом Министерства образования и науки РФ в рамках проектной части Госзадания, проект № 2.2327.2017/ПЧ «Интеграция моделей представления знаний на основе интеллектуального анализа больших данных для поддержки принятия решений в области программной инженерии».

Литература

- [1] DTI. Knowledge-based systems survey of UK applications // Department of Trade & Industry UK, 1992.
- [2] Schank, R.C. Scripts, Plans, Goals and Understanding /Schank R.C, Abelson R.P. // Erlbau, 1977.
- [3] Schank, R.C. Dynamic Memory: A theory of reminding and learning in computers and people // CambridgeUniversityPress, 1982.
- [4] Карпов, Л.Е. Методы добычи данных при построении локальной метрики в системах вывода по прецедентам / Карпов Л.Е, Юдин В.Н // М.: Питер, 2007.

- [5] Wittgenstein, L. *Philosophical Investigations* // Blackwell, 1953.
- [6] Watson, I. Case-based reasoning: A review/Watson I., Marir F. // *The Knowledge Engineering Review*, 1994, 9(4), pp. 327-354.
- [7] Варшавский, П.П. Методы правдоподобных рассуждений на основе аналогий и прецедентов для интеллектуальных систем поддержки принятия решений / Варшавский П.П., Еремеев А.П. // *Новости искусственного интеллекта*. 2006. №3., С. 39-62.
- [8] Варшавский, П.П. Моделирование рассуждений на основе прецедентов в интеллектуальных системах поддержки принятия решений / Варшавский П.П., Еремеев А.П. // *Искусственный интеллект и принятие решений*, №2, 2009, С. 45-57.
- [9] Нечипоренко, О.А. Использование технологии Case-Based Reasoning в проектировании программных систем // *Перспективные информационные технологии и информационные среды*. № 3, 2002, С. 27-32.
- [10] Bonzano, P. "Using introspective learning to improve retrieval in CBR: A case study in air traffic control," / Bonzano P., Cunningham P., Smith B. // *Proc. 2nd Int. Conf. Case-based Reasoning*, 1997, pp. 291-302.
- [11] Cercone, N. Rule-induction and case-based reasoning: Hybrid architectures appear advantageous / Cercone N., An A., Chan C. // *IEEE Trans. Knowledge and Data Engineering*, vol. 11, pp. 166-174, 1999.
- [12] Coyle, L. Improving recommendation ranking by learning personal feature weights / Coyle L., Cunningham P. // *Proc. 7th European Conference on Case-Based Reasoning*, 2004, pp. 560-572.
- [13] Jarmulak, J. Genetic algorithms to optimize CBR retrieval / Jarmulak J., Craw S., Rowe R. // *Proc. European Workshop on Case-Based Reasoning (EWCBR 2000)*, 2000, pp. 136-147.
- [14] Yang, H.L. Two stages of case-based reasoning - Integrating genetic algorithm with data mining mechanism / Yang H.L., Wang C.S. // *Expert Systems with Applications*. 2008, v. 35, pp. 262-272.
- [15] Adult Data Set // UCI Machine Learning Repository [Электронный ресурс], Режим доступа: <http://archive.ics.uci.edu/ml/datasets/Adult>
- [16] Авдеенко, Т.В. Метод определения релевантности прецедентов на основе нечетких лингвистических правил / Авдеенко Т.В., Макарова Е.С. // *Научный вестник НГТУ*, 2016, № 1 (62), с.17-34.